الجمهورية الجزائرية الديمقراطية الشعبية
People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Scientific Research
جامعة سعد دحلب البليدة
University of SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculty of Technology

قسم الآلية والإلكتروتقنية
Departement of control systems

Master disseration

Major : Automation
Speciality : Control systems and industrial Computer Science

Presented by

Imene Yous

&

Abdellah Bouhedir

Mobile Robot Using Multiple control approaches

Proposed By : Pr.Zoubir Abdslam Benslama

Year 2022-2023

Acknowledgements

Our genuine appreciation and sincere gratefulness to ***Pr. Zoubir Abdeslam Benslama*** and ***Mr. Allaeddine Damani***for directing us along our work with educational availability, also for their effective advice and encouragement, as well as for the great sense of intellectual property.

We also thank ***Dr. Noureddinne Benilla*** for his invaluable help and directions. We wish to express our sincere thanks and our deepest gratitude to all those who have contributed from near or far to the realization of this modest work. We sincerely appreciate all the members of the jury for taking an interest in our work, and for enriching it with their suggestions and proposals.

Saad Dahleb University

Blida, June 2023

Imene Yous

Abdellah Bouhedir

Dedication

In the name of Allah, the Most Benificent, the Most Merciful.

Praise be to ALLAH who endowed us with the wonderful gift of reasoning and prompted us to acquire knowledge.

I dedicate this work to

My family and friends for their support throughout this experience, especially

My father who endorsed me at full length,

My friend ***Rihab*** who shared with me all the emotional moments while carrying out this work,

Djawed, ***Malak*** and ***Nour*** without them this work would have been finished earlier.

Imene Yous

Dedication

In the name of Allah, the Most Benificent, the Most Merciful.

Praise be to ALLAH who endowed us with the wonderful gift of reasoning, prompted us to acquire knowledge and guided us to this point.

I dedicate this work to

To my Parents and brother for being at my side all the time and for helping me get through all the obstacles.

I would like to thank

Pr. Zoubir Benslama and ***Mr. Nouredinne Bennila*** for their support to achieve this work.

And finally,

Imene for being the best partner and colleague without her this work would be coreless.

Abdellah Bouhedir

ملخص: نمت السيطرة التعاونية على أنظمة الروبوتات المتعددة المتنقلة بشكل كبير في السنوات الأخيرة، بسبب اتساع نطاق التطبيقات. الفكرة هي أن المركبات ذاتية القيادة التي تتعاون يمكن أن تحقق نتائج أفضل. في هذه الأطروحة، نحن مهتمون بالتحكم في مجموعة من الروبوتات المتنقلة غير الهولوجية بناءً على نهج القائد والمتابع. هذه دراسة مقارنة لنظريات التحكم المتعددة، والهدف هو تصميم هذه الضوابط للتكيف مع كل روبوت، ويمكن لتلك المجموعة من الروبوتات المتنقلة اتباع المسار المطلوب مع الحفاظ على التكوين الهندسي المطلوب

كلمات مفتاحية: السلوك التعاوني، أنظمة الوكلاء المتعددة، القائدالتابع، متكامل الوضع المنزلق.

Abstract: Cooperative control of mobile multi-robot systems has grown considerably in recent years, due to the wide range of applications. The idea is that autonomous vehicles that collaborate can achieve better results. In this dissertation, we are interested in controlling a group of non-holonomic mobile robots based on the leader-follower approach. This is a comparative study of multiple control theories, the objective is to design these controls to adapt to each robot, and that group of mobile robots can follow a desired trajectory while maintaining a desired geometrical configuration.

Keywords – Cooperative behavior, multi agent systems, leader-follower, integral sliding mode, PID, Backstepping

---

Résumé: Le contrôle coopératif des systèmes mobiles multi-robots a considérablement augmenté ces dernières années, en raison de la large gamme d'applications. L'idée est que les véhicules autonomes qui collaborent peuvent obtenir de meilleurs résultats. Dans cette thèse, nous sommes intéressés à contrôler un groupe de robots mobiles non holonomiques basés sur l'approche leader-suiveur. Il s'agit d'une étude comparative de multiples théories de contrôle, l'objectif est de concevoir ces contrôles pour s'adapter à chaque robot, et ce groupe de robots mobiles peut suivre une trajectoire souhaitée tout en maintenant la configuration géométrique souhaitée.

Mot clés – Controle coopératif, systèmes multi-agents, leader-suiveur, mode de glissement intégral, PID, Backstepping

List of acronyms and abbreviations

AMR: Autonomous Mobile Robot

AGV: Autonomous Guided Vehicles

MRSC: Multi Robot System Coordination

PID: Proportional, Integral, Derivative

MA: Multi Agent

AIV: Autonomous Intelligent Vehicles

VSCS: Variable Structure Control Systems

ITSM: Integral Terminal Sliding Mode

ISM: Integral Sliding Mode

SMC: Sliding Mode Control

BS: BackStepping

ISE: Integral Square Error

ITSE: Integral Time Square Error

IAE: Integral Absolute Error

ISE: Integral Time Absolute Error

Contents

List of figures

List of Figures

List of tables

List of Tables

General Introduction

In recent years, the field of robotics has witnessed significant advancements, with applications ranging from industrial automation to autonomous vehicles. One particular area of interest is robot formation control, which involves coordinating the movements and interactions of multiple robots to achieve a desired collective behavior. Effective formation control strategies play a crucial role in various applications, such as cooperative surveillance, search and rescue missions, and distributed sensing. Within the realm of formation control, the choice of control strategy is a critical decision. In this master's thesis, we aim to investigate and compare three prominent control approaches: sliding mode PID control and backstepping control. Our goal is to determine which of these strategies is better suited for a leader follower robot formation control and to provide insights into their respective strengths and weaknesses.

The problematic we address in this study revolves around the selection of an appropriate control strategy for achieving stable and precise formation control. Sliding mode control, PID control and backstepping control have demonstrated their effectiveness in various control applications. However, their performance in the context of robot formation control remains an open question.

To address this problematic, we propose a comprehensive evaluation of these two control strategies in a simulated leader follower robot formation environment. Our study will involve developing mathematical models of the robots, controllers design and path design for the leader of the robot formation. By comparing the performance of sliding mode PID control and backstepping control under different scenarios and measuring the error each time we aim to determine the relative advantages and limitations of each

approach.

In conclusion, this master's thesis aims to contribute to the field of robot formation control by investigating the suitability of sliding mode PID control and backstepping control. Through a comprehensive comparative analysis and robustness assessment, we seek to identify the control strategy that offers superior performance in terms of stability, precision, and adaptability in achieving desired formation behaviors. By addressing this problematic, we hope to provide valuable insights for the design and implementation of efficient robot formation control systems in various domains.

Chapter 1

Autonomous Mobile Robots

1.1 Introduction

In this chapter, we will comprehensively present the key notions, methodologies and applications related to mobile robots and cooperative control. Later we will explain the kinematic and the mathematical models of the mobile robot used in this study: the hilare type mobile robot.

1.2 Mobile robots

In this section, We will fully outline the essential ideas, approaches, and applications pertaining to mobile robots.

1.2.1 Robot:

A robot is a programmable machine that can autonomously or semi-autonomously perform tasks or actions typically carried out by humans. It is designed to interact with its environment through sensors and actuators, allowing it to perceive, analyze, and respond to the surrounding conditions. Robots can range from simple machines with limited functionality to highly advanced systems capable of complex operations.[**?**]

1.2.2 Multi-robot formation:

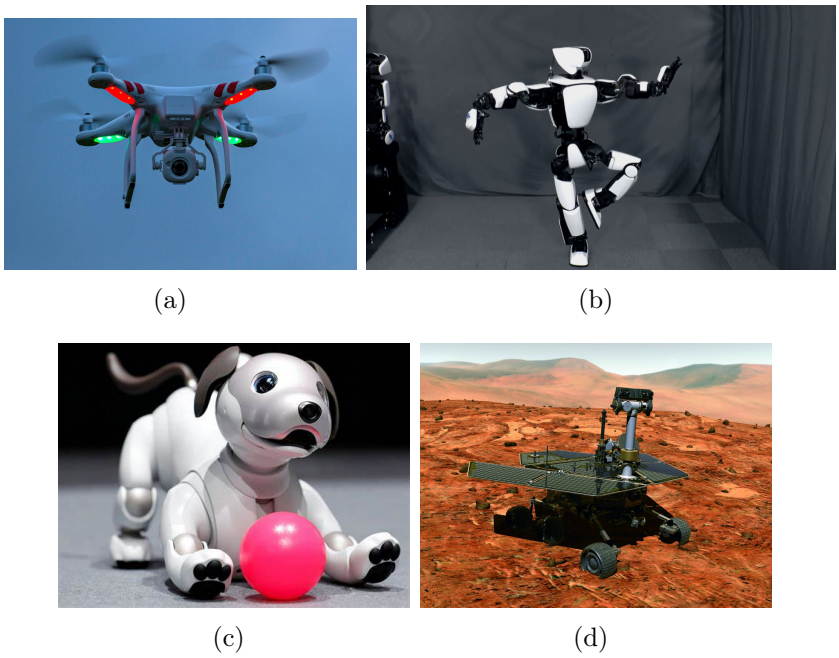
Multi-robot formation refers to a coordinated arrangement of multiple robots working together in a structured manner to achieve a common goal. It involves establishing and maintaining a desired spatial configuration or pattern among the robots, enabling them to perform tasks collectively and efficiently. The formation can be static or dynamic, depending on whether the robots maintain a fixed pattern or adapt their positions over time(Fig1.1). [**?**]

Figure 1.1: A group of mobile robots keeping formation

1.2.3 Mobile robot:

Mobile robots are distinct from fixed-base robotic manipulators as they have the ability to move in various environments, including on the ground, water, underwater, and in the air. Unlike fixed-base manipulators used in manufacturing, mobile robots operate in less structured environments and rely on a greater number of sensors. To ensure accurate data collection and utilization, it is crucial for the supervisor to have knowledge of the robot's location and orientation. Navigation plays a vital role in directing the mobile robot to a specific destination, and it must be accompanied by stable and efficient control strategies. The integration of navigation and control operations is essential. Once the robot reaches its destination, sensors can gather the required data for storage or immediate reporting to the higher-level system. Therefore, effective utilization of mobile robots necessitates a comprehensive system of functions

encompassing navigation, control, and data acquisition(See Fig1.2).[3]



(a) (b)

(c) (d)

Figure 1.2: Mobile robots: (a) Drone (b) Humanoid robot (c) Pet (d) Unmanned rover

1.2.4 Nonholonomic system:

In mathematics and physics, a nonholonomic system refers to a dynamical system or mechanical system that exhibits constraints on its motion, which are not integrable. These constraints typically involve limitations on the velocities or velocities and accelerations, rather than just positions, as in holonomic systems.

To provide a more formal definition, a nonholonomic system is defined by the presence of non-integrable constraints on the motion of its configuration space. In other words, the constraints cannot be expressed solely in terms of the positions of the system, but instead involve the velocities or higher derivatives of the positions.

Nonholonomic systems have been extensively studied in the fields of mechanics, control theory, and robotics. They often arise in practical situations where the motion of a physical system is subject to limitations imposed by external forces or mechanical constraints. Examples of nonholonomic systems include rolling bodies, vehicles with

differential drives, and systems with friction.[4]

1.2.5 Mobile robot applications:

Mobile robots have several applications employed in various domains, we cite:

1.2.5.1 Industrial and home use:

Mobile robots have the ability to navigate through various environments and communicate with each other using sensors and actuators. They are expected to be widely used in industries and homes in the near future(Figure 1.3).[5][6]

(a) (b)

Figure 1.3: (a) Industrial robots (b) Cleaning robot

1.2.5.2 Autonomous Mobile Robots (AMR) and Autonomous Guided Vehicles (AGV):

AMRs and AGVs are two types of mobile robots. AGVs navigate predefined routes in a predefined area, often used in manufacturing for repetitive tasks. AGVs lack decision-making capabilities based on artificial intelligence and require pre-programmed instructions.[7][8]

1.2.5.3 Multi-Robot System Coordination (MRSC):

In a community of mobile robots, each robot may have its own route planning approach. MRSC ensures optimal synchronization among multiple robots, with various

control approaches such as the leader-follower method.[9]

1.2.5.4 Healthcare:

Mobile robots are used in healthcare settings for tasks such as handling the coronavirus disease 2019 (COVID-19), disinfecting the environment, and supporting the elderly through visual interaction(Fig1.4).[10][11]

Figure 1.4: A mobile robot used by doctors in lockdown period

1.2.5.5 Education:

Mobile robots are used in education to provide a hands-on learning experience, allowing students to work on problems such as route planning and triangulation without the need for physical hardware(Fig 1.5).[12]

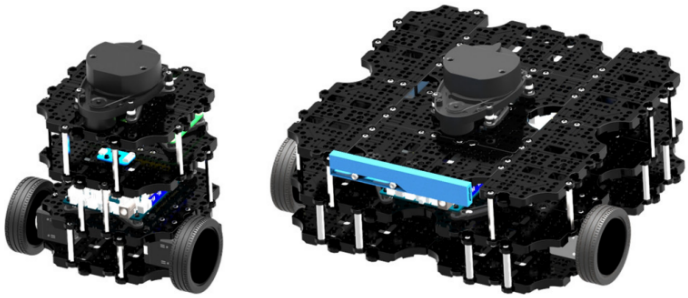


Figure 1.5: A mobile robot used for education purposes

1.2.5.6 Emergency relief:

Multi-agent (MA) architecture is used to intelligently control aerial robots for efficient surveying of disaster sites, especially when existing detection systems are ineffective. [13]

1.2.5.7 Cloud computing integration:

Cloud robotics platforms enable a fleet of autonomous robots to operate together, achieving shared targets with minimal human involvement. Cloud computing is utilized to meet real-time constraints, distributing workloads and reducing data transmission between the robot and the cloud server.[14]

1.2.5.8 Modular and Distributed System Architecture:

In multi-computer robots, a distributed system architecture is critical. Modular and distributed device architectures allow for efficient coordination and control of robotic systems.[15]

1.2.5.9 Blockchain integration:

The use of blockchain in conjunction with robotic swarm systems enhances the security, autonomy, scalability, and productivity of swarm operations. Robots can act as nodes within the network, and their transactions can be enveloped in chains.[16]

1.2.5.10 Sensor-based Localization and Planning:

Mobile robots are equipped with sensors to determine their distance from surrounding objects. Filters and position sensors based on computer vision are used for trajectory estimation and real-time position information.[17][18]

1.2.5.11 Agriculture:

Mobile robots play a significant role in modern agriculture, offering various benefits and applications. They are designed to automate and enhance agricultural tasks, increasing efficiency, precision, and productivity in the field. For example in: Crop Monitoring and Management, Precision Spraying and Irrigation, Harvesting and more(Fig1.6).[19]

Figure 1.6: A mobile robot used for agriculture

1.3 Cooperative control

Cooperative control refers to the utilization of multiple robots to perform tasks in a more efficient and reliable manner. It involves designing control strategies and coordination mechanisms that enable the robots to work together towards a common goal(Fig1.7). The aim of cooperative control is to achieve improved system performance, increased productivity, enhanced robustness to uncertainties, and efficient utilization

of resources. It encompasses various aspects such as formation control, trajectory generation, task allocation, and communication among the robots. By leveraging the collective capabilities of multiple robots, cooperative control enables the accomplishment of complex tasks that may be challenging or impossible for a single robot to achieve.

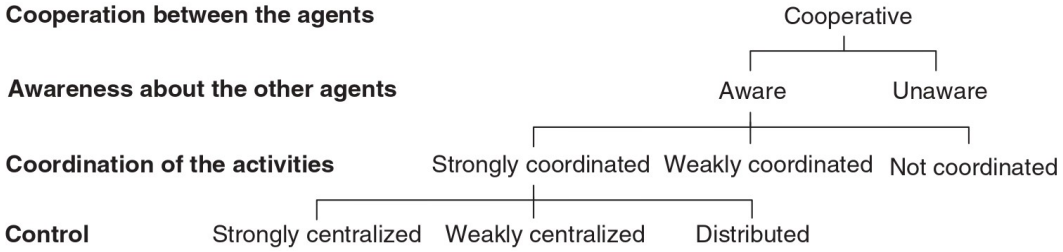
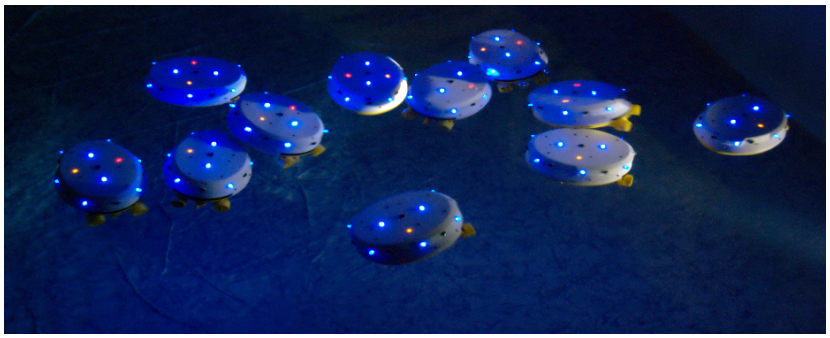


Figure 1.7: Taxonomy of the multi-agent systems with respect to the level of coordination between the agents.

The framework for decentralized cooperative control of multi-robotic systems focuses on simplicity in planning, coordination, and control. The framework consists of a two-level control hierarchy for each robot, comprising a trajectory generation level and a coordination level. The trajectory generator determines the robot's reference trajectory, while the coordination level selects the appropriate controller or behavior. Information availability and sharing play a crucial role in the design of each level, particularly in the trajectory generation. The trajectory generator can operate in a decentralized manner, where each robot generates its own reference trajectory based on the information obtained from its sensors and communication network. Alternatively, a designated leader can plan its trajectory, and other group members can follow suit. The trajectory generators are derived from potential field theory.[20]

Cooperative control can enhance the performance of multi-agent systems in many fields.(Figure 1.8)

(a)



(b)

Figure 1.8: (a) Military drone swarm (b)A group of submarines

We can notice that the idea of cooperation in Multi-Agent Systems is largely inspired by nature and the world around us. In these very powerful systems, individuals follow distant leaders without hitting their neighbors, including, but not limited to, schools of fish, swarms of bees, flocks of birds and teams of ants,...etc.(Fig1.9)

(a)

(b)

Figure 1.9: (a) Sterlings murmuration (b) A team of ants

1.4 Hilare Type Mobile Robot :

Autonomous intelligent vehicles (AIVs), which rely on concepts like pattern recognition and signal-image processing, are part of the hard study subject of mobile robotics. Wheels are one of the most crucial systems for robot mobility. When a robot is moving on flat, non-rugged ground, wheels are easier to design, build, and program than treads or legs. They frequently cost a lot less than their equivalents with legs. In compared to other options, wheel control is simpler, and they are less damaging to the surface on which they move. Another benefit is that because the robot is typically in contact with a surface, balance problems are not a major concern.

The fundamental drawback of wheels is their poor ability to maneuver over obstacles like rough ground, sharp edges, or low friction zones.

There are two common types of mobile robots that have well-designed kinematic drive chains, which reduce the chances of slip at the wheels of the robot . The two mobile robot types are Hilare-type and car-like mobile robots. Hilare-type robots(known also as differential steering) have two independently driven wheels as the drive mechanism and are usually balanced by passive caster wheel . They have good maneuvering abilities, e.g., a zero minimum turn radius, and are easier to control. They are also easier to build due to their simple drive mechanism. The Hilare-type mobile robots have different sizes and shapes depending on their application as mentioned in the previous section.[21][22] .

1.4.1 Kinematic Model of a Hilare-type Mobile Robot

A schematic figure of a Hilare mobile robot is shown in Fig.1.10. This type of robot is mostly used for indoor applications. The drive mechanism of a Hilare-type robot has two independent motors. Each of these motors power one of the robot's wheels. Thus, the actual kinematic inputs that drive the robot and affect its speed and direction of

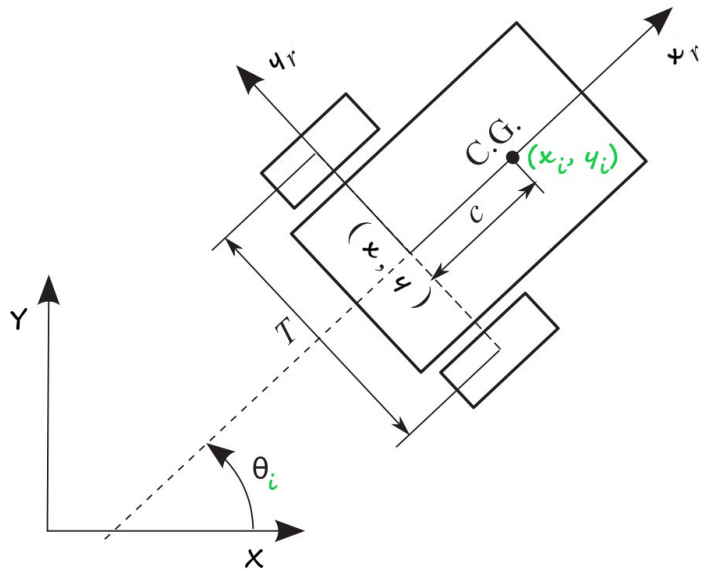


Figure 1.10: A non-holonomic mobile robot

motion are the two wheel speeds.

Now, consider the Hilare-type mobile robot shown in Fig.1.10. Assume that no-slip happens between the robot's tire and the floor during the whole motion of the robot. The first direct result of this assumption is that the velocities of the center of the robot's wheels do not have any lateral components. As a consequence, one can assume that the velocity of point (x, y), the midpoint of the line attaching the center of the wheels, does not have any lateral component and is parallel with the wheel planes. The second result of the no-slip assumption is that one can relate the velocity of point (x, y), the midpoint of the line attaching the center of the wheels, to the rotational velocity of the wheels.[23]

Before writing the kinematic equations of motion for the robot, one has to define the configuration variables of the robot. Let the coordinates of point (x, y) define the global position of the robot with respect to the inertial coordinate system x- y. Consider a line that is perpendicular to the wheel axis and goes through the point (x, y) as an orientation reference for the robot. The angle that this line makes with the positive x axis, θ, represents the orientation of the robot. The three variables that define the

geometrical configuration of the robot at any given time are [24]

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \tag{1.1}$$

Assume that the point $(x,\ y)$ on the robot moves with a linear speed of v, while the robot has an angular velocity of ω. Now, one can use the first direct result of the no-slip assumption and write the velocity components of the point $(x,\ y)$ in the inertial frame as

$$\begin{aligned} \dot{x} &= v\cos\theta \\ \dot{y} &= v\sin\theta \end{aligned} \tag{1.2}$$

Also, the rate of change of the robot's orientation is

$$\dot{\theta} = \omega \tag{1.3}$$

Combining Eqs. (1.2) and (1.3) results in the kinematic equations of motion of the robot, which can be written in the following matrix form:

$$\dot{q} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} u \tag{1.4}$$

where $u = \begin{bmatrix} v & \omega \end{bmatrix}^T$. Once the input vector u is known as a function of time, Eq. (1.4) can be numerically integrated to predict the motion of the robot. Note that one can choose inputs different than the ones that are used here. Example of other sets of inputs are the rotational velocity of the wheels, the linear velocity of the point $(x,\ y)$ and the difference of the linear velocity of the wheels, etc.

1.4.2 Mathematical model of non-holonomic mobile robot :

As shown in figure 1.10, the configuration of the i^{th} non-holonomic mobile robot R_i in the global (cartesian) frame (OXY) is

$$p_i = \begin{bmatrix} x_i \\ y_i \\ \theta_i \end{bmatrix} \tag{1.5}$$

where (x_i, y_i) is the robot position at time t ,θ defines the robot orientation.

Under the hypothesis of pure rolling and nonslipping [25], the kinematic constraint of the nonholonomic mobile robot is given as

$$\dot{y}_i \cos\theta_i - \dot{x}_i \sin\theta_i = c\dot{\theta}_i \tag{1.6}$$

where c is the distance from the rear axle to the front of the robot.

From the kinematic constraint (1.6), the kinematics model of the nonholonomic mobile robot R_i can be described as

$$\dot{p}_i = \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_i \\ \omega_i \end{bmatrix} \tag{1.7}$$

where $u_i = \begin{bmatrix} v_i & \omega_i \end{bmatrix}^T$ are are the linear and angular velocities of the robot.

1.5 Conclusion

In this chapter, we have explored the fundamental definitions and methodologies that form the basis of this thesis. These concepts provide the necessary groundwork for our study. In the next chapter, we will delve into the mathematical modeling of

the system under investigation. This modeling process will enable us to gain a deeper understanding of the underlying dynamics and behavior of the system.

Chapter 2

Modelling the Nonholonomic mobile robot Leader-follower formation

2.1 Introduction

Controlling the formation of a group of multi mobile robots means how to stabilize and maintain a desired formation shape in a distributed manner. The formation control according to several researches can be classified as virtual structure approach, behavior – based approach and leader follower strategy.

In leader follower method the robot in formation designed as a leader move along a predefined trajectory while the other robots (the followers) are to maintain a desired formation, a stable leader follower formation control low is investigated based on kinematic model and trajectory tracking technique. The advantage of this method: it's easy to understand and implement, also this approach has some disadvantages it asks for centralized control approach (follower use the position of leader as input control), which make it less suitable for large number of robots. There is no explicit feedback from the follower to the leader. For example if the leader moves to fast, or if the follower is blocked by an obstacle the formation will be destroyed. Another disadvantage is that if the leader has failed, the entire formation cannot be kept and the consequence will be very serious. In this chapter, the mathematical modelling of the leader-follower

formation in cartesian coordinates will be presented.

2.2 Leader-follower formation model :

In order to achieve the coordinated formation control, it is necessary to describe the relationship between the robots. In this thesis, we consider the leader–follower formation control scheme shown in figure (2.1), the follower robot tails its leader with a desired separation σ_{lf}^{d}, desired bearing φ_{lf}^{d} and desired orientation θ_{f}^{d} ($\theta_{lf}^{d} = \theta_{l}$). Consider $p_l = \begin{pmatrix} x_l & y_l & \theta_l \end{pmatrix}^T$ is the actual posture of the leader robot and (v_l, ω_l) are its the control entries, $p_f = \begin{pmatrix} x_f & y_f & \theta_f \end{pmatrix}^T$ is the actual posture of the follower robot and (v_f, ω_f) are its the control entries, the desired position of the follower in the leader's coordinate system is (x_d, y_d), and the actual position in the global coordinate system is $p_f^d = (x_c, y_c)$ can be obtained using the following equation :

$$p_f^d = \begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} x_l \\ y_l \end{bmatrix} + \begin{bmatrix} \cos\theta_l & -\sin\theta_l \\ \sin\theta_l & \cos\theta_l \end{bmatrix} \begin{bmatrix} x_d \\ y_d \end{bmatrix} \tag{2.1}$$

With:

$$\sigma_{lf}^{d} = \sqrt{x_d^2 + y_d^2} \quad and \quad \varphi_{lf}^{d} = \arctan(\frac{y_d}{x_d}) - \theta_l + \pi$$

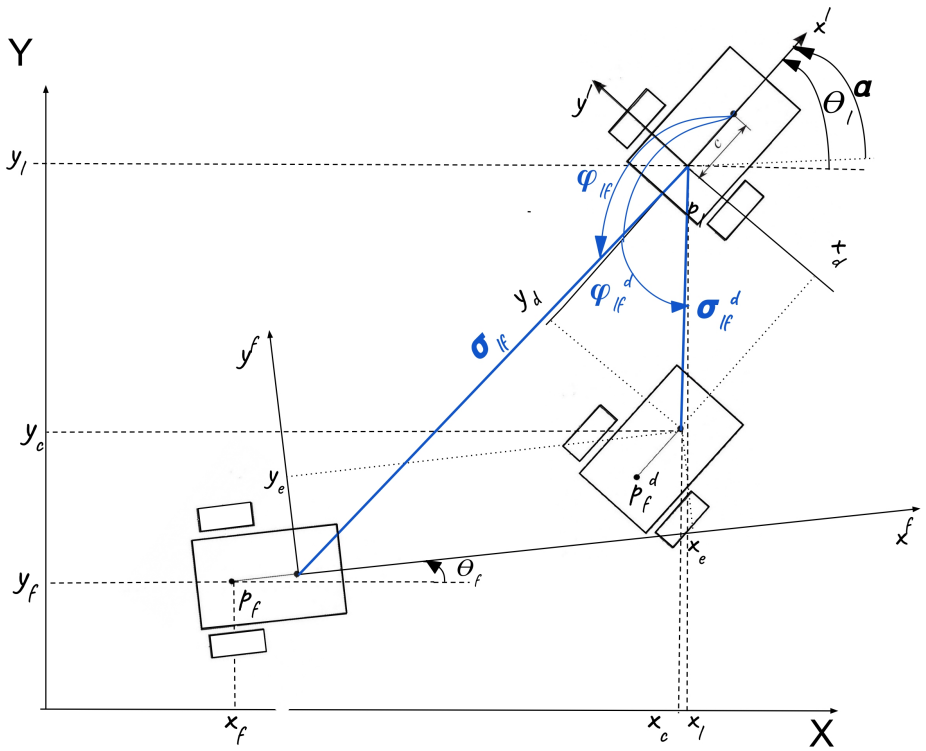


Figure 2.1: Leader-follower formation

By using the geometric relationships between the robots [26], the desired position of the follower robot p_f^d would be :

$$p_f^d = \begin{bmatrix} x_l - c\cos\theta_l + \sigma_{lf}^d \cos\left(\varphi_{lf}^d + \theta_l\right) \\ x_l - c\sin\theta_l + \sigma_{lf}^d \sin\left(\varphi_{lf}^d + \theta_l\right) \\ \theta_l \end{bmatrix} \tag{2.2}$$

The real posture of the follower robot p_f would be :

$$p_f = \begin{bmatrix} x_l - c\cos\theta_l + \sigma_{lf} \cos\left(\varphi_{lf} + \theta_l\right) \\ x_l - c\sin\theta_l + \sigma_{lf} \sin\left(\varphi_{lf} + \theta_l\right) \\ \theta_l \end{bmatrix} \tag{2.3}$$

Using (1.2), (1.3), the derivative of (2.1) and some trigonometric formulas mentioned

below. The follower's linear velocities in the global coordinate are

$$\dot{x}_c = \dot{x}_l - x_d\dot{\theta}_l \sin\theta_l - y_d\dot{\theta}_l \cos\theta_l$$
$$\dot{y}_c = \dot{y}_l + y_d\dot{\theta}_l \cos\theta_l - y_d\dot{\theta}_l \sin\theta_l$$

$$\dot{x}_c = v_l \cos\theta_l - x_d\omega_l \sin\theta_l - y_d\dot{\theta}_l \cos\theta_l$$
$$\dot{y}_c = v_l \sin\theta_l + x_d\omega_l \cos\theta_l - y_d\omega_l \sin\theta_l$$

$$\dot{x}_c = (v_l - y_d\omega_l)\cos\theta_l - x_d\omega_l \sin\theta_l$$
$$\dot{y}_c = (v_l - y_d\omega_l)\sin\theta_l + x_d\omega_l \cos\theta_l$$

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \end{bmatrix} = \begin{bmatrix} \cos\theta_l & -\sin\theta_l \\ \sin\theta_l & \cos\theta_l \end{bmatrix} \begin{bmatrix} v_l - y_d\omega_l \\ x_d\omega_l \end{bmatrix} \tag{2.4}$$

The distance between the front and the center of the robot p_f is c in the global coordinate, so :

$$x_{of} = x_f + c\cos\theta_f$$
$$y_{of} = y_f + c\sin\theta_f \tag{2.5}$$

Deriving (2.5), the velocities would be :

$$\dot{x}_{of} = v_f \cos\theta_f - c.\omega_f \sin\theta_f$$
$$\dot{y}_{of} = v_f \sin\theta_f + c.\omega_f \cos\theta_f \tag{2.6}$$

Using the equations (2.6) , (2.8) , the tracking error for the formation in the follower's

coordinate system :

$$\begin{bmatrix} x_{fe} \\ y_{fe} \end{bmatrix} = \begin{bmatrix} \cos\theta_f & \sin\theta_f \\ -\sin\theta_f & \cos\theta_f \end{bmatrix} \begin{bmatrix} x_c - x_{of} \\ y_c - y_{of} \end{bmatrix} \tag{2.7}$$

Also, the angular difference between the leader and the follower is notes as

$$\alpha = \theta_l - \theta_f \;\; et \;\; \dot{\alpha} = \omega_l - \omega_f \tag{2.8}$$

To keep the desired leader-follower formation, the separation-separation and the separation-bearing desired between the leader and the follower must be maintained. Therefore, the challenge of control is designing the follower robot's control entries (v_f, ω_f) as shown below :

$$\lim_{t\to\infty} x_{fe} = 0, \;\; \lim_{t\to\infty} y_{fe} = 0 \quad o \quad |\alpha| \prec \delta(\delta \succ 0) \tag{2.9}$$

Then , $\lim_{t\to\infty}(p_f^d - p_f) = 0$

2.2.1 The error in the leader-follower formation :

Assuming that (x_d, y_d) are constants. Using (2.4), (2.6),some trigonometric formulas and deriving (2.9), the error is obtained by:

$$x_{fe} = x_c\cos\theta_f - x_{of}\cos\theta_f + y_c\sin\theta_f - y_{of}\sin\theta_f$$

$$y_{fe} = -x_c\sin\theta_f + x_{of}\sin\theta_f + y_c\cos\theta_f - y_{of}\cos\theta_f$$

$$\dot{x}_{fe} = (v_l - y_d\omega_l)\cos\alpha - x_d\omega_l\sin\alpha - v_f + \omega_f(-x_c\sin\theta_f + x_{of}\sin\theta_f + y_c\cos\theta_f - y_{of}\cos\theta_f)$$

$$\dot{y}_{fe} = (v_l - y_d\omega_l)\sin\alpha + x_d\omega_l\cos\alpha - \omega_f(c + x_c\cos\theta_f - x_{of}\cos\theta_f + y_c\sin\theta_f - y_{of}\sin\theta_f)$$

In a matrix form, the error is

$$\begin{bmatrix} \dot{x}_{fe} \\ \dot{y}_{fe} \\ \dot{\theta}_{fe} \end{bmatrix} = \begin{bmatrix} (v_l - y_d\omega_l)\cos\alpha - x_d\omega_l \sin\alpha - v_f + \omega_f y_e \\ (v_l - y_d\omega_l)\sin\alpha + x_d\omega_l \cos\alpha - \omega_f + \omega_f(c + x_e) \\ \omega_f^d - \omega_f \end{bmatrix} \tag{2.10}$$

2.2.2 The separaration-bearing formulation:

As shows Fig(2.1), the goal of the formation control is to make the follower robot track the leader with desired separation $\sigma_{lf}^d(t)$ and desired bearing $\varphi_{lf}^d(t)$ with initial condition $z_{lf}^0 = [\sigma_{lf}^0 \ \ \varphi_{lf}^0]^T$ such that

$$\lim_{t\to\infty} \|e_{lf}(t)\| = \lim_{t\to\infty} \|z_{lf}^d(t) - z_{lf}(t)\| \tag{2.11}$$

where $e_{lf}(t)$ is the formation tracking error between l-f robots,$z_{lf}^d(t) = [\sigma_{lf}^d \ \ \varphi_{lf}^d]^T$ and $z_{lf}(t) = [\sigma_{lf} \ \ \varphi_{lf}]^T$ are the desired state and the current state respectively.

The time derivate model for relative distances $\sigma_{lf}^x(t)$ and $\sigma_{lf}^y(t)$ between $l - f$ robots can be obtained using (See Figure(2.1))

$$\sigma_{lf}^x(t) = x_l - x_f - c.\cos\theta_f(t)$$
$$\sigma_{lf}^y(t) = y_l - y_f - c.\sin\theta_f(t)$$

Using equations (1.2),(1.3) the derivative with respect to t is

$$\begin{aligned} \dot{\sigma}_{lf}^x(t) &= v_l\cos\theta_l(t) - v_f(t)\cos\theta_f(t) + c\omega_f(t)\sin\theta_f(t) \\ \dot{\sigma}_{lf}^y(t) &= v_l\sin\theta_l(t) - v_f(t)\sin\theta_f(t) + c\omega_f(t)\cos\theta_f(t) \end{aligned} \tag{2.12}$$

where $\sigma_{lf}(t) = \sqrt{(\sigma_{lf}^x(t))^2 + (\sigma_{lf}^y(t))^2}$and $\varphi_{lf}(t) = \arctan\frac{\sigma_{lf}^y(t)}{\sigma_{lf}^x(t)} + \pi - \theta_l(t)$ and then

the following leader–follower tracking model is obtained as described in

$$\dot{\sigma}_{lf} = -v_l \cos\varphi_{lf} + v_f \cos\gamma_{lf} + c.\omega_f \sin\gamma_{lf}$$
$$\dot{\varphi}_{lf} = \frac{1}{\sigma_{lf}}\big(v_l \sin\varphi_{lf} - v_f \sin\gamma_{lf} + c.\omega_f \cos\gamma_{lf}\big) - \omega_l$$

$$\begin{bmatrix}\dot{\sigma}_{lf}(t)\\ \dot{\varphi}_{lf}(t)\end{bmatrix} = \begin{bmatrix}\cos\gamma(t) & c\sin\gamma(t)\\ \frac{-\sin\gamma(t)}{\sigma_{lf}(t)} & \frac{c\cos\gamma(t)}{\sigma_{lf}(t)}\end{bmatrix}\begin{bmatrix}v_f(t)\\ \omega_f(t)\end{bmatrix} + \begin{bmatrix}\cos\varphi_{lf}(t) & 0\\ \frac{\sin\varphi_{lf}(t)}{\sigma_{lf}(t)} & -1\end{bmatrix}\begin{bmatrix}v_l(t)\\ \omega_l(t)\end{bmatrix} \tag{2.13}$$

$$\dot{z}_{lf}(t) = F(\sigma_{lf}(t), \gamma_{lf}(t))u_f(t) + L(\sigma_{lf}(t), \varphi_{lf}(t))u_l(t) \tag{2.14}$$

where $\dot{z}_{lf}(t) = [\dot{\sigma}_{lf}(t)\ \ \dot{\varphi}_{lf}(t)]$, $F(\sigma_{lf}(t), \gamma_{lf}(t)) = \begin{bmatrix}\cos\gamma(t) & c\sin\gamma(t)\\ \frac{-\sin\gamma(t)}{\sigma_{lf}(t)} & \frac{c\cos\gamma(t)}{\sigma_{lf}(t)}\end{bmatrix}$,

$u_f(t) = [v_f(t)\ \ \omega_f(t)]^T$, $L(\sigma_{lf}(t), \varphi_{lf}(t)) = \begin{bmatrix}\cos\varphi_{lf}(t) & 0\\ \frac{\sin\varphi_{lf}(t)}{\sigma_{lf}(t)} & -1\end{bmatrix}$, $u_l(t) = [v_l(t)\ \ \omega_l(t)]^T$,
and $\gamma_{lf}(t) = \theta_f(t) + \varphi_{lf}(t) - \theta_l(t)$ is the relative orientation between $l - f$ robots.

2.3 Conclusion

In our leader-follower formation, the followers' trajectories are typically not predetermined; rather, their leader decides them as they move in real time in order that the robot system can efficiently carry out its tasks. The tracking control for a single non-holonomic mobile robot is further developed in this chapter to the training control for a group of non-holonomic mobile robots allowing the follower to follow the leader in real time via the recommended kinematic controller.

Chapter 3

Leader-follower formation control

3.1 Introduction

The robot formation can be defined as a branch of robotic system which studies the coordination of a group of mobile robots to give a formation with assured shape. This formation of robots are used to perform collective tasks. The robot can perform the task alone or implemented more completely by the robots are bevy[27]. In the utmost years, the multi mobile robots have been used in distinct field of applications including essence localization , mobile sensor network, and medical operations . These application are more complex and difficult to be determined. The problems on how to implements these operations and controlling a group of robots to make them to move as a group to perform their tasks in good manner and fundamental one.In order to fulfill the goal of formation, several problems have to be solved. These problems are: the problem of knowing the initial position of these robots, and the path planning from initial to the final locations in formation. These problems has become more attractive and challenging for researchers to solve by designing new controller approaches and methods. In this chapter, linear and non linear approaches will be discussed to solve the problem of path planning in order to keep the desired formation.

3.2 Sliding mode control

Variable structure control systems (VSCS) evolved from the pioneering work in Russia's of Emel'yanov [28] and Barbashin in the early 1960s [29] [30]. The ideas did not appear outside of Russia until the mid 1970s when a book by Itkis [31] and a survey paper by Utkin [32] were published in English. The concept of VSCS has since been incorporated to the design of robust regulators, model-reference systems, adaptive schemes, tracking systems, state observers, and fault detection schemes. In VSCS, the control law has been intentionally altered during the control process to correspond to certain defined rules that depend on the state of the system. The ideas have successfully been applied to problems as diverse as automatic flight control, control of electric motors, chemical processes, helicopter stability augmentation systems, space systems and robots.

A decision rule and numerous feedback control regulations characterize sliding mode control, a special case of VSCS. The decision rule, also known as the switching function, requires a measure of the current setup's performance as an input and outputs the particular model of feedback controller that ought to be implemented at that exact moment in time. The VSCS have the goal of drive, then force, the system state to be approximate to the switching function whenever in sliding mode. The desired sliding mode dynamics have been achieved and sustained due to the control law.

3.2.1 Dynamics in the sliding mode: Non linear systems:

Let us consider the following nonlinear system affine in the control:

$$\dot{x} = A(x,t)u(t) + b(x,t) \tag{3.1}$$

and a set of m switching surfaces

$$s(x,t) = [s_1(x,t), ..., s_m(x,t)]^T = 0 \tag{3.2}$$

where $s_i(x,t)$ are continuous and differentiable functions of x and t. It is assumed, without loss of generality, that the system confined to each of the sliding manifolds $s_i = 0 \;\; i = 1, 2, ..., m$ is stable, at least in the Lyapunov sense.

The aim of the control input $u(t)$ is to bring the system states onto the intersection of the chosen manifolds in a finite time and then move the system states towards the origin of the state space(Fig3.1).

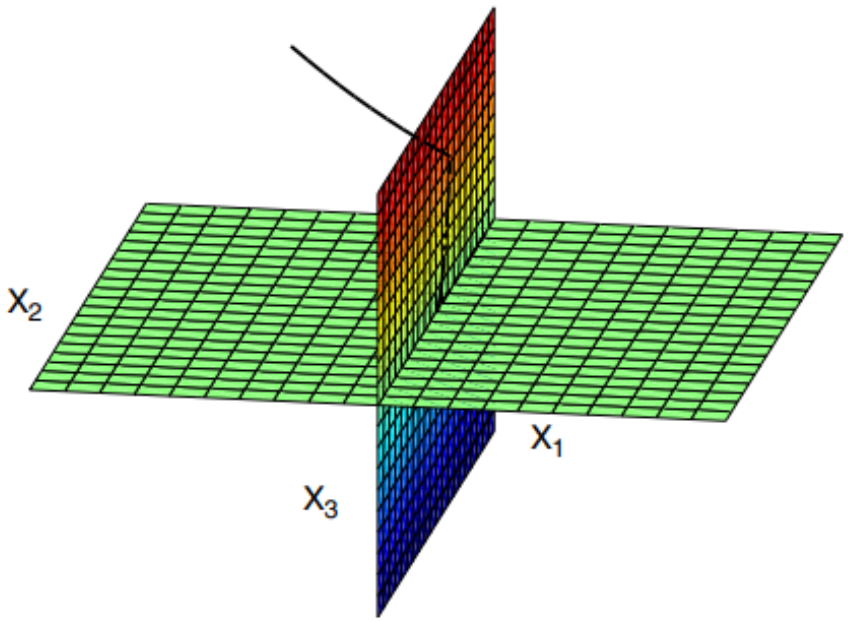


Figure 3.1: A sliding mode motion with two control functions

An extension of the results from [33]leads to:

- the associated equivalent control :

$$u_c = -\big[\frac{\partial s}{\partial x}A(x,t)\big]^{-1}\frac{\partial s}{\partial x}b(x,t) \tag{3.3}$$

obtained by writing that $\dot{s}(x) = \frac{\partial s}{\partial x}[A(x,t)u + b(x,t)] = 0$; The equivalent control is an intermediate control action lying between the two extreme values of the discrete set of available controls, which we have so far limited to switch position functions taking values in the discrete set $\{0, 1\}$.The equivalent control is therefore bounded by these two extreme values

$$0 < u_{eq} < 1$$

- the resulting dynamics, in sliding mode :

$$\dot{x}_e = \big[I - A(x_e,t)\big[\frac{\partial s}{\partial x_e}A(x_e,t)\big]^{-1}\frac{\partial s}{\partial x_e}\big]b(x_e,t) \tag{3.4}$$

3.2.1.1 The reachability condition:

As stated by some, the motion during sliding was independent to the control. Nevertheless, it is evident that the control must be designed in an efficient manner that forces the trajectories to the switching surface and retains them there once they remain. The condition can be used to express the sliding surface's local attractivity.

$$s\dot{s} < 0 \tag{3.5}$$

which is called the reachability condition [34].

3.2.1.2 Equivalent Control and the Reaching Law Approach:

Equivalent control constitutes an equivalent input which, when exciting the system, produces the motion of the system on the sliding surface whenever the system is on the surface [35]. Suppose, the system trajectory intersects the sliding surface at time t_1, and a sliding mode exists. The existence of sliding mode implies that for all t, the system trajectory would satisfy $\dot{s}(x) = 0$. Thus, the equivalent control that maintains

the sliding mode is the input u_{eq} satisfying

$$\dot{s} = \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} b(x,t) + \frac{\partial s}{\partial x} A(x,t) u_{eq} = 0 \tag{3.6}$$

Assuming that the matrix $\frac{\partial s}{\partial x} A(x,t)$ is non-singular, the equivalent control may be calculated as

$$u_{eq} = -\Big(\frac{\partial s}{\partial x} A(x,t)\Big)^{-1} \Big(\frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} b(x,t)\Big) \tag{3.7}$$

The equivalent control, however, operates only after the state trajectory reaches the sliding manifold. To shift the system states into the sliding manifold, a formal control technique must be developed, which may employ variable structure. The so-called reaching law approach is one of the methods for designing a sliding mode controller in a generic dynamical system. The switching function's dynamics are explicitly described in the reaching law method. Following that, a general framework can be used to define sliding function dynamics.

$$\dot{s} = -Q f_s(s) - K sgn(s) \tag{3.8}$$

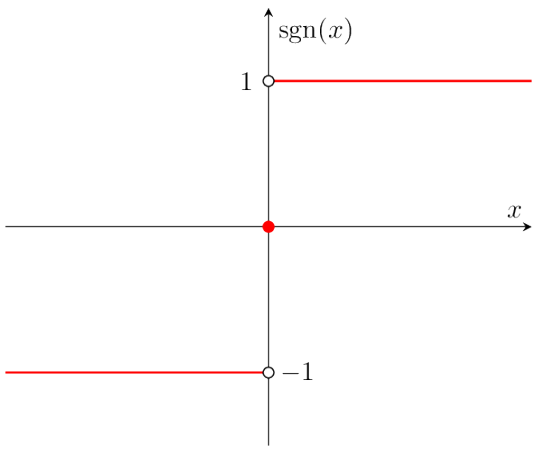


Figure 3.2: Sign function

where,$sgn(s)$ is signum function(Fig3.2), Q and K are positive definite matrices of appropriate dimensions and $f_s(s)$ is such that $f_{s,i}(s)s_i > 0, \forall s_i \neq 0$. Some of the possible dynamics are shown below [36] :

1. The constant rate reaching law :

$$\dot{s} = -Ksgn(s) \tag{3.9}$$

2. The constant plus proportional rate reaching law :

$$\dot{s} = -Qs - Ksgn(s) \tag{3.10}$$

3. The power-rate reaching law:

$$\dot{s} = -K_i|s_i|^{\alpha_i} \quad , 0 < \alpha_i < 1 \tag{3.11}$$

A control input can now be constructed using (3.8) for the system (3.1)

$$u(t) = -\Big(\frac{\partial s}{\partial x}A(x,t)\Big)^{-1}\Big(\frac{\partial s}{\partial t} + \frac{\partial s}{\partial x}b(x,t) + Qf_s(s) + Ksgn(s)\Big) \tag{3.12}$$

It is worthy to note at this juncture that the reaching law based control in (3.12) would become the equivalent control (3.7) when the system state is on the sliding manifold.

3.2.2 The chattering phenomenon

An ideal sliding mode does not exist in practice since it would imply that the control commutes at an infinite frequency. The discontinuity in the feedback control causes a specific dynamic behavior in the vicinity of the surface known as chattering in the presence of switching flaws, such as switching time delays and small time constants in the actuators (Fig 3.3).

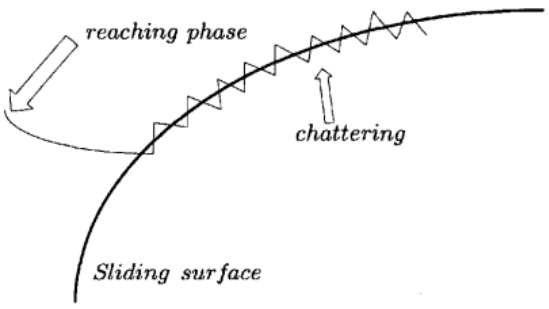


Figure 3.3: The chattering phenomenon

This phenomenon is a disadvantage because, even if it is filtered at the process' output, it may stimulate unmodeled high frequency modes, which lowers the system's performance and even increases the risk of instability [37]. Additionally, chattering causes high heat losses in electrical power circuits and severe wear of moving mechanical parts. Many techniques have been created to lessen or stop this chattering because of this.One of them is a regulatory strategy in a region around the switching surface, which, in the simplest instance, entails no more complicated than swapping out the signum function with a continuous approximation with an important advantage in the boundary layer: for instance, sigmoid functions (see [38]) or saturation functions as shown in Figure 3.4.

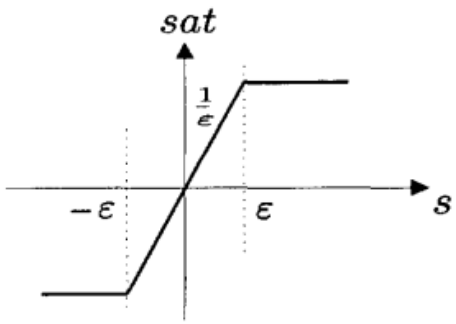


Figure 3.4: Saturation function $sat(s)$

However, although the chattering can be removed, the robustness of sliding mode is also compromised. Another solution to cope with chattering is based on the recent theory of higher-order sliding modes.

3.2.3 Integral Terminal Sliding Mode Formation Control:

ITSM control is used to obtain the control input for the follower robot to follow the time varying leader robot's trajectory. The proposed control law does not consider leader robot's velocity $u_l(t) = [v_l(t)\omega_l(t)]^T$ and this is more meaningful and practical because there may exist sensors limitation and/or communication link failure. Therefore, ITSM controller is designed for stable formation tracking control in the presence of model uncertainties and disturbances with no need of leader's velocity information.

The design of ITSM involves two steps, first is the selection of an appropriate terminal sliding surface $S_{lf}(t)$ which defines the desired system dynamics. The second step is to design a discontinuous control law so that the system dynamics stay on terminal sliding surface and the stability of the system is guaranteed. Let a continuous non-singular integral terminal sliding surface is defined as [39]

$$S_{lf}(t) = \beta_1 e_{lf}(t) + \beta_2 \int_0^t e_{lf}^{q/p}(k)dk \tag{3.13}$$

where e_{lf} is the formation tracking error between the leader-follower robots, $\beta_k \in \mathbb{R},\ k = 1\ and\ 2$, are strictly positive constants, both p and q are positive odd integers and satisfying $p \geqslant q \geqslant 0$.

The time derivative of $S_{lf}(t)$ is given as

$$\dot{S}_{lf}(t) = \beta_1 \dot{e}_{lf}(t) + \beta_2 e_{lf}^{q/p} \tag{3.14}$$

In order to keep the output trajectory on the sliding surface $S_{lf}(t)$, the equivalent control law $u_{i_{eq}}$ is obtained by setting $\dot{S}_{lf} = 0$

$$\dot{S}_{lf}(t) = \beta_1 \dot{e}_{lf}(t) + \beta_2 e_{lf}^{q/p} = 0 \tag{3.15}$$

Substituting (2.12) and into (3.14) we will give the equivalent control law for the follower robot f as

$$u_{f_{eq}} = F_n^{-1}(\sigma_{lf}(t), \gamma_{lf}(t))[\dot{z}_{lf}^d(t) + \frac{\beta_2}{\beta_1} e_{lf}^{q/p}(t)] \tag{3.16}$$

where $F_n^{-1}(\sigma_{lf}(t), \gamma_{lf}(t))$ is defined as

$$F_n^{-1}(\sigma_{lf}(t), \gamma_{lf}(t)) = \begin{bmatrix} \cos\gamma_{lf} & -\sigma_{lf}(t)\sin\gamma_{lf} \\ \frac{\sin\gamma_{lf}}{c} & \frac{\sigma_{lf}(t)\cos\gamma_{lf}}{c} \end{bmatrix} \tag{3.17}$$

In order to keep the system state on sliding surface the equivalent control law $u_{f_{eq}}(t)$ must be augmented with switching control law $u_{f_{sw}}(t)$ which is defined as

$$u_{fsw} = F_n^{-1}(\sigma_{lf}(t), \gamma_{lf}(t))[ksat(S_{lf}(t))] \tag{3.18}$$

where $k = \begin{bmatrix} k_1 & 0 \\ k_2 & 0 \end{bmatrix}$ is a switching gain matrix. The saturation function, sat(.), is used to eliminate chattering phenomenon and ϵ is the width of a boundary layer that defines the relationship between steady-state errors and tracking accuracy. The sat(.) function is given as

$$sat(S_{lf}(t)) = \begin{cases} \frac{S_{lfk}(t)}{\epsilon} & if \quad \left|\frac{S_k(t)}{\epsilon}\right| \leq 1 \\ sgn(\frac{S_{lfk}(t)}{\epsilon}) & if \quad \left|\frac{S_k(t)}{\epsilon}\right| \geq 1 \end{cases} \tag{3.19}$$

where sgn(.) is the standard signum function. The complete control law for leader follower tracking control can be written as

$$u_f(t) = F_n^{-1}(\sigma_{lf}(t), \gamma_{lf}(t))[\dot{z}_{lf}^d(t) + \frac{\beta_2}{\beta_1} e_{lf}^{q/p}(t) + ksat(S_{lf}(t))] \tag{3.20}$$

3.2.3.1 Stability :

Consider the Lyapunov function

$$V_{lf}(t) = \frac{1}{2} S_{lf}^T(t) S_{lf}(t) \succ 0 \tag{3.21}$$

Taking the time derivative of $V_{lf}(t)$ yields

$$\dot{V}_{lf}(t) = S_{lf}^T(t) \dot{S}_{lf}(t) \tag{3.22}$$

and using the value of $S_{lf}(t)$ from Eq. (3.14) will give

$$\dot{V}_{lf}(t) = S_{lf}^T(t)[\beta_1 \dot{e}_{lf}(t) + \beta_2 e_{lf}^{q/p}(t)] \tag{3.23}$$

Substituting (2.12) one can obtain

$$\dot{V}_{lf}(t) = S_{lf}^T(t)[\beta_1(z_{lf}^d(t) - F_n(\sigma_{lf}(t), \gamma_{lf}(t))) + \beta_2 e_{lf}^{q/p}(t)] \tag{3.24}$$

Substituting the control law in (3.20) into (3.24), implies

$$\dot{V}_{lf}(t) = S_{lf}^T(t)[-ksat(S_{lf}(t))] \tag{3.25}$$

$$\dot{V}_{lf}(t) = -k\|S_{lf}(t)\| \leqslant 0 \tag{3.26}$$

From (3.26), it can be observed that $S_{lf}(0) = 0$ and $\dot{V}_{lf}(t) \leqslant 0$, the system will always remain on the terminal sliding surface $S_{lf}(t)$, therefore separation and bearing errors will converge to zero in finite time t_{fin} as $t \to \infty$. The convergence time can be obtained on the surface $\dot{S}_{lf}(t) = 0$, as

$$\dot{e}_{lf}(t) = -\frac{\beta_2}{\beta_1} e_{lf}^{q/p}(t) \tag{3.27}$$

and

$$t_{fin} = \frac{|e_{lf}(0)|^{1-q/p}}{\beta_2/\beta_1(1-q/p)} \tag{3.28}$$

3.3 PID controller

The most widely used control technique is the PID controller, which performs a number of crucial tasks, including providing feedback, eliminating steady state offsets by integral action, and predicting the future through derivative action. PID controllers are adequate for a variety of control issues, especially when the process dynamics are benign and the performance demands are low. PID controllers are widely used across many industries. The controllers come in a variety of shapes and sizes. Each year, tens of thousands of standalone systems in boxes for one or a few loops are produced. An essential component of a distributed control system is PID control. Several control systems with specialized purposes also incorporate the controllers. More than 95% of the control loops used in process control are PID-type loops, with PI control making up the majority. Because they were regarded as trade secrets, many beneficial aspects of PID control were not publicly known. Techniques for mode changes and anti-windup are common examples.

The standard PID control configuration is as shown in Fig (3.5). It is also sometimes called the “PID parameter form.”

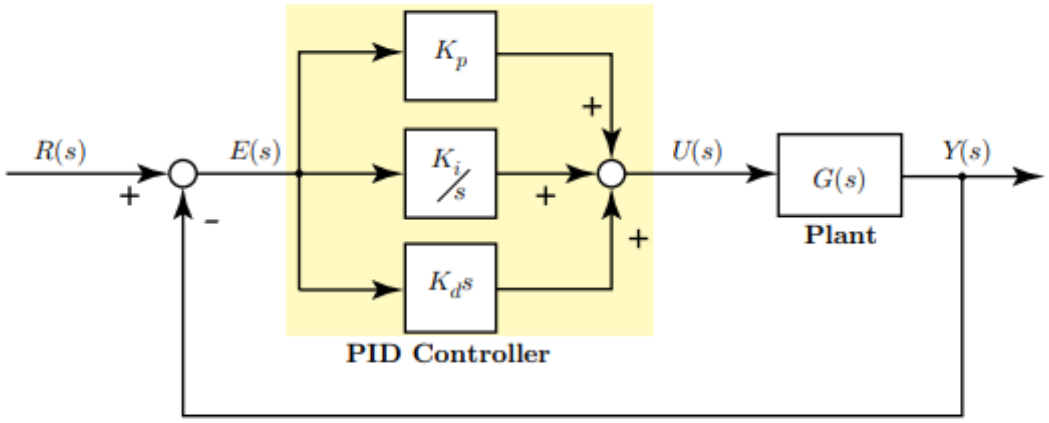


Figure 3.5: PID controlled system

In this configuration, the control signal $u(t)$ is the sum of three terms. Each of these terms is a function of the tracking error $e(t)$. The term K_p indicates that this term is proportional to the error. The term K_i/s is an integral term, and the term $K_d s$ is a derivative term. Each of the terms works "independently" of the other.

3.3.1 Interpretation of the terms :

A typically PID controller involves three types of control actions: a proportional action, an integral action, and derivative action, which can be mathematically expressed as

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{de(t)}{dt} \tag{3.29}$$

where K_p, K_i, and K_d denote the proportional, integral, and derivative gain, respectively.

The proportional control action is proportional to the current control error, which can be expressed as

$$u(t) = K_p e(t) \tag{3.30}$$

where K_p is the proportional gain. Since it implements the standard operation of

increasing control effort when the control error is big (with the appropriate sign), the function of such a control is pretty evident.

The integral action is proportional to the integral of the control error, i.e.,

$$u(t) = K_i \int_0^t e(\tau)d\tau \tag{3.31}$$

where K_i is the integral gain. With the integral action, the resultant control makes use of the past values of the control error to generate its control signal.

The derivative control takes the following form,

$$u(t) = K_d \frac{de(t)}{dt} \tag{3.32}$$

where K_d is the derivative gain, which makes use of the predicted future values of the control error.

3.3.2 PID controller conception

The difference between a measured process variable and the desired setpoint is used by a PID controller to calculate an error value. The controller modifies the process control inputs in an effort to reduce the error. The following is the stated equation for the formation control applied PID controller [40] :

$$v_f = v_l + [K_p E_\tau + K_i \sum E_\tau + K_d \Delta E] \tag{3.33}$$

$$\omega_f = K_p E_v + K_i \sum E_v + K_d \Delta E \tag{3.34}$$

The error $e(t)$ is decomposed with respect to E_τ and E_v, as shown in Fig (3.6)

$$\begin{aligned} E_\tau &= d_{lf} \cos \phi_{lf} - {d_{lf}^d} \cos {\phi_{lf}}^d \\ E_v &= d_{lf} \sin \phi_{lf} - {d_{lf}^d} \sin {\phi_{lf}}^d \end{aligned} \tag{3.35}$$

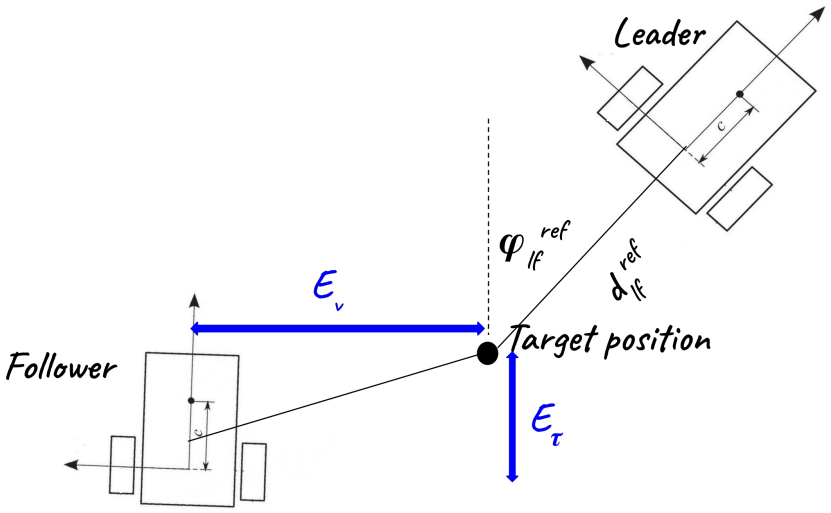


Figure 3.6: Decomposition of error

3.4 Backstepping

Backstepping is a recursive Lyapunov-based scheme proposed in the beginning of 1990s which provides an alternative to feedback linearization. The technique was comprehensively addressed by Krstic, Kanellakopoulos and Kokotovic in [41]. The idea of backstepping is to recursively design controllers by treating certain state variables as "virtual controllers" and designing intermediate control laws for them. The main idea of the backstepping method is to divide the whole system into n cascaded subsystems. As a result, the state of the first subsystem acts as the control variable for the next subsystem. In the backward stepping method, the ideal inputs that would excite the desired output of the first subsystem are first calculated. Because subsystems are cascaded, the input to the first subsystem is automatically taken from the output of

the second subsystem. The ideal input to the second subsystem is calculated similarly to the first subsystem.The desired inputs for all the subsystems are calculated in a sequential manner until the last subsystem is arrived. As a case in point,the desired input to the last subsystem provides an expression for the actual control input, so the designer can implement a state feedback law for the entire nonlinear system [43]. This backstepping recursive approach is often an advantageous feature when designing control laws for complex nonlinear dynamical systems. Backstepping achieves the goals of stabilization, tracking and is particularly successful in the area of nonlinear control [42].

3.4.1 Backstepping controller conception

Using Eqs (2.7),(2.8) and applying simple trigonometric identities, the tracking error for leader–follower formation is obtained as

$$e_f = \begin{bmatrix} x_{fe} \\ y_{fe} \\ \theta_{fe} \end{bmatrix} = \begin{bmatrix} \sigma_{lf}^d \cos\left(\varphi_{lf}^d + \alpha\right) - \sigma_{lf} \cos\left(\varphi_{lf} + \alpha\right) \\ \sigma_{lf}^d \sin\left(\varphi_{lf}^d + \alpha\right) - \sigma_{lf} \sin\left(\varphi_{lf} + \alpha\right) \\ \theta_f^d - \theta_f \end{bmatrix} \tag{3.36}$$

Assuming that the desired separation σ_{lf}^d and the desired bearing φ_{lf}^d are constants. It then follows that $\dot{\sigma}_{lf}^d = 0$ and $\dot{\varphi}_{lf}^d = 0$. Thus, one can obtain the error dynamics of the mobile robot by taking the time derivative of (3.36) along (2.2) and (2.3) as

$$\begin{aligned} \dot{x}_{fe} &= v_l \cos\alpha + \omega_f y_{fe} - v_f - \sigma_{lf}^d \omega_l \sin\left(\varphi_{lf}^d + \alpha\right) \\ \dot{y}_{fe} &= v_l \sin\alpha - \omega_f x_{fe} - c.\omega_f + \sigma_{lf}^d \omega_l \cos\left(\varphi_{lf}^d + \alpha\right) \\ \dot{\theta}_{fe} &= \omega_f^d - \omega_f \end{aligned} \tag{3.37}$$

Due to the nature of the error dynamics system (3.37), input-output feedback control cannot be used to solve this model (3.37). Due to the incomplete constraints of each robot and the control goal of the leader-follower formation, the direction of each follower is different when the formation rotates, so the reference direction cannot be selected as $\theta_f^d = \theta_l$. Here we choose the derivative of the reference direction as follows:

$$\dot{\theta}_f^d = \big(v_l \sin\alpha + \sigma_{lf}^d \omega_l \cos(\varphi_{lf}^d + \alpha) + 2k_2 y_{fe}\big)/c \tag{3.38}$$

The asymptotic stability of all error states can be discovered by selecting (3.38). Consequently, the error dynamic system

$$\begin{aligned} \dot{x}_{fe} &= \omega_f y_{fe} - v_f + v_l \cos\alpha - \sigma_{lf}^d \omega_l \sin(\varphi_{lf}^d + \alpha) \\ \dot{y}_{fe} &= -\omega_f x_{fe} - c.\omega_f + v_l \sin\alpha + \sigma_{lf}^d \omega_l \cos(\varphi_{lf}^d + \alpha) \\ \dot{\theta}_{fe} &= \big(v_l \sin\alpha + \sigma_{lf}^d \omega_l \cos(\varphi_{lf}^d + \alpha) + 2k_2 y_{fe}\big)/\big(c - \omega_f\big) \end{aligned} \tag{3.39}$$

Consider the following backstepping control inputs [44]

$$\begin{aligned} v_f &= k_1 x_{fe} + v_l \cos\alpha - \sigma_{lf}^d \omega_l \sin(\varphi_{lf}^d + \alpha) \\ \omega_f &= \big(v_l \sin\alpha + \sigma_{lf}^d \omega_l \sin(\varphi_{lf}^d + \alpha) + k_2 y_{fe} + k_3 \theta_{fe}\big)/c \end{aligned} \tag{3.40}$$

where k_1, k_2 and k_3 are positive contants. Hence, the closed-loop kinematics error dynamic becomes

$$\begin{aligned} \dot{x}_{fe} &= \omega_f y_{fe} - k_1 x_{fe} \\ \dot{y}_{fe} &= -\omega_f x_f e - k_2 y_{fe} - k_3 \theta_{fe} \\ \dot{\theta}_{fe} &= (-k_3 \theta_{fe} + k_2 y_{fe})/c \end{aligned} \tag{3.41}$$

To prove that the trajectory tracking control system (3.41) under the controller laws (3.40) is asymptotically stable and the tracking errors converge to zeros, we choose the following Lyapunov function candidate

$$V(t) = \frac{1}{2}(x_{fe}^2 + y_{fe}^2) + \frac{d k_3 \theta_{fe}^2}{2k_2} \tag{3.42}$$

It's obvious that $V(t) \geq 0$, and $V(t) = 0$ if and only if $x_{fe} = 0$, $y_{fe} = 0$ and $\theta_{fe} = 0$.

The derivative of the Lyapunov function (3.42) is given by

$$\dot{V}(t) = x_{fe}\dot{x}_{fe} + y_{fe}\dot{y}_{fe} + \frac{d k_3 \theta_{fe}\dot{\theta}_{fe}}{k_2} \tag{3.43}$$

Substituting (3.41) into (3.43), we have

$$\dot{V}(t) = -k_1 x_{fe}^2 - k_2 y_{fe}^2 - k_3 \theta_{fe}^2 \leq 0 \tag{3.44}$$

Therefore, the tracking controller (3.40) for the follower robot is stable.

3.5 Conclusion

In control theory, the PID controller is considered as a classical and is frequently used for a wide range of electrical and mechanical systems. A PID controller has considerable advantages over more complicated formulations, which makes it well-suited to a wide variety of applications. Among the main advantages of PID controllers include: their ease of implementation and can be applied without knowing the dynamical model, the ease of tuning the gains (parameters), and the reliability and consistency of their algorithms.

To overcome some of the shortcomings of the linear controllers, such as guaranteeing stability only around an equilibrium point and the inability to handle the nonlinear part of the model, therefore, several nonlinear controllers have been derived based on the nonlinear dynamics of the system. Among them:

- The ISM has been significantly used for its attractive finite-time convergence characteristics and robustness to parametric uncertainties and perturbations. Since the SMC suffers from the chattering phenomena caused by the reaching law and has high control effort, many researchers have been working on those troubles. One of the proposed solutions is the integral sliding mode control. The integral action added to the sliding manifold has the ability to eliminate the reaching phase and reduce the chattering on the control inputs.

- The backstepping control technique is based on the Lyapunov theory for the stability study. The application of this theory is often inhibited by difficulties in finding the appropriate Lyapunov function. However, the recursive look of

the backstepping command provides a systematic algorithm that makes this task easier.

In this chapter, we defined every approach and designed its leader-follower formation controller in the interest of examining them in the next chapter.

Chapter 4

Simulation and results

4.1 Introduction

In this chapter, we'll illustrate the theoretical findings from the previous chapter via some simulation results obtained by implementing the control laws in MATLAB.

Using different scenarios(trajectories) to test the effectiveness of each control law. First, we consider a group of five robots: a leader and its followers, each follower will tail its leader while keeping the desired angle and bearing; many tests were done so the optimal constants for each controller to be smoother were found. Then every controller will be represented by a follower in the same simulation so the differences between them would be clear.

Simulations are being conducted to assess the effectiveness of the proposed study method.

4.2 1^{st} trajectory: Line

Consider a leader and four followers. Suppose the distance of the rear axle to the front of the robot is $c = 0.1m$, the linear velocity v_l and angular velocity ω_l of the leading robot are respectively $[1; 0]^T$ for its trajectory to be a straight line, and the initial posture of the leading robot is $[0; 0; 0]$. In order to maintain the formation

of leader-follower, the follower must maintain the desired separation $\sigma_{lf}^{d} = 0.5m$ and orientation $\varphi_{lf}^{d} = \frac{\pi}{2}$ with his leader.

4.2.1 Integral sliding mode control:

The efficiency of the proposed tracker controls (3.31) based on the switching function (3.24) is used with $\lambda = 1.2$ and $k = 8.5$.

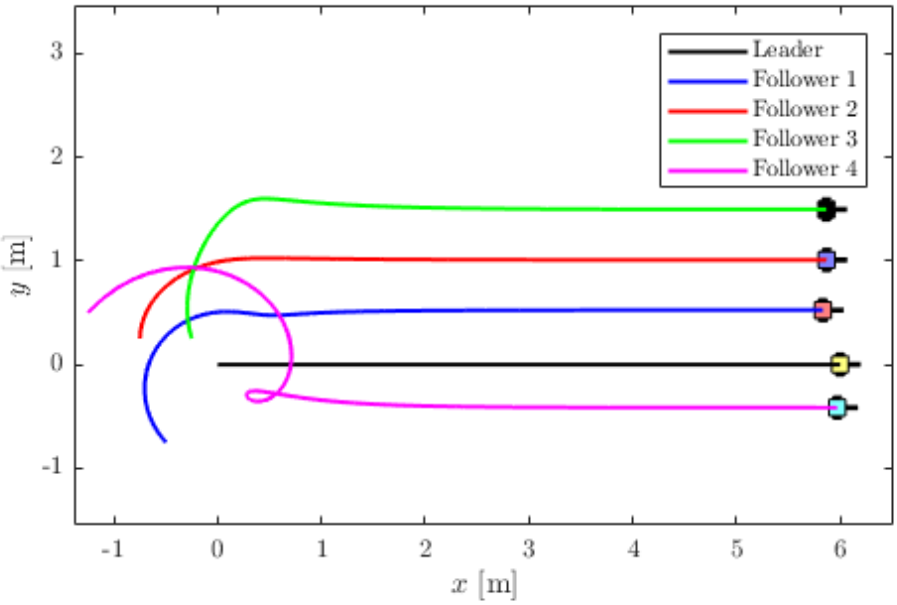
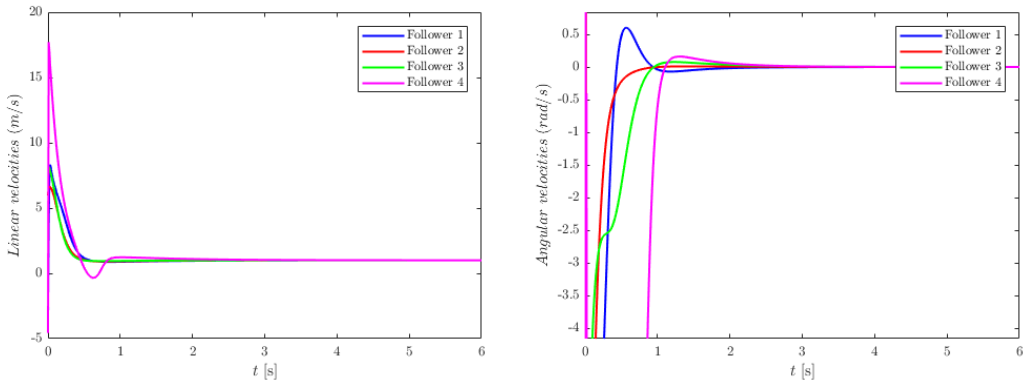
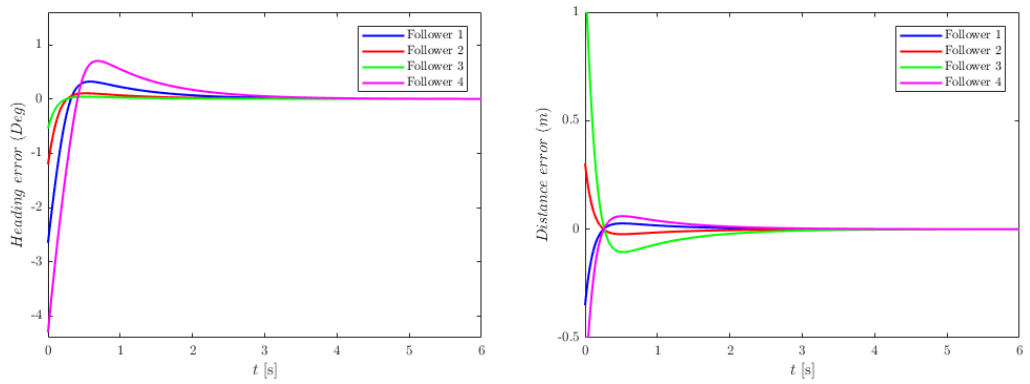


Figure 4.1: Leader and followers paths using ISM



(a) (b)

Figure 4.2: Followers velocities using ISM:(a) Linear (b) Angular



(a) (b)

Figure 4.3: Followers tracking errors using ISM: (a)bearing error (b) Separation error

4.2.1.1 Results:

Figure(4.1) is the trajectory of the robots. It shows that the followers can follow the leader and maintain the desired separation and rotation.Figure (4.2) shows respectively the linear and angular velocity of the follower. From the latter, the followers starts with a high linear and angular speeds. This implies that the initial linear and angular accelerations are very important, that is, the force and the following torque are very important. Figures (4.3) shows respectively the followers heading angle and show that followers tracking errors converge to zero after (t = 3s).

4.2.2 PID control:

Using the previous initial conditions (leader position, velocities and followers positions) The performance of the suggested tracker controls (3.44, 3.45) is evaluated through the constants $k_p = 2$,$k_i = 4$ and $k_d = 0.2$.

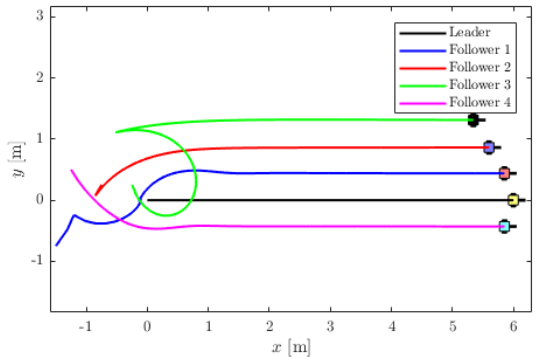
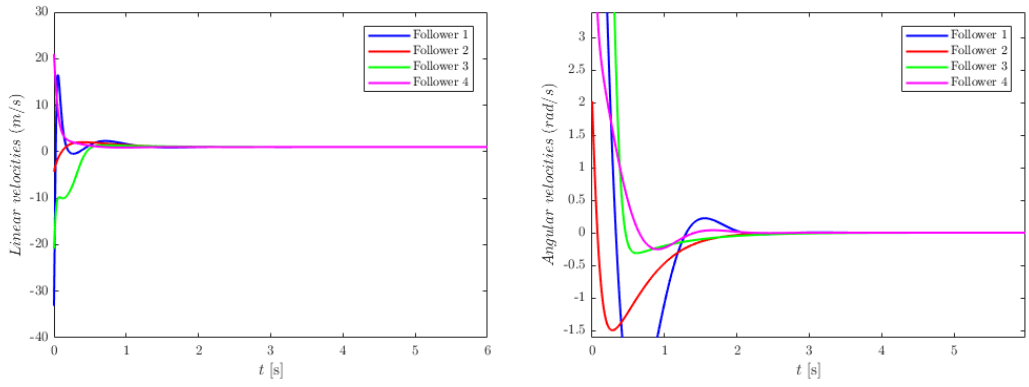
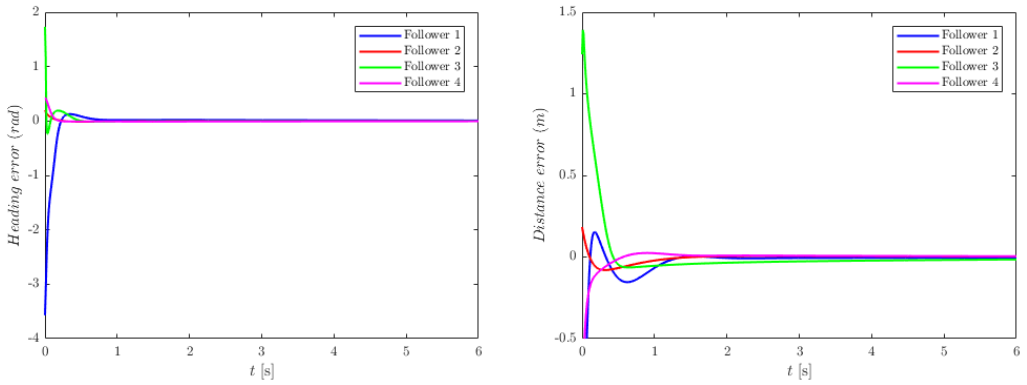


Figure 4.4: Leader and followers paths using PID



(a) (b)

Figure 4.5: Followers velocities using PID:(a) Linear (b) Angular



(a) (b)

Figure 4.6: Followers tracking errors using PID: (a)bearing error (b) Separation error

4.2.2.1 Results:

As seen in Figure (4.4), the followers follow their leader while preserving the correct separation and rotation. Figure(4.5) represent the follower's linear and angular velocities, respectively. After (t=2s), the followers' velocities achieve constant values. Figure (4.6) which depict the direction and distance errors of the followers, respectively, demonstrate how the followers' tracking errors converge to zero after (t = 2s).

4.2.3 Backstepping:

Using the previous initial conditions (leader position, velocities and followers positions) and desired distance and bearing. The performance of the suggested tracker controls (3.51) is evaluated through the constants $k_1 = 10$,$k_2 = 2$ and $k_3 = 1$.

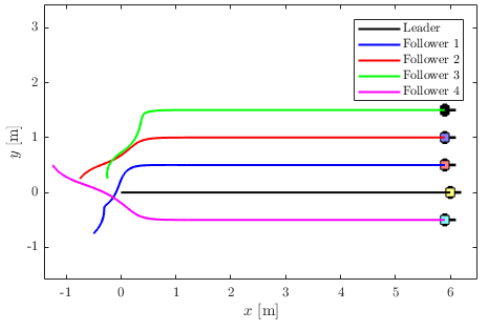
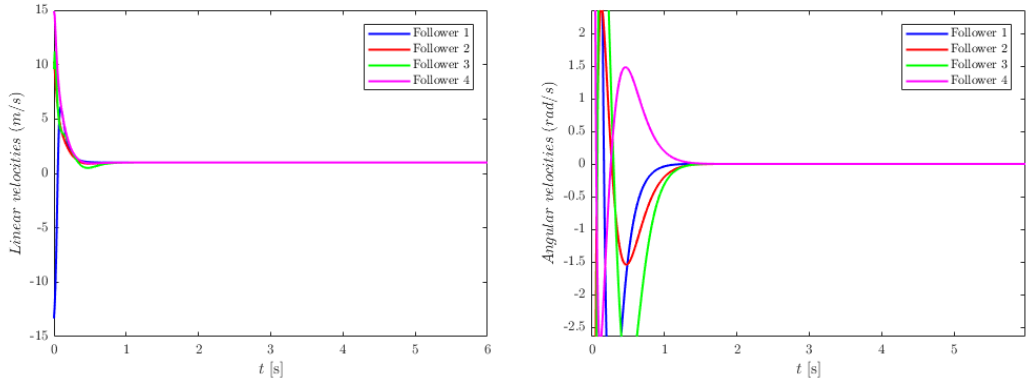
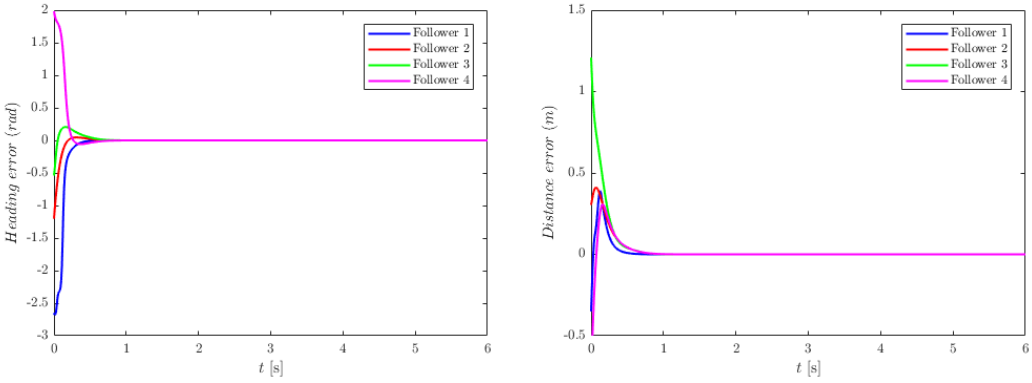


Figure 4.7: Leader and followers paths using Backstepping



(a) (b)

Figure 4.8: Followers velocities using Backstepping:(a) Linear (b) Angular



(a) (b)

Figure 4.9: Followers tracking errors using Backstepping: (a)bearing error (b) Separation error

4.2.3.1 Results:

As shown in Figure (4.7), the followers can follow the leader while maintaining the proper separation and rotation. Figure (4.8) show the followers linear and angular velocities, respectively. Around $t = 1s$, the followers velocities reach constant values. Figure (4.9) which depict the direction and distance errors of the followers, respectively, demonstrate how the followers tracking errors converge to zero within a second.

4.3 2^{nd} trajectory: Circle

Consider a leader and four followers. Suppose the distance of the rear axle to the front of the robot is $c = 0.1m$, the linear velocity v_l and angular velocity ω_l of the leading robot are respectively $[2; 1]^T$ for its trajectory to be circular, and the initial posture of the leading robot is $[0, 0, 0]^T$. In order to maintain the formation of leader-follower, the follower must maintain the desired separation $\sigma_{lf}^d = 0.5m$ and orientation $\varphi_{lf}^d = \frac{4\pi}{3}$ with his leader.

4.3.1 Integral sliding mode control:

Using the same initial conditions mentioned in 4.2.1 with $\lambda = 1$ and $k = 7$.

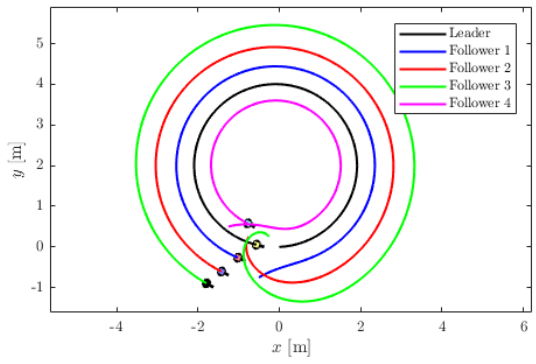
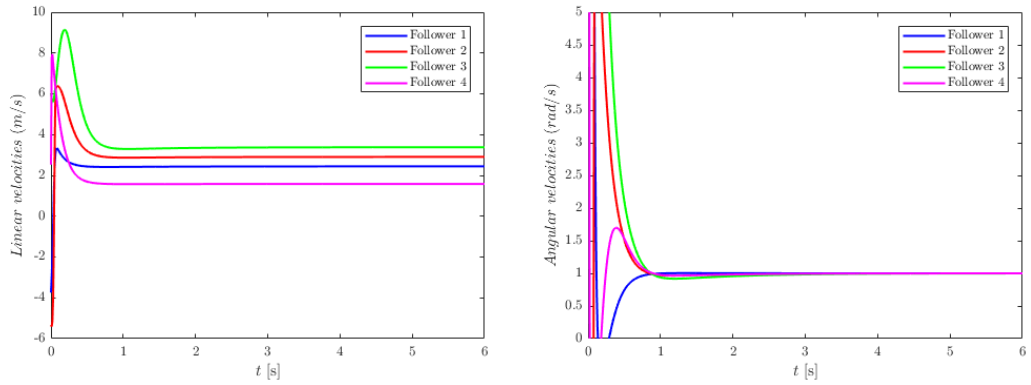
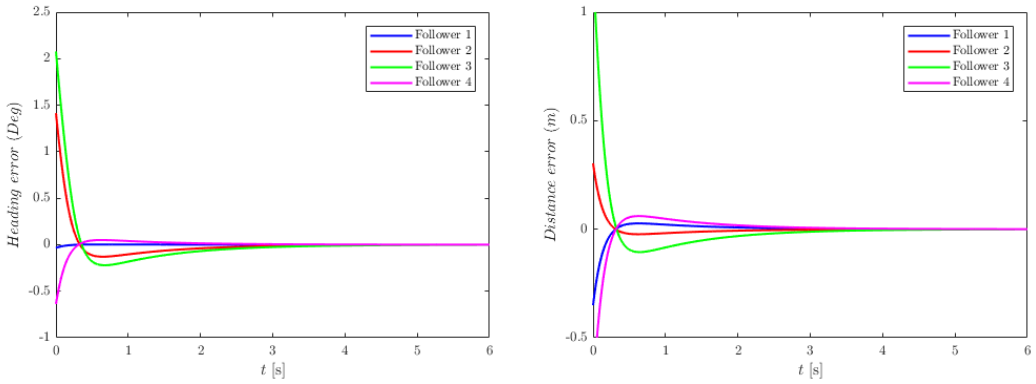


Figure 4.10: Leader and followers paths using ISM



(a) (b)

Figure 4.11: Followers velocities using ISM:(a) Linear (b) Angular



(a) (b)

Figure 4.12: Followers tracking errors using ISM: (a)bearing error (b) Separation error

4.3.1.1 Results:

The leader-follower formation is maintained smoothly as in figure(4.10). The followers's linear and angular velocities starts high and then it converges towards the leader's within 1s (Figure (4.11)). Figure (4.12) represents the followers heading angle and show that followers tracking errors converge to zero after (t = 3s).

4.3.2 PID control:

Using the previous initial conditons mentionde in 4.2.2

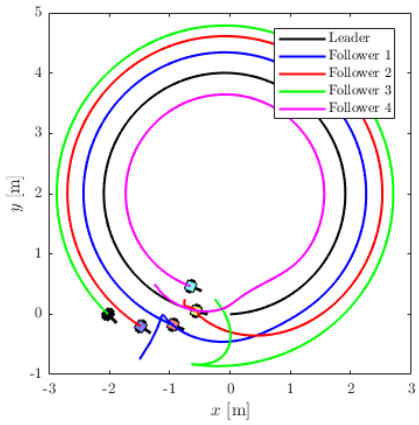
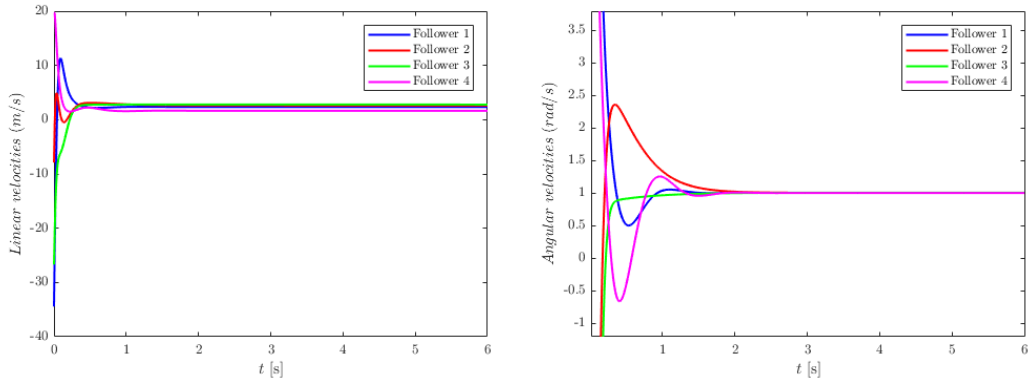
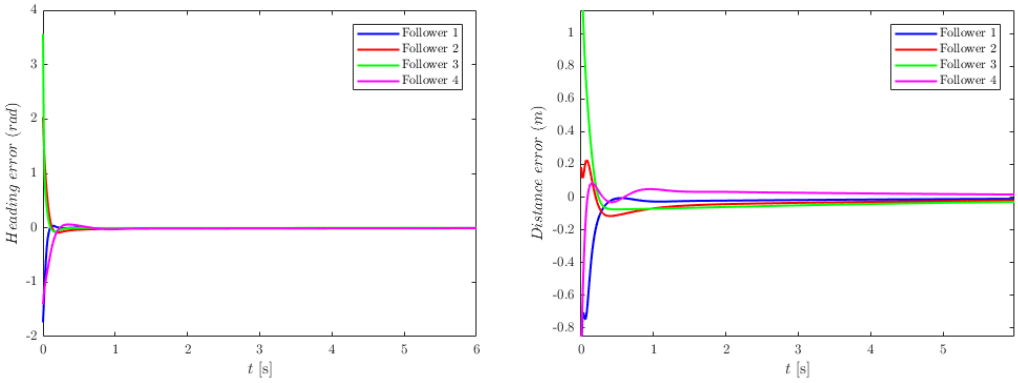


Figure 4.13: Leader and followers paths using PID



(a) (b)

Figure 4.14: Followers velocities using PID:(a) Linear (b) Angular



(a) (b)

Figure 4.15: Followers tracking errors using PID: (a)bearing error (b) Separation error

4.3.2.1 Results:

As seen in Figure (4.13), the followers tail their leader while preserving the correct formation. After (t=1s), the followers' velocities achieve constant values as seen in figure (4.14). Figure (4.15) which depict The direction error of the followers converges to zero in a matter of 0.8s while the distance errors took 6s to be null.

4.3.3 Backstepping:

Using the previous initial conditons from 4.2.3

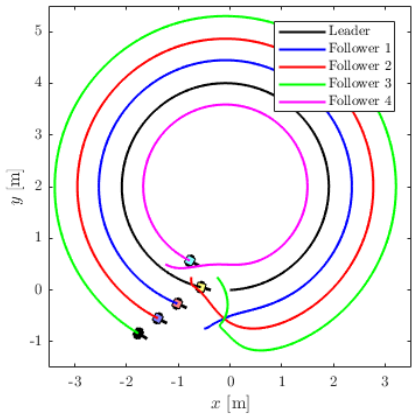
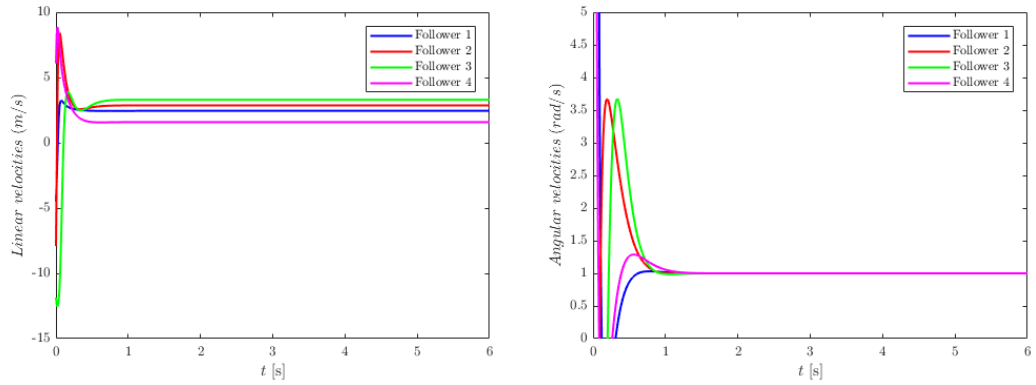
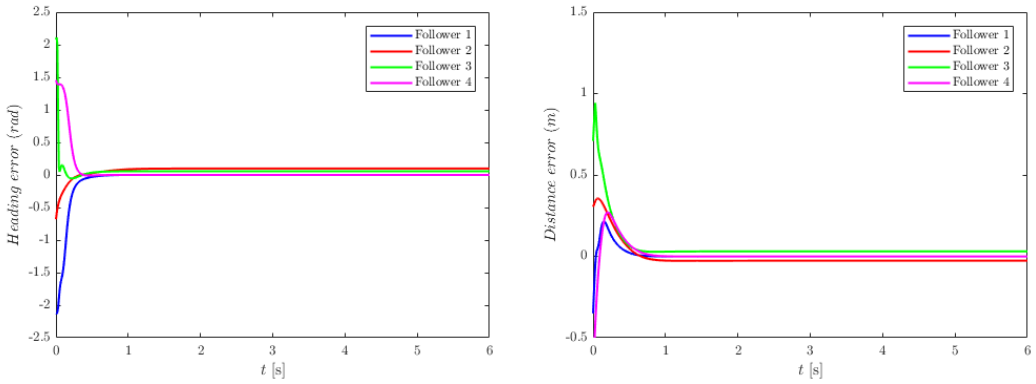


Figure 4.16: Leader and followers paths using Backstepping



(a) (b)

Figure 4.17: Followers velocities using Backstepping:(a) Linear (b) Angular



(a) (b)

Figure 4.18: Followers tracking errors using Backstepping: (a)bearing error (b) Separation error

4.3.3.1 Results:

As shown in Figure (4.16), the followers can follow the leader while maintaining the proper separation and rotation . The linear and angular speeds stabilize at : 0.5s and 1s (figure(4.17)) respectively. The followers tracking errors converge to zero at $t = 0.8s$ as seen in figures (4.18).

4.4 3^{rd} trajectory: Infinity shape

For the reason that the performance of the backstepping controlled followers we implemented a backstepping control to the leader so it can follow a complicated trajectory. Other than the leader's control laws, no changes in the control constants or the initial conditions will be made.

4.4.1 Integral sliding mode control:

Using the same initial conditions mentioned in 4.3.1.

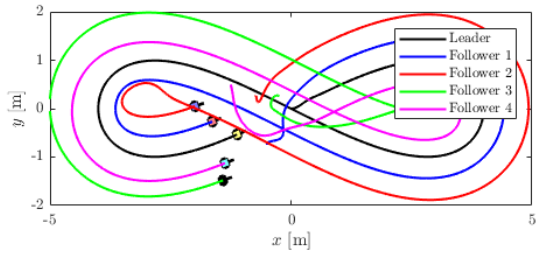
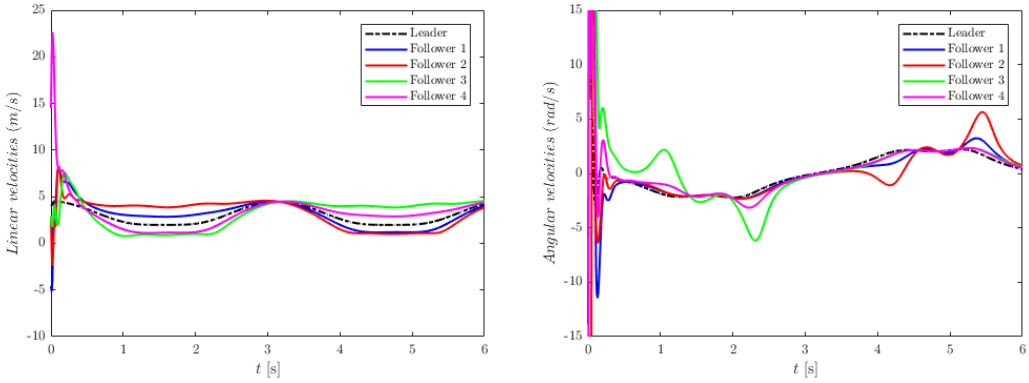
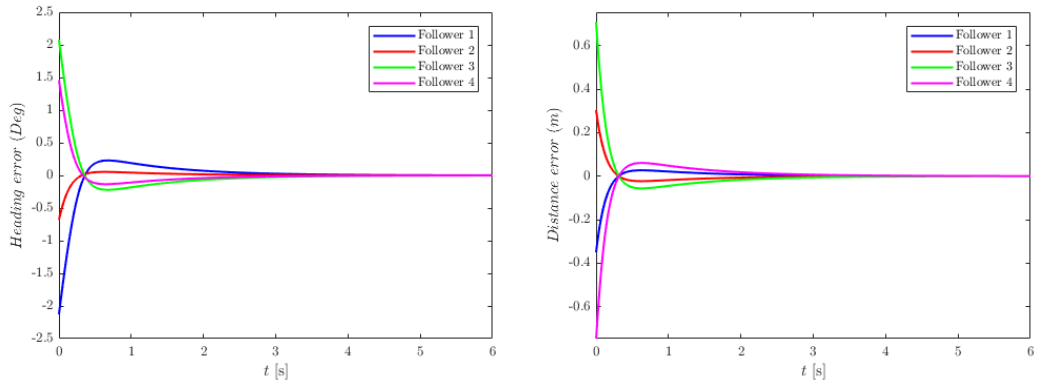


Figure 4.19: Leader and followers paths using ISM



(a) (b)

Figure 4.20: Followers velocities using ISM:(a) Linear (b) Angular



(a) (b)

Figure 4.21: Followers tracking errors using ISM: (a)bearing error (b) Separation error

4.4.1.1 Results:

Figure(4.19) shows that the leader-followers formation is maintained smoothly. The linear and angular velocities joins the leader's (at t= 1s) throughout the path with some disturbance in the angular velocities (figure (4.20)). The tracking errors converge to zero (t = 3s) (figures (4.21)).

4.4.2 PID control:

Using the previous initial conditons mentionde in 4.3.2.

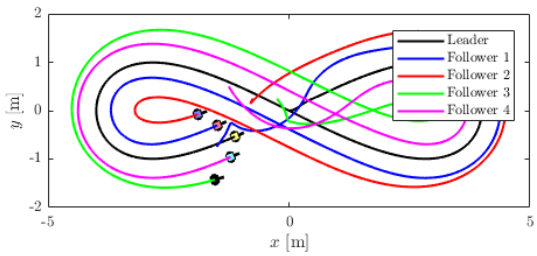
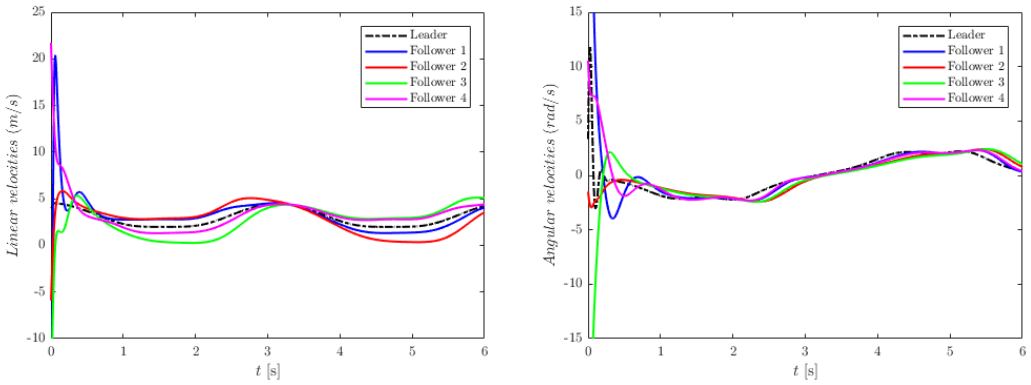
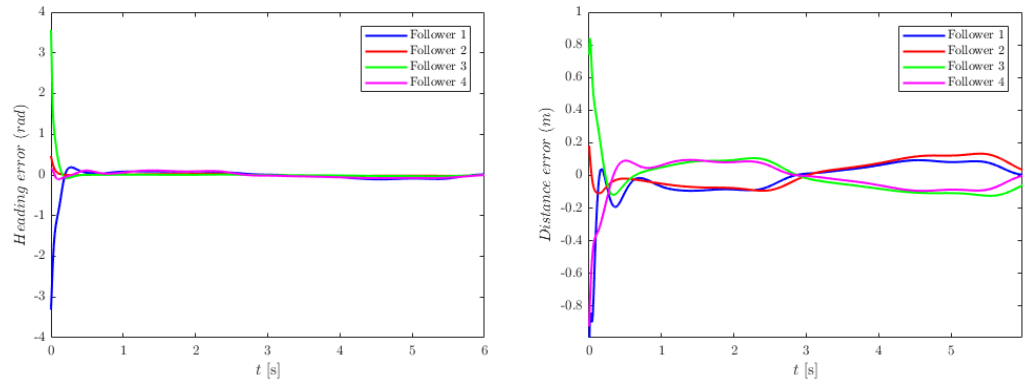


Figure 4.22: Leader and followers paths using PID



(a) (b)

Figure 4.23: Followers velocities using PID:(a) Linear (b) Angular



(a) (b)

Figure 4.24: Followers tracking errors using PID: (a)bearing error (b) Separation error

4.4.2.1 Results:

As seen in Figure (4.22), the followers follow their leader while preserving the correct separation and rotation . Figure (4.23) shows that the followers linear and angular velocities approaches the leader's around (t=1s) with no chattering through the path. The direction error converges to zero at (t= 0.5s) with $\pm 0.05m$ and the distance errors converges with $\pm 20\%$ error as shown in figure(4.24).

4.4.3 Backstepping:

Using the previous initial conditons from 4.3.3.

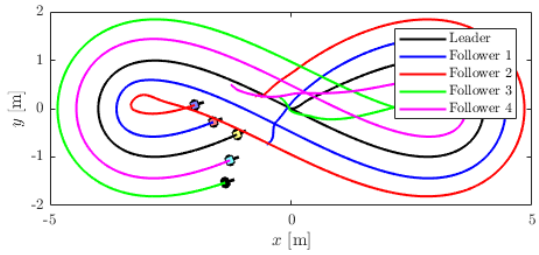
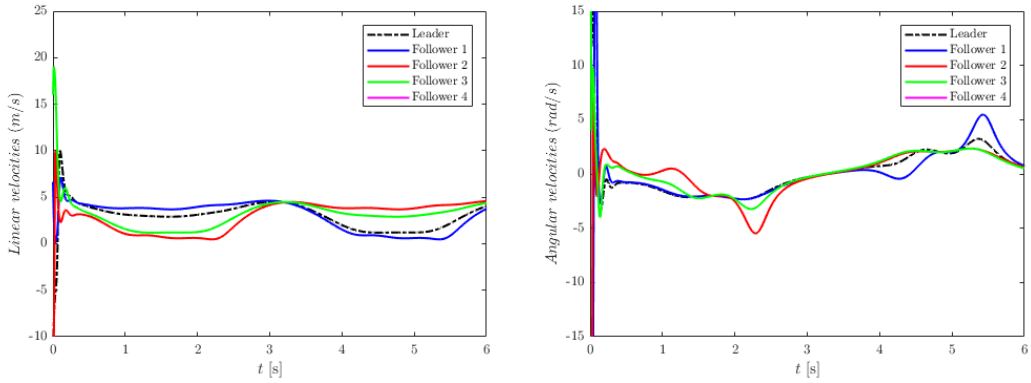
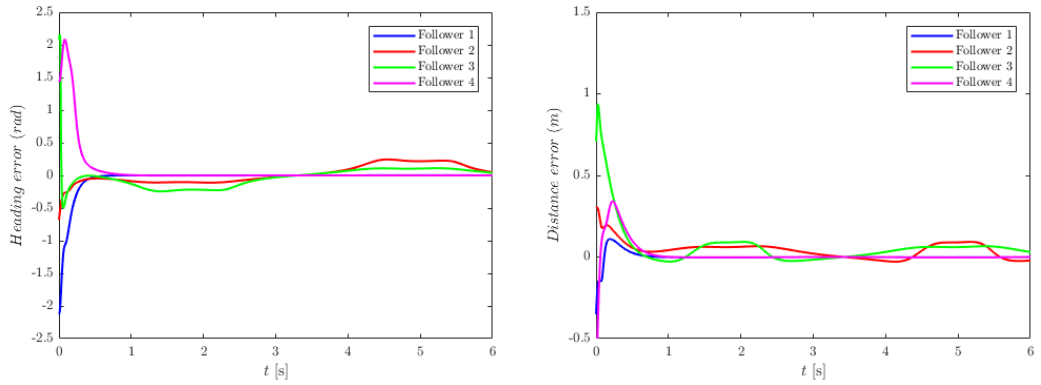


Figure 4.25: Leader and followers paths using Backstepping



(a) (b)

Figure 4.26: Followers velocities using Backstepping:(a) Linear (b) Angular



(a) (b)

Figure 4.27: Followers tracking errors using Backstepping: (a)bearing error (b) Separation error

4.4.3.1 Results

The leader-follower formation is maintained smoothly as in figure(4.25). The followers's linear and angular velocities starts high and then it converges towards the leader's under 0.3s with some chattering in angular velocity: Figure (4.26). Figure(4.27): The tracking errors converges to zero at (t= 1s) with $\pm 0.05 to 0.1m$ in distance error.

4.5 Analysis

Consider a leader and three followers. Suppose the distance of the rear axle to the front of the robot is $c = 0.1m$, the initial posture of the leading robot is $[0, 0, 0]^T$ and for all followers it's $[-1.25, 1.5, 0]^T$. In order to maintain the formation of leader-follower, the follower must maintain the desired separation $\sigma_{lf}^d = 1.5m$ and orientation $\varphi_{lf}^d = \frac{2\pi}{3}$ with his leader.

ISM constants are: $\lambda = 1$ and $k = 7$. PID constants are: $k_p = 2$, $k_i = 4$ and $k_d = 0.2$. Backstepping constants: $k_1 = 10$, $k_2 = 2$ and $k_3 = 1$.

4.5.1 Leader's velocities: fixed

Consideing $u_l = (2, 1)^T$, the robots paths would be as shown in Fig(4.28):

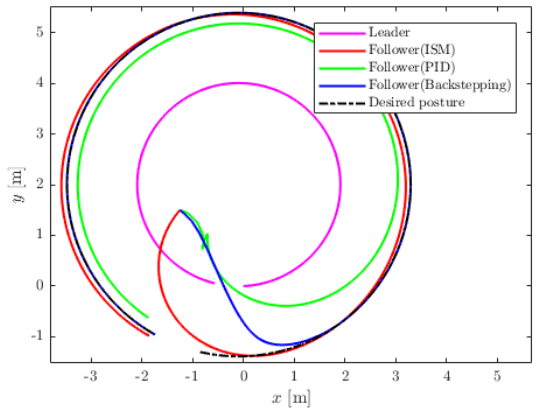
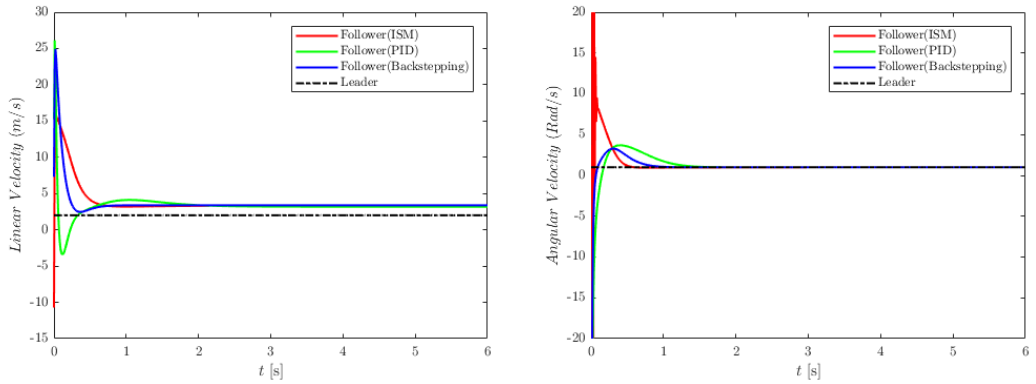


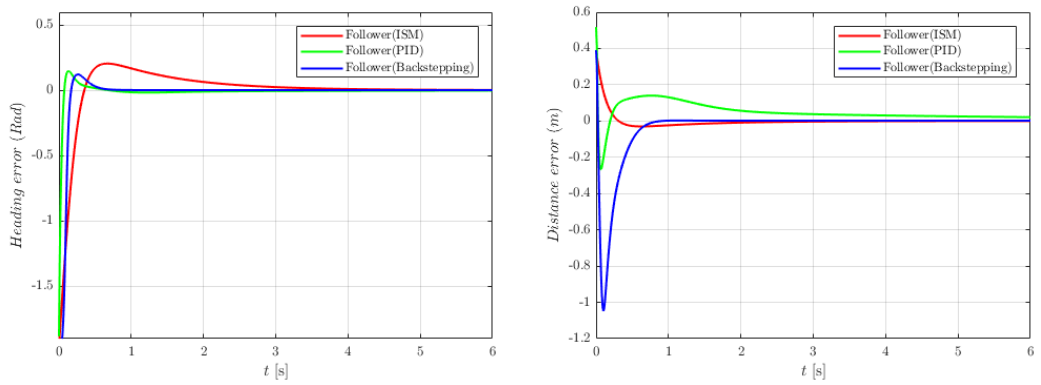
Figure 4.28: Leader and followers: circular trajectory



(a) (b)

Figure 4.29: Followers velocities:(a) Linear (b) Angular

Since the leader's velocities are constants each follower's control laws goal is to match its velocities (the leader's). We can see from figure(4.29) that the follower with: ISM, BS, PID will match the leader's linear velocity respectively at $0.8s$,$0.8s$,$1.9s$ and its (the leader's) angular velocity at $0.5s$,$0.95s$,$1.4s$. Some chattering in ISM angular velocity is noticed.



(a) (b)

Figure 4.30: Followers tracking errors: (a)bearing error (b) Separation error

The time needed for tracking errors to converge to zero with a tolerance of ±0.66% can be extracted from figure(4.30), we'll represent it in the table below:

	ISM	PID	BS
$d_e(s)$	1.89	-	0.77
$h_e(s)$	4	2.53	0.65

Table 4.1: Error convergence time

The PID distance error does not converge to zero. The BS has the fastest response.

For the purpose of judging the performance of each control system, we'll calculate the performance indices for both distance between the follower and its leader and the heading angle. The data is shown in the tables below:

Control	ISE	ITSE	IAE	ITAE
ISM	0.0094	0.0012	0.0083	0.0619
PID	0.0337	0.0356	0.3498	0.6733
BS	0.1488	0.0226	0.2499	0.0818

Table 4.2: Performance indices: Separation

Control	ISE	ITSE	IAE	ITAE
ISM	0.3612	0.0582	0.5494	0.4377
PID	0.0549	0.0202	0.1198	0.1214
BS	0.3179	0.0145	0.2224	0.0212

Table 4.3: Performance indices: Bearing

Since the goal of this study is to keep the follower at a desired path by keeping the error null. The better performing control has the smallest indices values. From tables (4.2) and (4.3), it's obvious that the outstanding control is ISM when it comes to keeping the desired separation while it's the PID for keeping the desired heading angle.

4.5.2 Leader's velocities: variable

Choosing the desired separation $\sigma_{lf}^{d} = 1.5m$ and orientation $\varphi_{lf}^{d} = \frac{3\pi}{4}$. The paths travelled by the robots is presented in figure(4.31):

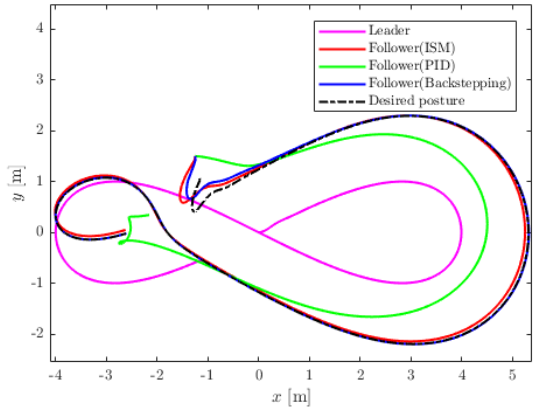
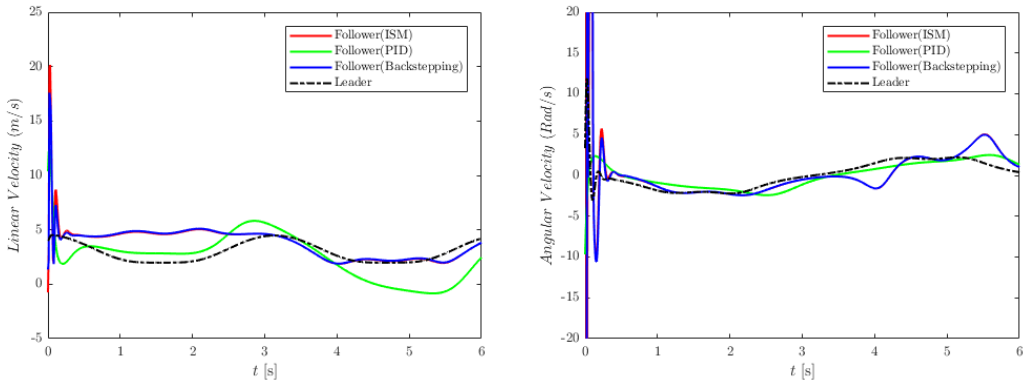


Figure 4.31: Leader and followers: S trajectory

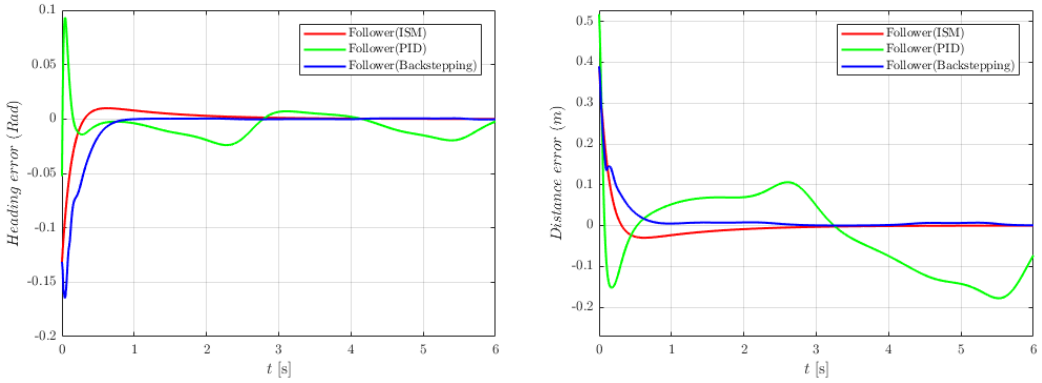
Figure(4.32) describes the followers speeds, we can see chattering in ISM and BS at $t = [0, 0.1s]$ and some at $t = 3.5s$ and $t = 5.5s$ otherwise their plots are corresponding.



(a) (b)

Figure 4.32: Followers velocities:(a) Linear (b) Angular

Figure(4.33) represents the tracking errors. The BS converges in less than 1s, ISM converges within 2s while the PID does not converge.



(a) (b)

Figure 4.33: Followers tracking errors: (a)bearing error (b) Separation error

The performance indices shown in tables (4.4) and (4.5):

Control	ISE	ITSE	IAE	ITAE
ISM	0.0094	0.0012	0.0803	0.0619
PID	0.0630	0.2257	0.5276	1.8237
BS	0.0102	0.0013	0.0839	0.0786

Table 4.4: Performance indices: Separation

Control	ISE	ITSE	IAE	ITAE
ISM	0.001	0.0001	0.0266	0.0205
PID	0.0013	0.0026	0.0651	0.1840
BS	0.0035	0.0004	0.0392	0.0128

Table 4.5: Heading errors

In regards to the separation the finest control is ISM, the BS performance is acceptable. While PID and BS perforemed remarquably when it came to the heading angle.

4.6 Conclusion

In terms of rapidity(response time), BS is the fastest then PID and the slowest is ISM. Although ISM is indistinguishably accurate while the PID is not certain. In regards to stability, ISM begins with very high velocities compared to the PID which is the most stable and BS shows instability but it can be reduced.

General Conclusion

This thesis presents a multi-robot control framework using the leader-follower approach. The main objective of this dissertation is to study the performance of a multi robot system using linear and nonlinear control.

In this dissertation, we have defined the multi-robot systems and cooperative control. Then we explained the modelling of the leader-follower control approach, which is the main subject of the dissertation. We gave a brief definition of the control approaches: Integral sliding mode, PID and Backstepping. We developed an algorithm for each of them and implemented it in Matlab. The simulation results on MatLab demonstrated the efficiency and stability of the robots and the tracking algorithm using the leader-follower approach, which also proved their correct functioning. The advantages and disadvantages of each method had surfaced. We compared them in terms of rapidity,accuracy, and stability.

The choice of the controller is based on the system's constraints. For example, if the system does not have to be accurate to the millimeter and the chattering might damage it which is the case of many industrial systems, the PID can be used.

As an extension of this thesis, disturbances, uncertanties, and obstacle avoidance and collision avoidance will be considered as future work.

Bibliography

Bibliography

[1] R. M. Voyles and M. K. O'Malley: Robotic Systems and Intelligent Machines. Wiley,Encyclopedia of Electrical and Electronics Engineering, 2015.

[2] A. Agogino: Formations of multi-robot teams. In: Robotics Research, Springer, Berlin, Heidelberg, 2000.

[3] b

[4] Bloch, Am and Brogliato, B: Nonholonomic Mechanics and Control. Applied Mechanics Reviews (2004). 57. 3-. 10.1115/1.1641775.

[5] N. Naidoo: A distributed framework for the control and cooperation of heterogeneous mobile robots in smart factories. Kwazulu-Natal University, 2017.

[6] A. Gil, J. Aguilar, E. Dapena, and R. Rivas: Emotional model for a multi-robot system with emergent behavior. IAES International Journal of Robotics and Automation (IJRA), vol.9, no. 3, pp. 220–232, Sep. 2020, doi: 10.11591/ijra.v9i3.pp220- 232.

[7] G. I., N. G., and S. M: Path planning and collision avoidance regime for a multi-agent system in industrial robotics. International Scientific Journals of Scientific Technical Union of Mechanical Engineering "Industry 4.0," vol. 11, no. 11, pp. 519–522, 2017.

[8] S. G. Tzafestas: Mobile robot control and navigation: a global overview. Journal of Intelligent and Robotic Systems, vol. 91, no. 1, pp. 35–58, Jul. 2018, doi: 10.1007/s10846-018-0805-9.

[9]] J. Cortes and M. Egerstedt: Coordinated control of multi-robot systems: a survey. SICE Journal of Control, Measurement, and System Integration, vol. 10, no. 6, pp. 495–503, Nov.2017, doi: 10.9746/jcmsi.10.495.

[10] Y.-C. Wu and H.-C. Wang: Laboratory environment monitoring and specimen transport robots. IAES International Journal of Robotics and Automation (IJRA), vol. 8, no. 4, pp.313–326, Dec. 2019, doi: 10.11591/ijra.v8i4.pp313-326.

[11] P. Megantoro, H. Setiadi, and B. A. Pramudita, "All-terrain mobile robot desinfectant sprayer to decrease the spread of COVID19 in open area," International Journal of Electrical and Computer Engineering (IJECE), vol. 11, no. 3, pp. 2090–2100, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2090-2100.

[12] D. De Martini, A. Bonandin, and T. Facchinetti: eduMorse: an open-source framework for mobile robotics education. In International Conference on Robotics and Education RiE, 2018, pp. 289–300.

[13] T. Samad, S. Iqbal, A. W. Malik, O. Arif, and P. Bloodsworth: A multi-agent framework for cloud-based management of collaborative robots. International Journal of Advanced Robotic Systems, vol. 15, no. 4, Jul. 2018, doi: 10.1177/1729881418785073.

[14] M. Badawy, H. Khalifa, and H. Arafat: New approach to enhancing the performance of cloud-based vision system of mobile robots. Computers and Electrical Engineering, vol. 74, pp. 1–21, Mar. 2019, doi: 10.1016/j.compeleceng.2019.01.001.

[15]] W. Tanaka Botelho, M. das Gracas Bruno Marietto, E. de Lima Mendes, J. Carlos da Motta Ferreira, and V. Lucia da Silva: Multi-robot system for tracking and surrounding a stationary target: A decentralized and cooperative approach. In

2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), Dec. 2017, pp. 793–798, doi: 10.1109/ROBIO.2017.8324514.

[16] E. Castelló Ferrer: The blockchain: a new framework for robotic swarm systems. In Advances in Intelligent Systems and Computing, 2019, pp. 1037–1058.

[17] O. Bayasli and H. Salhi: The cubic root unscented kalman filter to estimate the position and orientation of mobile robot trajectory. International Journal of Electrical and Computer Engineering (IJECE), vol. 10, no. 5, pp. 5243–5250, Oct. 2020, doi: 10.11591/ijece.v10i5.pp5243-5250.

[18] G. Farias, E. Fabregas, E. Peralta, H. Vargas, S. Dormido-Canto, and S. Dormido: Development of an easy-to-Use multi-agent platform for teaching mobile robotics. IEEE Access, vol. 7, pp. 55885–55897, 2019, doi: 10.1109/ACCESS.2019.2913916.

[19] P. G. K. Elgezainy and A. Rizk: Review on the Applications of Robotics in Precision Agriculture. In 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), 2019.

[20] Fierro, R., Song, P., Das, A., Kumar, V. (n.d.): Cooperative Control of Robot Formations. Cooperative Control and Optimization, 73–93. doi:10.1007/0-306-47536-75.

[21] Farbod Fahimi.Autonomous Robots : Modeling, Path Planning, and Control. Springer, 2009.

[22] Kevin M. Lynch and Frank C. Park. Modern Robotics : Mechanics, Planning, and Control. Cambridge University Press,2017.

[23] Kagan, Eugene. Autonomous Mobile Robots and multi robot systems : Motion, Plannig, Communication, and Swarming. Wiley,2020.

[24] XiaoQi Chen, Y.Q.Chen and J.G.Chase. Mobile Robots: State of the art in land, sea, air and collaborative missions.

[25] R. Fierro, F. Lewis, Control of a nonholonomic mobile robot using neural networks, IEEE Transactions on Neural Networks 9 (1998) 589–600.

[26] Habibi SR. The smooth variable structure filter. In : IEEE Conference. Vol. 95. 2007. p. 1026-1059.

[27] A. Handayani, S. Nurmaini, N.Husni, and I.Yanithe : Survey paper: formation control for swarm robots. Proceedings of the International Multi Conference of Engineers and Computer Scientists 2016 Vol I, IMECS 2016, March 16 - 18, 2016, Hong Kong.

[28] S.V. Emelyanov, On pecularities of variable structure control systems with discontinuous switching functions, Doklady ANSSR, Vol. 153, pp. 776-778, 1963.

[29] C.Edwards and S.K.Spurgeon, Sliding mode control : Theory and Application, Taylor and Francis, 1998.

[30] V.I.Utkin , Sliding modes in Control Optimisation, Springer-Verlag, Berlin, 1992.

[31] U.Itkis, Control systems of variable structure, Wiley, New York, 1976.

[32] V.I.Utkin, Variable structure systems with sliding modes, IEEE Transactions Automatic Control, 22, 1977, pp.212-222.

[33] B.Bandyopadhyay and S.Janardhnan, Discrete-time Sliding Mode Control: A Multirate Output Feedback Approach, Springer-Verlag Berlin Heidelberg, 2006.

[34] Susy Thomas and B. Bandyopadhyay. Comments on 'a new controller design for one link manipulator'. IEEE Trans. on Auto. Contr., 42:425–429, Mar. 1997.

[35] R. A. DeCarlo, S. H. Zak, and S. V. Drakunov. Variable structure, sliding-mode controller design. In William S. Levine, editor, The Control Handbook, chapter Design Methods, pages 941–951. CRC Press, Boca Raton, Florida, USA, 1996.

[36] J. Y. Hung, Weibing Gao, and J. C. Hung. Variable structure control : A survey. IEEE Trans. on Ind. Electron., 40(1):2–21, Feb 1993.

[37] B. Heck, "Sliding mode control for singularly perturbed systems", Int.J. Control, Vol. 53, pp. 985-1001, 1991.

[38] J.J.E. Slotine, "Sliding controller design for nonlinear systems", Int.J. Control, Vol. 40, No. 2, pp. 421-434, 1984.

[39] M.Asif, M.J.Khan and A.Y.Memon, Integral sliding mode formation control of nonholonomic robots using leader- follower approach. Robotica, Cambridge University Press, 2016.

[40] I.S.Choi and J.S.Choi, Leader-Follower Formation Control Using PID Controller. Intelligent Robotics and Applications,5th International Conference, ICIRA 2012 Montreal, QC, Canada, October 3-5, 2012.

[41] Krstic, M., Kanellakopoulos, I., Kokotovic, P.V.: Nonlinear and Adaptive Control Design. Wiley, New York (1995)

[42] Miroslav Krstic and Andrey Smyshlyaev: Boundary Control of PDEs A Course on Backstepping Designs. SIAM,2008.

[43] Shubhobrata Rudra, Ranjit Kumar Barai and Madhubanti Maitra: Block Backstepping Design of Nonlinear State Feedback Control Law for Underactuated Mechanical Systems. Springer Science+Business Media Singapore, 2017.

[44] Z.Peng, G.Wen, A.Rahmani and Y.Yu: Leader–follower formation control of nonholonomic mobile robots based on a bioinspired neurodynamic based approach. Robotics and Autonomous Systems, Volume 61, Issue: 9, September 2013, p988-996.

[45] Consolini, L., Morbidi, F., Prattichizzo, D., Tosques, M.: Leader-Follower Formation Control of nonholonimic mobile robots with input constraints. Automatica 44(5), 1343– 1349 (2008)