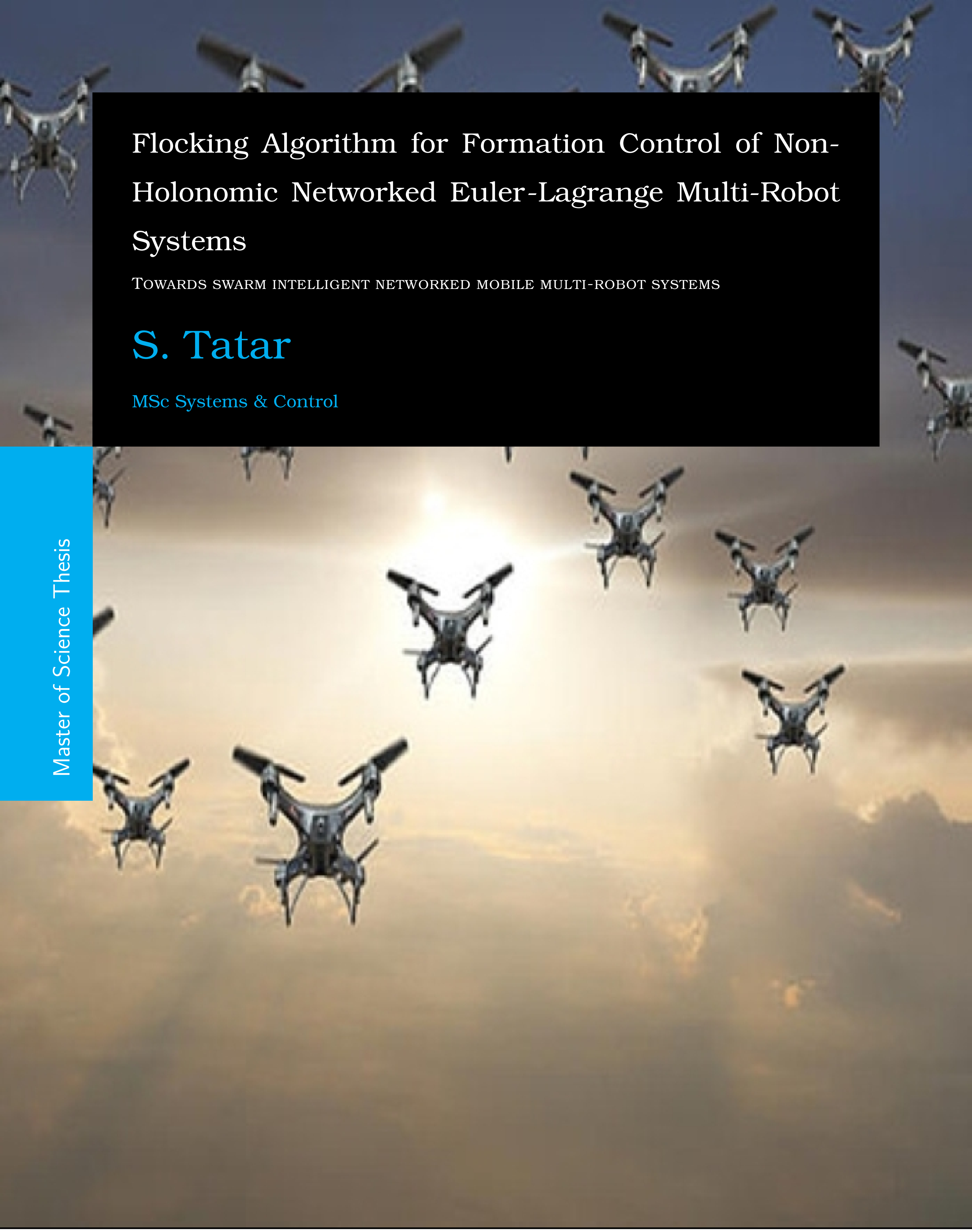


Flocking Algorithm for Formation Control of Non-Holonomic Networked Euler-Lagrange Multi-Robot Systems

Towards swarm intelligent networked mobile multi-robot systems

S. Tatar

MSc Systems & Control

Master of Science Thesis

Delft Center for Systems and Control

Flocking Algorithm for Formation Control of Non-Holonomic Networked Euler-Lagrange Multi-Robot Systems

Towards swarm intelligent networked mobile multi-robot systems

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

S. Tatar
MSc Systems & Control

August 30, 2020

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Abstract

Recent activities in the research on swarm robotics have emerged from the application of concepts from swarm intelligence into multi-robot systems (MRSs) that model the realistic interaction between robots in the system and the environment. Fundamentally, the literature on swarm robotics is biologically inspired by systems as insect colonies, flocks of birds, schools of fish and bacteria colonies. Recently, the flocking formation control behaviour in multi-agent systems (MASs) has encouraged astounding attention among the researchers. Researchers from various disciplines including physics, biophysics, computer science and control engineering have been fascinated by the emergence of flocking, swarming and schooling in MASs under local interactions. In this research, we focus on flocking algorithms for MRSs. The flocking phenomenon is characterized as a form of collective behaviour of a swarm of robots with a distributed architecture that involves locality of the computation, sensing, communication and effector capabilities. Flocking algorithms have the potential to introduce self-healing, self-organizing and self-configuring capabilities in the functioning of distributed MRSs.

However, despite an exhaustive list concerning flocking formation control algorithms is given in the literature, most of the existing results deal with simple mathematical modelled robots. In practice, mobile robots embrace more complex nonlinear dynamic mathematical models and involve non-holonomic constraints. Therefore, it is of scientific and practical interest to study the effectiveness of the flocking algorithms for such complex nonlinear systems involving non-holonomic constraints.

This thesis study, expanding novel features on the existing literature, presents a connectivity-preserving artificial potential function (APF)-based flocking algorithm for formation control of mobile networked non-holonomic Euler-Lagrange (EL) dynamical agents under a proximity graph interaction architecture involving a limited sensing radius. In specific, we consider three algorithms: (i) flocking; (ii) (virtual) leader-following flocking; (iii) flocking with obstacle avoidance. Proximity graphs are viewed as a useful and decent mathematical tool to incorporate the practical time-varying communication topology of MRSs in flocking algorithms. The preservation of the network connectivity is of significant importance for the flock stability and synchronization (i.e. consensus) since they firmly depend on it. The use of APF, to encode the local interaction rules for achieving global performance, is inspired by the observations and models of the biologists. APF-based flocking control algorithms are mainly interesting as they are not limited to higher-level models and can be exploited for more advanced nonlinear dynamic models and control strategies for flocking and collision avoidance purposes.

The aforementioned algorithm setting improves the practical relevance of the problems to be addressed in this study and meanwhile, it poses technical challenges to the design of the flocking control algorithm and theoretical stability proof, respectively. In all proposed algorithms in this study, being the first author in the literature to study flocking algorithms for non-holonomic EL systems in specific, novel theoretical results for this class of systems, exploiting nonlinear control theory concepts where a nonnegative lower bounded ”energy-like” Lyapunov function candidate is defined, are obtained. Advanced numerical simulation studies and some performance metrics are presented as a complement to the analytical framework in order to verify the effectiveness of the theoretical results.

Contents

List of Figures

List of Tables

Preface

In the initial phase of my thesis study, I was convinced that it would be much like the previous courses in my study, but with writing a larger report in the end. However, this was a misunderstanding. The thesis work differs from the rest of your study in that you are expected to already be able to perform research at the level of an engineer with an MSc degree. A year of pure hard work on your MSc thesis teaches you a lot of new skills, such as understanding scientific papers, profound understanding of the theory behind the research topic and discipline.

Also, I was motivated and encouraged by my supervisor to decide myself which research topic got my interest the most. This was highly appreciated since it gives you the opportunity to constitute your own thesis topic. This was not an easy task, however, in the end, it teaches you to think like a critical and dignified engineer who is able to find limitations in the concerned research field and strives for innovation.

The most valuable thing that I have learnt during my research is to set up a proper research methodology. It is very difficult to predict and to conclude on the basis of a theoretical analysis what the cooperative behavior of the networked system under the flocking protocol will look like. Also, it is not sufficient to quickly write a simulation that proves your hypothesis. However, to initialize your research results, a theoretical analysis is the proper and least expensive way of studying the flocking characteristics of the control system.

My main message concerning this, though simulations are very important in nonlinear control, in order to perform an appropriate numerical simulation study, it has to be guided by theoretical results to prevent blind simulation and misleading results. This is especially fundamental in the case of nonlinear control systems as they exhibit great richness in their behavior, depending on its initial conditions and control inputs.

I would like to thank my supervisor dr.ir. E. Steur for his guidance during the process and for his assistance in teaching me the required skills that are needed to become a respectable engineer. His door was always open for questions and he was always interested in my latest results or encountered problems. This really lifted my research to a higher level.

Lastly, I would also like to express my gratefulness to my family and friends for supporting me in both good and difficult times to help me along my growth.

Delft, University of Technology
MSc Systems & Control
August 30, 2020

S. Tatar

“The key with autonomous is the whole ecosystem. One of the keys to having a truly fully autonomous is vehicles talking to each other.”

— *Mary Barra, General Motors (GM) CEO*

Chapter 1

Introduction

The focus of this thesis study is to design a novel flocking algorithm for formation control of networked non-holonomic multi-robot systems (MRSs) to obtain innovative scientific and practical insights to the distributed control of such complex systems. As will be evident later on, non-holonomic systems are exceptionally interesting since they appear in many advanced robotic structures and involve under-actuated systems, which are of fundamental practical relevance.

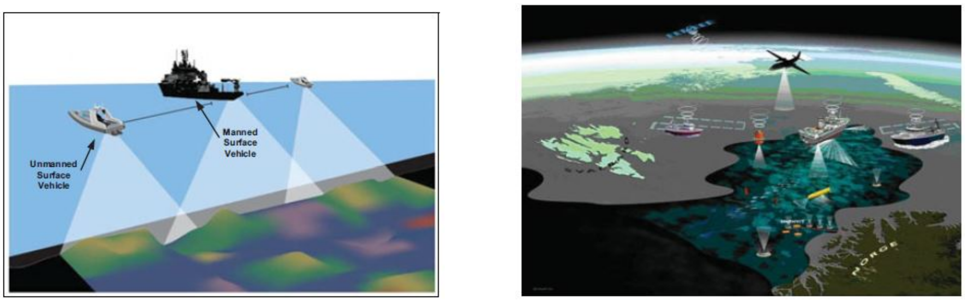
Initially, the background and motivation for studying formation control of MRSs is discussed in perspective of swarm intelligence in robotics with specific focus on flocking algorithms. Then, the research gaps are identified by discussing the motivation of this study and the problem is formulated. In addition, the main contributions and challenges of this study are given. Finally, an outline of this study is provided to discuss the remainder of the chapters in this study.

1-1 Background & Motivation

Recent technological advances offer more efficient computation and less expensive communication. The technological revolution that came along in the last century with the arrival of wireless communication brought a span of innovation[37]. As a consequence, interacting systems require no longer the constraint to have a physical interconnection. The ability to locally compute and share information has facilitated the development of multi-agent systems (MASs). The development of MASs, compared to an individual and complex agent, promises increased performance, efficiency and robustness, utilizing distributed local sensing and actuation[58;57].

In some practical applications, a single complex system can be replaced by interacting MASs with simpler structure. In fact, a group of robots, e.g. Unmanned Ground Vehicles (UGVs), Unmanned Aerial Vehicles (UAVs), Unmanned Surface Vehicles (USVs) or Autonomous Underwater Vehicles (AUVs), with simple structure, is able to perform more complex tasks with reduced operational costs compared to a single complex robot[33]. The reason for this is the adaptability and flexibility of the multi-robot system. Coordinated mobile robot group motion has its applications in the sea, on land, in the air, the stratosphere and in space[29].

Formation control of MRSs has been one of the most fundamental problems for decades in cooperative control of a group of robots which aims at forcing robots to converge to, and maintain, a specific geometric pattern.



(a) (b)

Figure 1-1: The formation deployment of USVs with (a) the sea mapping operation achieved by formation control of both USVs and manned vehicles and (b) cooperative formation deployment between USVs, AUVs and UAVs[33].

In fact, the importance and benefits of increasing the use of unmanned vehicles (UVs) formations in future battlefield has even been addressed by the U.S. Department of Defense (DoD) by investing the defense budget in the study and development of unmanned vehicle robot systems[33].
An interesting potential future application of formation control might be the cooperation of both manned and unmanned vehicles in formation where the manned vehicle serves as a guidance for the USVs in performing the mission. In fact, due to the nature of surface operations of USVs, more important roles are going to be played in large-scale-cross platform cooperation acting across different UVs[33]. In addition, another promising potential application might be the formation deployment of cooperating USVs with other unmanned vehicles to achieve a network of unmanned systems.
By using the ability of the USVs regarding communication with both aerial and vehicles under water, it can uniquely serve as an interchange station for distribution of the obtained real-time information among the other vehicles to enhance the efficiency of the communication. This formation control concept is illustrated in Figure 1-1.
Then, the motivation behind the tremendous attention that attracted the researchers to perform studies on nature-inspired formation control algorithms has its roots in several fields. Biologists who study animal aggregations such as swarms, flocks, schools and herds have observed remarkable group-level characteristics that are exhibited as emergent properties from individual-level behaviors[29]. This includes the ability of quick and efficient coordinated maneuvers, the ability to quickly process data and a significantly improved and advanced decision making ability[29].
Swarm robotics (SR) research has recently emerged from the application of swarm intelligence concepts into MRSs, that focuses on the physical embodiment and realistic interactions among the individuals themselves and also between the individuals and the environment[2]. It involves a large group of locally interacting agents where the emergence of global behavior can be observed. The intelligence of swarms refers to the superior capabilities of a swarm of MRSs, that e.g. perform the flocking phenomenon, compared to individual robots. Typically, the swarm robotics research area has the following related characteristics[4]: individual autonomous robots with limited capabilities (e.g. in terms of local sensing, communication and computation) to ensure distributed algorithms and do not rely on coordination mechanisms that hinder scalability; large numbers of identical individuals rather than centrally planned heterogeneous teams with predefined roles. It is important to note that the locality of the interactions and communication, which may be considered as a limitation, has beneficial effects on the robustness and scalability of MRSs. Therefore, this interaction and communication structure is generally preferred above global communication and sensing. Summarizing, the main inspirational sources that attracted the attention of the researchers can be listed as follows[24]:

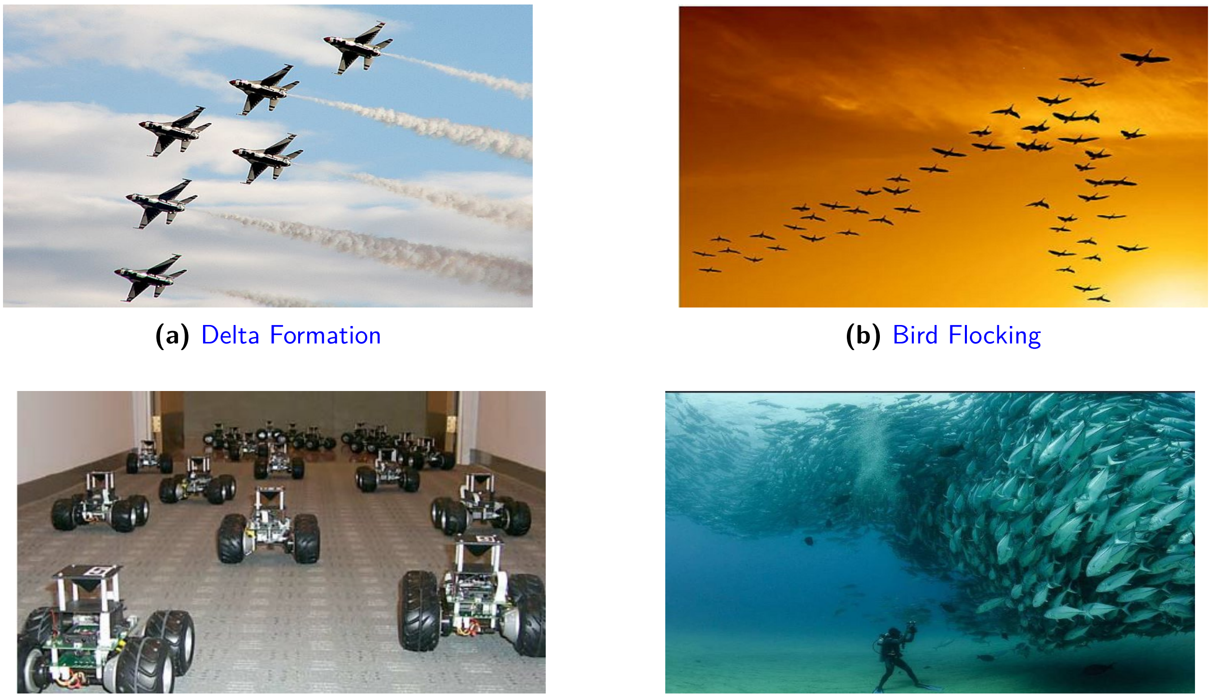
- *Biological inspiration:* remarkable collective group behavior characteristics are observed in biological systems that emerge from simple individual-level local interactions, including efficient formation maneuvering, advanced decision making ability and intelligence. It is essential to understand those characteristics for the implementation of swarm intelligence concepts into formation control of MRSs.

- *Challenging control problems:* designing decentralized control algorithms for distributed systems involves control challenges that are not present in single robotic systems or centralized systems. These challenges include complex interactions, high system dimensionality, incomplete information and uncertainties. This motivates studying the swarm robotics concepts in perspective of the theoretical framework which can have significant effects on the development and understanding of nature-inspired formation control algorithms.

- *Demand for multi-robot systems:* a single robot may not be able to physically execute some task which can be achieved by employing MRSs. Moreover, MRSs can achieve same tasks while reducing the operational time and increasing the efficiency and performance.

According to[17], an agent is any living or non-living, virtual or physical computational quantity, which demonstrates autonomous behaviour and the properties reactivity, pro-activity and the social ability to communicate with other agents. Some examples of agents include living organisms, robots, autonomous vehicles and sensors. In a formal context, a multi-agent system can be defined as a collective group of agents that comprise the following elements:

- interaction through communication
- act/react in an environment
- different "spheres of influence"
- linked by organizational relationships



(a) Delta Formation **(b)** Bird Flocking

(c) Swarm Robotics **(d)** Fish Schooling

Figure 1-2: Illustration of flocking and schooling behavior in nature (b,d) and formation control in engineering (a,c).

In fact, the flocking problem can be viewed as a nature-inspired swarm robotics (SR) concept of the formation control methodology, requiring robots to move together along some aggregate path, however, with minimal requirements for paths taken by particular robots[45]. In general, the formation control algorithms have more restrictive requirements on formation maintenance as the multi-robot system maneuvers in the environment, whereas the flocking algorithms support flexibility, self-organizing and self-reconfigurable features in the system. It is important to note that the flocking phenomenon involves locality of computation, effector, sensing and communication capabilities. The fundamental objective in flocking algorithms for formation control is to design local control laws to achieve the desired emergent collective behavior.

Flocking is a form of collective behavior of a large number of local interacting agents with a common group objective[41]. This emerging collective behavior of MASs is biologically inspired from animal behaviour in e.g. birds, fishes, ants and bees. Some examples of flocking behavior in both nature and engineering are demonstrated in Figure 1-2. Flocks are networks of dynamic systems with a dynamic topology. The topology is a proximity graph with limited sensing range, representing its limited communication capabilities that depends on the state of all agents which is locally determined for each individual agent[28].

A first computer animation model of flocking of birds in flight by exploiting a behavioral model based on some simple rules and local interactions was developed by Reynolds in 1986[48]. Most fundamental to flocking algorithms, the three heuristic flocking rules, introduced by[48], can be formulated as

- flock centering (*cohesion*): attract the agents to stay close to its flock-mates
- collision avoidance (*separation*): guidance to avoid collisions with nearby flock-mates
- velocity matching (*alignment*): the velocity vector (i.e. including direction) of the agents will be aligned with the nearby flock-mates.

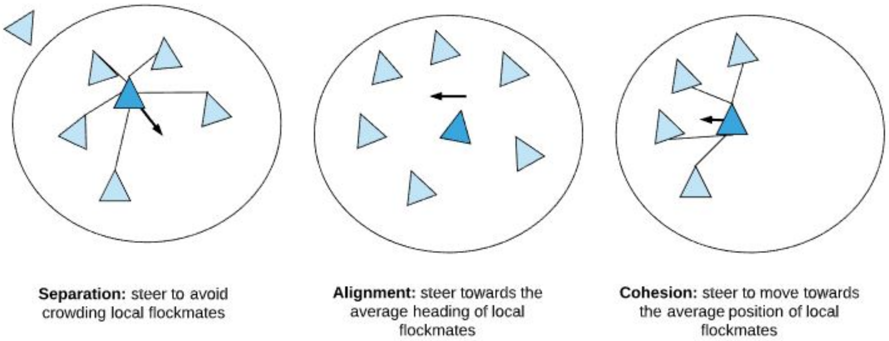

Separation: steer to avoid crowding local flockmates
Alignment: steer towards the average heading of local flockmates
Cohesion: steer to move towards the average position of local flockmates

Figure 1-3: The fundamental flocking rules of Reynolds.

Figure 1-3 demonstrates the three heuristic flocking rules that lead to the emergent collective flocking behaviour of MASs. The self-organized feature of flocks can provide a heuristic conception in the design of mobile sensor networks and robotics systems[38]. The flocking rules of Reynolds manifested its way to be the fundamental foundation in developing new flocking control algorithms. Further studies in the development of the flocking techniques is conducted by several other researchers where a modest list of one of the first researchers involved in this topic is discussed in Chapter 2.

The flocking framework is primarily used in massive sensing using mobile sensor networks, self-assembly of connected mobile networks and for military missions. The flocking algorithms have the potential to introduce self-healing, self-organizing and self-configuring capabilities in the functioning of distributed MASs. Moreover, the flocking algorithms exploit various intelligence techniques, potential functions and geometric approaches in order to realize the objective of the collective group by locally controlling the parameters of the flock-members[3]. Flocking involves swarm intelligence behaviour which refers to the superior capabilities of a swarm of robotic agents compared to individual agents. The use of the artificial potential function (APF), to encode the local interaction rules for achieving global performance, is inspired by the observations and models of the biologists. Groups in nature make use of a distributed control architecture whereby individuals respond to their sensed environment, but are constrained by the behavior of their neighbors[29].

Recently, the flocking formation control behaviour in MASs has encouraged astounding attention among the researchers. Researchers from diverse disciplines including physics, biophysics, computer science and control engineering have been fascinated by the emergence of flocking, swarming and schooling in MASs with local interactions[41]. This attention arises from its broad potential applicability for practical purposes where a distributed control framework is desired. To name a few, some representative applications include universal flight control system design, mapping and exploration, surveillance, search and rescue missions in dangerous environments, regional coverage and intelligence in military missions and object transportation[17].

Moreover, flocking algorithms in the framework of formation control have more interesting applications. Regarding small satellite clustering, it reduces the consumption of the fuel utilized for propulsion and increases the sensing capabilities. In automated highway systems, the throughput of the transportation network can be greatly increased if vehicles can form platoons at a desired velocity while maintaining a specified distance in between the vehicles[9]. The prevalent military and civilian deployment of unmanned vehicles, e.g. ground mobile robots, has been clearly seen in the last decades[17]. It is important to note that flocks of unmanned vehicles (UVs) can perform better and protect humans in an efficient manner in dangerous situations that might occur. Additionally, MRSs are also being employed in highly task-oriented missions as the lunar polar crater exploration missions that were conducted by exploiting UGVs, legged scout and immobile payload items[33].

Concluding, a distributed MRSs framework of flocking algorithms for formation control purposes has beneficial contributions to solutions that involve great advantages such as robustness of the system, efficiency and operational speed, maintainability, re-usability, flexibility, adaptability, fault-tolerance of the system and reduced operational costs.

1-2 Problem Formulation
Limitations in the State-of-The-Art Research

This section discusses the main motivation for studying APF-based flocking algorithms in perspective of more advanced modelled nonlinear agents involving non-holonomic constraints. This is done by discussing the limitations of the state-of-the-art literature in scope of flocking algorithms for formation control of MRSs. This study will consider a connectivity-preserving flocking algorithm for non-holonomic mobile networked MRSs. Moreover, the problem that will be investigated in this thesis study is formulated.

1-2-1 Motivation of Thesis Study

Despite a reasonable amount of literature concerning the flocking formation control of MASs is given, most of the existing results deal with simple double-integrator dynamics[41;55], holonomic mobile robots or simple dynamic/kinematic non-holonomic systems[72;74;23]. Moreover, some recent adaptive and connectivity-preserving flocking algorithms are presented in the literature concerning EL systems[11;14;64;15] and heterogeneous nonlinear dynamic systems[65;71], however, non-holonomic constraints are not considered in the mathematical modelling and control algorithm design of the systems. In fact, for practical application purposes, mobile robots involve non-holonomic constraints and embrace more complex mathematical models. The design of flocking formation control laws for dynamic EL non-holonomic systems is rather more challenging and involves additional complexity layers to the theoretical derivation due to the complicated coupling between the states of the non-holonomic agents. A more detailed discussion about the literature study on flocking formation control algorithms is provided in the Chapters 4 and 5 of the literature review in[61].

The behavior of non-holonomic systems appears in many advanced robotic systems that are practically relevant for applications, such as mobile robots, space manipulators, and multi-fingered robot hands[34]. In fact, non-holonomic dynamic systems are exceptionally interesting since they involve under-actuated systems, i.e. a system where the order of the control inputs is less than the order of the generalized coordinates. The analysis, control and planning of non-holonomic mechanical systems involve more challenging and complex techniques[34].

Motivated by the literature study in[61], the design of an APF-based flocking algorithm for non-holonomic Euler-Lagrange (EL) systems is of fundamental importance to obtain innovative and practical insights into the distributed control of such complex nonlinear systems. To the best knowledge of the author, this is the first study to initialize a flocking algorithm for formation control of such advanced nonlinear systems involving much complexity layers such as the non-holonomic constraints and Brockett's necessary conditions[5]. This introduces more challenges and difficulties in the design and analysis of control algorithms.

APF-based flocking control algorithms are not limited to higher-level models and can be exploited for more advanced nonlinear dynamic models and control strategies to achieve formation flocking control with collision avoidance. The main advantages of this method are that the algorithm can be applied in real-time control, requires less computation, has flexibility and adaptability properties and is very suitable for effective implications in distributed control of MRSs. The fundamental flocking rules can mathematically be implemented through APF techniques of inter-agent distances in combination with consensus algorithms for synchronization purposes.

After an extensive literature study by[61], to the best author's knowledge, a flocking control algorithm for formation control of non-holonomic EL systems has never been studied and evaluated. The study of such systems might lead to interesting results and understanding in perspective of a theoretical and computational framework to get insights into the development of swarm intelligent robotics concepts from nature-inspired flocking algorithms.

1-2-2 Problem Description

In this thesis study, we primarily focus on the distributed flocking with connectivity preservation problem for homogeneous networked non-holonomic EL systems under a proximity graph defined according to the maximum of the relative distance between a pair of agents. This study is expanding novel features on results presented in the current existing work in the literature on the flocking and formation control of non-holonomic agents[31;74;23;32;72]. A distributed control architecture in which the controller is locally implemented using only local information from the neighboring agents in the proximity graph, to complete global tasks cooperatively, increases flexibility and robustness to the control algorithm[15].

The flocking control algorithms to be designed are distributed in the sense that only neighboring agents in proximity can obtain information from each other, involving only local interactions. In all proposed control algorithms, only one-hop neighbors' information is required, i.e. the relative positions and velocity vector measurements between the proximity neighbors and the absolute position and velocity vector measurements.

This study will mainly discuss three flocking algorithms for non-holonomic EL agents from the perspective of (nonlinear) control theory, analysis and mathematical modelling of EL non-holonomic systems and the understanding of the heuristic flocking behavior in nature. The ***first algorithm*** discusses a **connectivity-preserving APF-based flocking control protocol for formation control of non-holonomic EL multi-robot systems**. The preservation of the network connectivity is of significant importance for the flock stability and synchronization (i.e. consensus), since these properties firmly depend on it. It is important to note that in real applications, such as flocking and rendezvous, the proximity graph of the flock is dynamically changing and state-dependent[11]. Therefore, assuming that the underlying communication network is connected or jointly connected for all future time is not a reliable and practical assumption. As a consequence, the problem of connectivity-preserving has to be explicitly integrated in the design of the flocking algorithm that guarantees and preserves the connectivity for any initial connected proximity graph at all time instants.

Then, the ***second algorithm*** builds further on the first algorithm by **including a (virtual) leader, encapsulated as a reference trajectory, in the flock that is compatible with the non-holonomic kinematic constraints of the following agents**. The purpose of the virtual leader(s) is to introduce the mission, i.e. to provide guidance to the flock. Moreover, the virtual leader is employed for robustness against uncertainties in the model and the system parameters which might lead to convergence to a zero velocity which is of fundamental importance for the intrinsic stability in terms of the orientation angle. A virtual leader is used for cohesive flock guidance to prevent fragmentation and collapse in the flock[41]. We emphasize that there is no (physical) leader among the vehicles. More specifically, no ordering of vehicles is required a priori, such that any vehicle is interchangeable with any other, supporting the self-organization trait into the control algorithm. This feature of the approach adds robustness and fault-tolerance of the group to the failure of an individual vehicle.

In practice, the system moves in constrained flocking environments where interaction with obstacles can not be neglected in the design of the control algorithm. Therefore, the ***third algorithm*** includes **obstacle avoidance capabilities to the flocking algorithm**. In fact, it is not an easy task for the robot to distinguish between fellow robotic agents and obstacles. This is due to the fact that robots commonly employ laser range finders that do not have the capability of distinguishing between robots and obstacles[51]. To be more precise, the reason for this is that laser range finders only return range data. The neighboring robots and the detected obstacles that are detected within the detection range are both considered as objects and no distinction is made between them in this algorithm.

After establishing theoretical results for the flocking algorithms, exploiting nonlinear control theory concepts where a lower bounded ”energy-like” Lyapunov function candidate is defined that is non-negative, a computational analysis will be executed by performing an advanced numerical simulation study as a complement to the analytical framework in order to visualize the performance for verification of the theoretical analysis.

Control Objective

- *The **primary objective** of this study is to design a connectivity-preserving APF-based flocking algorithm for formation control of networked non-holonomic Euler-Lagrange (EL) multi-robot systems.*
- *The second algorithm builds further upon the first algorithm by including a virtual leader in the proximity network that is compatible with the non-holonomic kinematic constraints of the following agents for cohesive flock guidance in the environment.*
- *Then, to obtain a more practical algorithm, the third flocking algorithm includes obstacle avoidance capabilities without distinguishing between mobile robots and obstacles, as a complement to the second algorithm.*

The problem formulation will more extensively be discussed in Chapter 4 through a mathematical framework.

1-3 Main Contributions & Challenges

This study on the design of an innovative flocking algorithm for formation control of non-holonomic networked EL multi-robot systems has the following novel and/or expanded features compared with the current existing literature.

- This study considers each agent as a non-holonomic Euler-Lagrange (EL) system involving non-holonomic constraints, which is more practical and realistic, while in most of the exisiting literature, the agents are modelled as single- or double-integrator dynamics (e.g.[41;35;54;38;6;51;60]), EL (e.g.[11;14;64;15]) and general nonlinear dynamic systems (e.g.[65;71;66]), without considering non-holonomic constraints. The mobile agents are mathematically modelled by exploiting the

EL methodology for the derivation of the non-holonomic dynamical model equations of motion[34;19;39].
In reality, many physical systems are inherently nonlinear and cannot be embraced by simple (linear) mathematical models. Among the nonlinear systems, non-holonomic EL systems can be described for a large class of physical mechanical systems of practical interest such as autonomous mobile robots and space manipulators[34]. Therefore, it is of scientific and practical interest to study flocking algorithms for such complex nonlinear systems involving non-holonomic constraints, in perspective of a theoretical and computational framework, to obtain valuable and innovative scientific and/or practical insights.

- To solve the flocking problem for a network of non-holonomic EL mobile robots, a novel distributed control algorithm has to be designed to achieve the objective. Due to the complexities that arise with the control and modelling of such advanced nonlinear systems, mainly owing it to the involved non-holonomic constraints and high nonlinearities of the system, the current algorithms developed in the literature for the already existing mathematical models of mobile agents can not straightforwardly be implemented.
  Therefore, this study considers a novel nonlinear flocking control protocol, well-known as computed torque nonlinear feedback control in the robotics literature[34;53], that is able to effectively solve the problem by obtaining a more convenient model for control and analysis purposes. The control algorithm is conceptually relative simple and can fully compensate the nonlinear forces in the motion of the robots and achieve high accuracy control for a large range of robot speeds and workspace[53].
  It is important to note that, in addition to alignment of linear velocities, our flocking algorithm has to consider also the synchronization of the velocity vector, which includes both the velocity and angular velocity, and the heading angles, for the robots in proximity. This is done by conveniently decoupling and designing the translational and rotational control inputs, respectively. Therefore, the proposed algorithm is more practical and involves more complexities as those in the existing flocking algorithms.

- This study considers the integration of a flocking algorithm (including connectivity-preservation, collision and obstacle avoidance, and velocity vector alignment) with a virtual leader encapsulated as a reference trajectory to guide the mission of the flock. The virtual leader is included in the proximity network of the multi-agent system, i.e. the information of the leader is constrained to its followers in its proximity[15]. The above and the fact that non-holonomic EL agents are subject to non-holonomic constraints, introduce further complexities to the algorithm, since the generated trajectory has to be compatible with the non-holonomic constraints and not all followers have explicit access to the leader's information. Note that, even for agents modelled as single- or double-integrator dynamics, this is a very challenging problem[7], not to mention the case of nonlinear Lagrange systems involving non-holonomic constraints.
  Note that, although it might have beneficial contributions to the cohesiveness of the flock and alignment, the assumption that the information of the virtual leader is available to all its followers in its proximity[8;29] is against the local interaction rule of the nature phenomena of our problem and may be difficult to implement in engineering applications[55;15].
  The connectivity-preservation, collision avoidance and velocity vector alignment problems are explicitly included into the flocking control algorithm by conveniently designing the APF and consensus protocol, which is a mathematically elegant tool to encode the local interaction rules.

- The performance evaluation of the flocking control algorithm is a difficult task. It is difficult to predict and analytically investigate the cooperative behavior of the system under the flocking protocol. There does not exist a general analytical framework of performance metrics for evaluation of the algorithm. Similar to e.g.[15;14;72;23], initially the flocking algorithms will be analyzed through a theoretical framework using nonlinear stability concepts for control systems. As initialization of the study, theoretical analysis is the least expensive way of studying the flocking characteristics of the networked mobile robot system.
  Then, different to the computational framework in most of the existing flocking algorithms in

the literature, the performance of the system will be extensively simulated by a more advanced computational analysis, in both free and constrained environments, where robots are modelled isomorphic to a corresponding rigid body with a fixed dimension and a sensing radius R_s representing the interaction radius of the proximity graph, respectively. Note that simulations, though very important in nonlinear control, have to be guided by theoretical concepts. The blind simulation of nonlinear control systems is most likely to produce few and/or misleading results, especially due to the great richness of behavior that nonlinear systems can exhibit, depending on initial conditions and inputs[53].
By mathematically simulating the behavior of the nonlinear control system, valuable insights in the emerging global swarm dynamics can intuitively and guided by theory be obtained, and the effectiveness of the algorithm can be verified. This allows assessing the performance of the control algorithm, eliminating or mitigating the need for time-consuming experiments with real robots and facilitating the algorithm optimization within a heuristic framework.

.

- **Challenges in the Proposed Flocking Algorithms**

– The design of (flocking) control algorithms for non-holonomic systems is quite involved, mainly due to the non-holonomic constraints, and requires careful consideration in perspective of nonlinear control theory. In particular, it does not satisfy the Brockett's necessary conditions[5], where as a consequence the system can not be feedback stabilized by using static continuous feedback control laws. This motivates the development of other control strategies as will be discussed more elaborately in Subsection C-1-2 and 3-1-2.
The mathematical modelling and control of non-holonomic mobile robots involves complexity in the dynamic model, electronics and control theory. The system dynamics consist of highly nonlinear terms, involves under-actuated systems and non-holonomic constraints, which complicate the mathematical modelling, control and analysis.

– Designing decentralized control algorithms for distributed systems involves control challenges that are not present in single robotic systems or centralized systems. These challenges include complex interactions, high system dimensionality, incomplete information and uncertainties. This motivates studying the swarm robotics concepts in perspective of the theoretical framework which can have significant effects on the development and understanding of nature-inspired formation control algorithms.

– In the literature of flocking algorithms, very few results exists on flocking algorithms with obstacle avoidance capabilities, beside the results in[41;51] for agents simply modelled by double-integrator dynamics. In practice, the robots move in a constrained environment with obstacles, therefore, interaction of the robots with obstacles can not be neglected. This is a very challenging feature to include in the control algorithm. The presence of obstacles imposes even more complexities and difficulties in the theoretical analysis and control, as it will cause interference to the connectivity of the network of the flock, where the stability of the system will become questionable.

1-4 Outline of MSc Thesis

This section provides an outline of the remainder of this study to provide guidance in the reading of the thesis. The remainder of this study can be listed as follows.

- Chapter 2, provides an overview of the relevant related background information to understand the state-of-the-art methods in the literature on formation control and flocking algorithms.
- Chapter 3, provides preliminaries to the theoretical analysis of the flocking algorithms used in this study by discussing some essential concepts in algebraic graph theory, proximity graphs, mathematical analysis of non-holonomic mechanical robotic systems, nonlinear control theory and mathematical definitions. In addition, Appendix C provides supplementary information on the mathematical modelling and control of non-holonomic mechanical systems.
- Chapter 4, introduces the problem formulation through a mathematical framework and discusses some preliminary results and definitions regarding the mathematical modelling of dynamic non-holonomic EL multi-agent systems, proximity network architectures and the general structure of the flocking control algorithm to be designed in the subsequent chapters.
- Chapter 5, discusses a flocking algorithm for formation control of networked non-holonomic EL systems, with its application to a networked differential-drive wheeled mobile robots (WMRs) system. The flocking algorithm is initially analyzed through a theoretical framework. Then, the theoretical results are verified through a numerical simulation study through a computational framework.
- Chapter 6, introduces the concept of a virtual leader in the proximity network of the flocking control algorithm as a complement to the previous flocking algorithm. Similarly, the extended flocking algorithm is initially analyzed through a theoretical framework. After, the theoretical results will be used for guidance and insights in performing the numerical simulation study to verify the obtained theoretical results.
- Chapter 7, introduces obstacle avoidance capabilities to the flocking control algorithm of Chapter 6 without distinguishing between robots and obstacles. The flocking algorithm is analyzed through both a theoretical framework and a computational framework, as in the previous chapters.
- Chapter 8, provides the reader with some concluding remarks on this study and future recommendations for prospective studies on the discussed flocking algorithms to gradually work towards a reliable practical control algorithm for the flocking of networked mobile robotic systems.

Chapter 2

Background

In this chapter, an overview of the relevant related background work is given to understand the state-of-the-art literature on formation control techniques and flocking algorithms for formation control.

2-1 Overview of Related Work

Initially, it is of significant importance to highlight the relevance of a convenient agent structure and underlying control architecture in the design of multi-robot control systems. Then, the relevant theory to understand the current state-of-the art methods used in flocking and formation control algorithms are concisely and effectively discussed.

2-1-1 Heterogeneous & Homogeneous Agents

Initially, one of the primary issues to deal with when designing formation control algorithms is to decide which agent structure will be used in the multi-robot system as it is fundamental for the operation of the underlying control architecture. The group of agents can consist of homogeneous or heterogeneous teams. Homogeneous teams consist of flock-members that contain completely the same control software and hardware. In the multi-robot system, the capabilities of the homogeneous individual robots are identical, although the physical structures can be different. On the other hand, heterogeneous robots have different hardware and/or control software. In a heterogeneous multi-robot system, the robot capabilities are particular to a specific robot where the different functionalities and designs commonly complement each other. Therefore, the robots can be specialized for performing specific tasks.
Systems of homogeneous robots are robust due to the fact that no critical robot exist by performing a mission. However, using heterogeneous robots may be desired or necessary in several applications. These applications involve the case where only a few robots are able to have sensors and serve as leader for the multi-robot system in order to provide intelligence to the followers. However, heterogeneous systems generally involve more complexity compared to homogeneous systems regarding the task planning. In addition, heterogeneity introduces more complexities in the modelling of robots that might potentially fail in the group. This reduces the robustness in the SR system towards the individual failure of robots. Therefore, most of the literature on flocking algorithms are based on homogeneity of systems as it is generally preferred above heterogeneous systems.

2-1-2 Control Structure: Centralized & Decentralized

It is of fundamental importance to conveniently choose a control and communication structure in the design of SR multi-robot systems. The robustness of systems is strictly related to the control and communication structure that is implemented to coordinate the robots into a desired formation. Formation control of multi-robot systems (MRSs) involves both *centralized* and *decentralized* control structures. In general, the control architectures can be decomposed into centralized, decentralized and distributed control structures, see Figure 2-1.

The centralized control structure has a single computational unit that processes all the information that is needed to achieve the desired control objectives. Over time, many studies on formation control of MRSs implemented through a centralized control structure have been discussed. In the centralized framework, the central control agent has global information through collection and synthesis of data of individuals. The strengths and drawbacks of this control structure are summarized in Table 2-1.

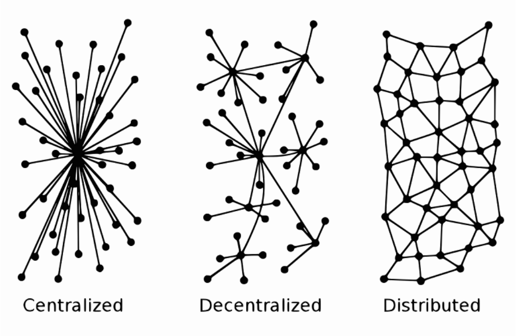


Figure 2-1: The centralized, decentralized and distributed control architectures [source: control structure]

The decentralized control structures can further be branched into *distributed architectures* and *hierarchical architectures*, where the distributed control structure is most frequently used in the literature for controlling MRSs in formations. Exploiting the distributed control framework as control architecture of the system satisfies more desirable practical requirements. The multiple robots in the system act completely autonomous in the decision-making process based on local control laws and local information of the robots and environment. Moreover, limited communication among the robots, lack of sensing to attain global information and scalable capabilities of the robotic agents are involved characteristics in the distributed control framework. The strengths and drawbacks that come with the control structures are summarized in Table 2-1.

The hierarchical structure is a hybrid architecture, intermediate between a centralized architecture and a distributed architecture[69]. A hybrid control architecture can be desirable by conveniently exploiting a balanced framework between both the centralized and decentralized structure. This can be achieved by integrating the characteristics of decentralized control structures with some high-level supervision of the system through incorporating functions that support the use of a central controller through leader election, command of a leader or command of several leaders[2]. In fact, the use of hybrid systems can provide guidance to the overall behavior of the cohesive flock by exploiting e.g. a virtual leader, while simultaneously reducing the computational complexity of the system by its distributed architecture where the interaction and communication involves locality with their proximity neighbors.

Concluding, using hybrid control structures can overcome the deficiencies associated with centralized or decentralized control structures and hybrid-distributed architectures can provide more efficient and robust algorithms in the formation control of MRSs.

Table 2-1: Summarizing analysis of control architectures in formation control of MRSs[61].

Summary of control architectures	
Strengths	Drawbacks
Centralized	
The collective swarm behavior can be global optimally be planned a priori in accordance with the global available knowledge. Centralized control structures support direct control of each individual robot that facilitate prediction of the swarm behavior in the overall system.	Computational complexity of the system increases with number of individual robots. Lacks in robustness in relation to communication failures and dynamic environments. Poor scaling in control algorithms. Highly vulnerable to malfunctions of the central agent that globally controls the system.
Decentralized	
Significantly reduce of signals, delays and impracticalities in relation to the centralized framework, Less computational effort as a result of the distribution of the computational complexity Supports the scaleability, increased flexibility and robustness through exploiting parallelism in MRSs. The multiple robots in the systems act completely autonomous in the decision-making process based on local control laws and local information of the robots and environment	Can reduce the stability of the system in terms of oscillatory behavior due to the lack of global knowledge and supervision. The complexity in prediction of the behavior of decentralized systems, as their behavior emerges from numerous local interactions rather than specifications

2-1-3 Control Approaches: Flocking & Formation Control

Formation Control Approaches

The most existing formation control strategies can be classified into some basic strategies. The most common strategies are summarized in Figure 2-2, where each formation control strategy has its own advantages and disadvantages. This is analyzed in Table 2-2.

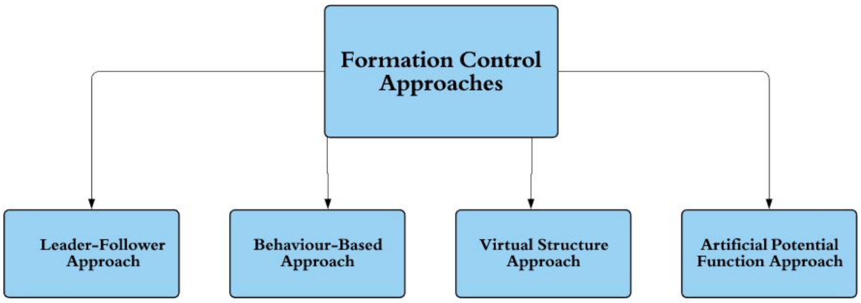

Formation Control Approaches
Leader-Follower Approach
Behaviour-Based Approach
Virtual Structure Approach
Artificial Potential Function Approach

Figure 2-2: Overview of formation control structures to control the flocking phenomenon in MRSs[61].

The methods for formation control of MRSs are evaluated in an innovative way to assist with the assessment of the suitability of the methods for its applicability in practice. The interested reader might consult the literature review in[61] for a more elaborate discussion on the operation of the methods.

Table 2-2: Summarizing analysis of formation control techniques of MRSs[61].

Methods	Strenghts	Drawbacks
Leader-follower	Reduces to a tracking problem where standard control theory techniques can be exploited for stability. Furthermore, the approach is easy to understand and implement in formation control applications	Poor disturbance rejection property due to the chain structure. This centralized method is less suitable for a large multi-robot system. (i.e. it does not support scalability) No explicit feedback from follower to leader and has no leader fault-tolerance property.
Behavior-based	Distributed approach, all behaviors run in parallel, real-time and less information is required for communication. Useful practical features in guiding the MRSs in unknown and dynamically changing environments with only local sensory information. Capability to solve multi-objective problems.	The complexity in the mathematical analysis and design. Difficulty in the prove and guarantee of the stability of the system.
Virtual structure	Good performance in formation maintenance during maneuvers. Easy to prescribe the coordinated behavior between the individual robots in the formation	The requirement of a rigid formation structure, as a consequence, the control approach lacks in flexibility and adaptability.
APF	Can be applied in real-time, has flexibility and adaptability behaviors incorporated and very effective in distributed control of MRSs. The approach is very attractive for its mathematical elegance, relative simplicity and effectiveness in creating reactive behaviors. The use of this technique is not limited to higher-level models and can be exploited in complex dynamical models of agents.	This approach involves the local minima. The combination of both attractive and repulsive potential functions might lead to equilibria that trap the multi-robot system into undesired positions[41;37]. This will be discussed in more detail in Chapter 7 and Appendix B. In fact, due to the presence of uncertainties in practical situations, this might not be a big problem in practice.

Flocking: Formation Control Algorithms

As already mentioned, it is revealed that a lot of collective behaviours discovered in different fields are based on relatively simple local interaction rules. Being inspired by, Reynolds proposed a bird flocking animation model based on three basic rules, also known as the rules of Reynold[48], see Figure 1-3. The flocking rules of Reynolds manifested its way to be the fundamental foundation in developing new flocking control algorithms. Further studies in the development of the flocking techniques is conducted by several other researchers where a modest list of one of the first researchers involved in this topic is discussed next.

Vicsek et al.[63] investigated the emergence of self-ordered motion in particle systems with biologically motivated interacting particles mainly focusing on alignment in self-driven particle systems. This is a simplified and special form of the work done by[48] and is based on the self-propelled particle model. Moreover, the study in[63] presents simulation results where the nearest neighbor rule is investigated, i.e. the agents only locally interact with their nearest neighbors. Toner et al.[62] presented a quantitative continuum theory of flocking. Then, in[30], Levine et al. developed a novel algorithm where the particles rotate around a common center to simulate ant mills. Moreover, Helbing et al. performed[20] a study to simulate the features of the escape panic phenomenon employing an empirical particle-based flocking model.

Inspired by these studies, the research in the flocking behavior with focus on the consensus problem and network topology of systems was further developed by well-known researchers in this field like Olfati-Saber, Murray, Jadbabaie, Fax and Tanner, respectively, on sub-topics of the flocking phenomenon[42;43;22;50;13]. The objective of the networked dynamic systems is distributed computation through agreement. The alignment problem in the framework of flocking does not involve constraints on the consensus value. A vast and exhaustive amount of control laws have been proposed in the literature for collective behaviour purposes[61]. The majority of them involve the implementation of Reynolds rules. As explicated in Section 1-1, the cohesion and separation rules are frequently implemented through APF of inter-agent distances. The alignment rule can be realized using a consensus protocol of the multi-agent system[41;70;7]. The fundamental objective in flocking formation control algorithms is to design local control laws to achieve the desired emergent collective behavior. The flocking communication architecture for MRSs is illustrated in Figure 2-3. The existing control structures provided in the literature to perform and maintain the flocking phenomenon are discussed in the subsequent section.

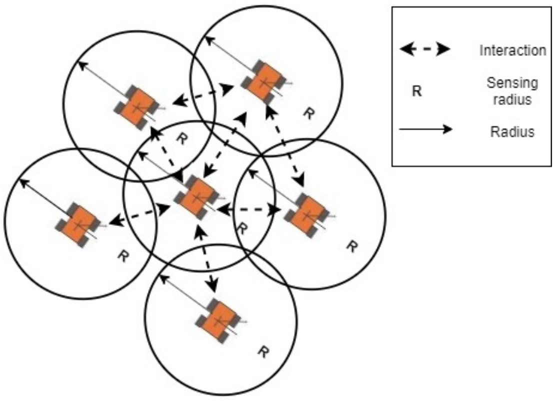


Figure 2-3: Flocking formation control of robots in a multi-robot system with locality of communication (i.e. limited communication sensing radius R) and locality of interaction (i.e. only interaction with neighbors). The robots locally interact within their proximity neighborhood.

Flocks of Multi-Agent Systems: Control Structures

Basically, the formation control framework is inspired by the natural behaviour of animals involving birds flocking or fish schooling. In this concept, the agents adopt to certain formations to enhance the chance of survival of the individual agents in a group strategy. By resembling the formation behaviour of animals, groups of unmanned vehicles can be employed in formation to achieve an enhanced level of autonomy and for accomplishment of tasks that involve much complexity. Advances in the development of the technology offers the possibility to implement the formation control concept to practical real-word applications involving MRSs.

A crucial application of the flocking formation control of MRSs is the rescue missions carried out by autonomous Unmanned Ground Vehicles (UGVs) formations in disaster areas to minimize exploration time and reduce the risk of further casualties[33]. Moreover, efforts have been made to pave the formation control concept for area mapping and border patrol and surveillance purposes. Additionally, highly task-oriented missions exploit multiple unmanned vehicles (UVs) in special cases such as the lunar polar crater exploration missions conducted using wheeled UGVs, a legged scout and several immobile payload items[33].

The flocking algorithms involve knowledge distribution which can be local and global. The distribution of the knowledge is of significant importance for the underlying control algorithm in flocking. The non-uniform distribution of the knowledge and control of the flocking phenomenon externally gave rise to some control structures that are proposed and studied in the literature[3]. Figure 2-4 provides an overview of the most existing control structures employed for controlling and performing the flocking phenomenon.

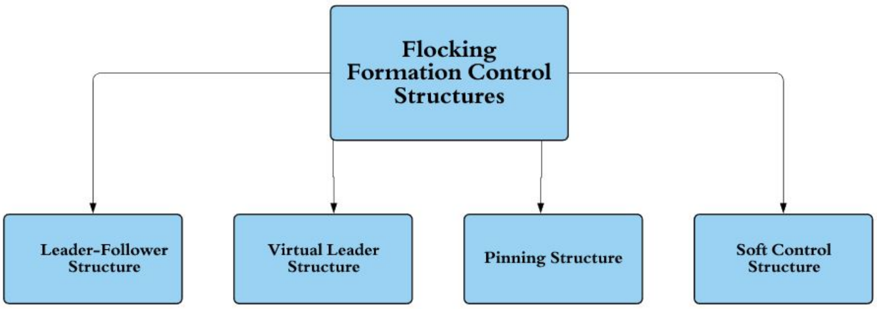


Figure 2-4: Overview of formation control structures to control the flocking phenomenon in MRSs[61].

Leader-Follower Structure Similar to the leader-follower structure summarized in Table 2-2, two kind of agents are involved, leader and followers. The leader has an independent and enhanced capability compared to the followers. It is required for the followers to follow the leader while a formation structure is maintained. A system that uses the leader-following flocking control structure can have one or more agents that function as flock leaders with global information about e.g. a desired path. On the other hand, robots only locally communicate with their neighbors.

Virtual Leader Structure The concept of virtual leaders was first developed by[41] to study the flocking behaviour in constrained environments with obstacles and a collective objective. It is important to note that the virtual leader is not a physical agent. In fact, it denotes a moving reference point for guidance of the flock of the multi-robot system.[55] proposes a stable flocking algorithm to study the cases where a minority of agents are informed and a virtual leader with varying velocity. Furthermore,[35;56] propose multi-target tracking flocking algorithms for multi-agent systems with multiple virtual leaders. This might be useful in multi-objective flocking scenarios.

Pinning Structure The direct control of an individual agent in a dynamic topology of MRSs does not necessarily require stable flocking behaviour. Hence, only a fraction of agents may have the information about the leader and are denoted as informed agents in the literature (e.g.[54;15]). If the entire flock has information about the target, local inter-robot interactions mainly serve the purpose of cohesive formation maintenance while simultaneously avoiding collisions. However, in case only a fraction of agents are informed about the target (i.e. informed agents), the local interactions in addition serve the purpose of spreading the knowledge to all individuals in the flock. Only the informed agents contain a navigational feedback term and it is shown that the flocking phenomenon still can be achieved.

Soft Control Structure Usually, soft control structures involve an externally controlled special agent[18]. Local control structures of the existing agents in the system remain unchanged. This structure can be implemented in complex MRSs to intervene in the collective behavior (headings) of the group by soft control. A specified method is to add a special agent, called a 'shill', which can externally be controlled and is treated as an ordinary agent[18]. It is important to note that this approach is different than a distributed architecture. It may be used as a natural and efficient intervene method in the challenging control of complex MRSs from the approach of distributed control[3].

Chapter 3

Theory

Mathematical Background

Throughout this thesis numerous mathematical definitions and theoretical results will be used. This is briefly reviewed in this chapter and Appendix C. The analysis and theoretical framework of the flocking algorithm for advanced non-holonomic systems rely on some essential concepts exploited from (algebraic) graph theory[16], spatially induced graphs/proximity nets[41], mathematical modelling of non-holonomic mechanical robotic systems[34], Lyapunov-like stability theory[25] and controllability for nonlinear non-holonomic systems[5;34;52].

3-1 Mathematical Modelling & Control of Non-Holonomic Mechanical Systems

The aim of this section and Appendix C is to provide a theoretical framework using tools from nonlinear control theory for analysis of non-holonomic mechanical systems. This is an interesting topic since non-holonomic constraints are present in many advanced robotic architectures[34]. In this study, we discuss a unified framework for mathematical modelling of non-holonomic mechanical systems exploiting the Euler-Lagrange (EL) methodology.
The behavior of non-holonomic systems in advanced robotic structures is exceptionally interesting since it involves under-actuated systems. Under-actuated systems have very important applications such as free-flying space robots, underwater robots, surface vessels, manipulators with structural flexibility, etc[68]. They are used for reducing weight, cost or energy consumption, while still maintaining an adequate degree of dexterity without reducing the reachable configuration space. Some other advantages of under-actuated systems include less damage while hitting an object, and tolerance for failure of actuators[68]. The kinematic, dynamic modelling and control of non-holonomic mechanical systems are discussed. Some examples of the kinematic and dynamic modelling of non-holonomic systems can be consulted from[34;1].

3-1-1 Methodology for Dynamic Modelling of Non-Holonomic Mechanical Systems

In the literature, the most used mathematical modelling methods involve the Euler-Lagrange (EL) and Newton-Euler methodology, respectively[19]. Also, various other formalisms exist for the modelling of systems such as the Kane's method. The Newton-Euler method involves two kind of forces; the externally impressed forces by the actuators and the constraint forces due to the interaction of the robot platform with the ground through the wheels[19]. Moreover, a system that consists of interconnected elements may have interaction forces between the elements through elements like gears, springs and frictional elements.

Normally, these forces are not easy to be quantified which leads to practical difficulties of the Newton-Euler approach. On the other hand, the EL methodology exploits the energies of the system, i.e. the kinetic and potential energy, respectively. The expression of the forces into energies involves scalar quantities which can easily be expressed in terms of the coordinates of the system[19]. Moreover, the Lagrangian approach requires the generalized coordinates of the system to be independent. The Lagrangian formulation provides a powerful and versatile method to model the equations of motion for holonomic systems.

However, when dealing with non-holonomic systems, the kinematic constraint equations are incorporated in the dynamic equations employing additional Lagrange multipliers[34]. The unknown Lagrange multipliers are usually functions of the generalized coordinates and time, and are obtained as an element of the solution. Then, as in[34;19], a method is proposed to eliminate the Lagrange multipliers by exploiting the properties of the nullspace of the constraint matrix and obtaining a reduced dynamic model to facilitate the theoretical analysis of the system and control algorithm.

It is important to note that due to the mapping to a lower dimension, relevant information on the system might be lost and hence it is of significant importance taking this into consideration for the design and implementation of flocking control algorithms. It is well known that the nonlinear model of a non-holonomic robot can be transferred into single-integrator kinematics utilizing feedback linearization techniques[34;59]. However, this approach will lead to the loss of orientation information of the robot which is very vital for high operational precision in engineering applications[10].

Euler-Langrange Methodology

In this subsection, a unified framework for the dynamic modelling of non-holonomic mechanical systems, exploiting the EL methodology, is provided. Then, the dynamical model of the n-dimensional mechanical system involving k kinematic constraints is partially linearized through a computed torque nonlinear feedback approach[34].

The Lagrangian $\mathcal{L}$ is defined as the difference between the kinetic and potential energy of the system

$$\mathcal{L}(q,\dot{q}) = \mathcal{T}(q,\dot{q}) - U(q) = \frac{1}{2}\dot{q}^T\mathcal{M}(q)\dot{q} - \mathcal{U}(q) \tag{3-1}$$

where $\mathcal{M}$(q) and $\mathcal{U}(q)$ denote the positive-definite inertia matrix and the potential energy of the mechanical system, respectively. Then, the EL motion equations are formulated as

$$\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{q}}\right)^T - \left(\frac{\partial\mathcal{L}}{\partial q}\right)^T = \mathcal{A}(q)\lambda + \mathcal{S}(q)\tau \tag{3-2}$$

with $\mathcal{S}(q) \in \mathbb{R}^{n\times m}$ a matrix mapping the external inputs $\tau \in \mathbb{R}^m$, where $m = n - k$, to forces and torques that execute work on the generalized coordinates q. The vector $\lambda \in \mathbb{R}^m$ denotes the unknown *Lagrange multipliers*. Moreover, the term $\mathcal{A}(q)\lambda$ represents the constraint force vector with $\mathcal{A} \in \mathbb{R}^{n\times m}$. By solving the Euler-Lagrange motion equations, we obtain

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{q}}\right)^T &= \mathcal{M}(q)\ddot{q} + \dot{\mathcal{M}}(q)\dot{q} \\ \left(\frac{\partial\mathcal{L}}{\partial q}\right)^T &= \frac{1}{2}\left(\frac{\partial}{\partial q}(\dot{q}^T\mathcal{M}(q)\dot{q})\right)^T + \left(\frac{\partial\mathcal{U}(q)}{\partial q}\right)^T\end{aligned} \tag{3-3}$$

Then, by combining (3-1) and (3-2) and substituting the results of (3-3), the dynamical model of the non-holonomic constrained mechanical system can be formulated as

$$\underbrace{\mathcal{M}(q)\ddot{q} + \dot{\mathcal{M}}(q)\dot{q} - \frac{1}{2}\left(\frac{\partial}{\partial q}(\dot{q}^T\mathcal{M}(q)\dot{q})\right)^T + \left(\frac{\partial\mathcal{U}(q)}{\partial q}\right)^T}_{d(q,\dot{q})} = \mathcal{A}(q)\lambda + \mathcal{S}(q)\tau \tag{3-4}$$

which can be simplified and rewritten to

$$\begin{aligned} &\mathcal{M}(q)\ddot{q} + d(q,\dot{q}) = \mathcal{A}(q)\lambda + \mathcal{S}(q)\tau \\ &\mathcal{A}^T(q)\dot{q} = 0 \end{aligned} \tag{3-5}$$

Next, we consider a matrix $G(q)$ where its columns span a basis for the nullspace of the transposed constraint matrix $\mathcal{A}^T(q)$, as in (C-7), such that we obtain

$$\mathcal{A}^T(q)G(q) = 0$$

This result is exploited to eliminate the Lagrange multipliers by left multiplication of both sides in the equations of (3-5) by $G^T(q)$. Then, the *reduced dynamical model* is obtained

$$G^T(q)[\mathcal{M}(q)\ddot{q} + d(q,\dot{q})] = \cancel{G^T(q)\mathcal{A}(q)}\lambda + G^T(q)\mathcal{S}(q)\tau$$

as a set that consists of m differential equations. Now, a realistic assumption is

$$det[G^T\mathcal{S}(q)] \neq 0$$

which is valid in most practical scenarios. A convenient way is to merge the kinematic and dynamical models into a $(n+m)$-dimensional reduced state-space model[34]

$$\begin{aligned} \dot{q} &= G(q)v \\ \dot{v} &= \hat{\mathcal{M}}^{-1}(q)m(q,v) + \hat{\mathcal{M}}^{-1}(q)G^T(q)\mathcal{S}(q)\tau \end{aligned}$$

where $v \in \mathbb{R}^m$ denotes the reduced velocity vector and the new system matrices are

$$\begin{aligned} \hat{\mathcal{M}}(q) &= G^T(q)\mathcal{M}(q)G(q) > 0 \\ m(q,v) &= G^T(q)\mathcal{M}(q)\dot{G}(q)v + G^T(q)n(q,G(q)v) \end{aligned}$$

Then, a *partial linearization* can be performed by exploiting a computed torque nonlinear feedback strategy[34]

$$\tau = [G^T(q)\mathcal{S}(q)]^{-1}(\hat{\mathcal{M}}(q)a + m(q,v))$$

with $a \in \mathbb{R}^m$ denoting the reduced vector of accelerations. As a result, the system can be formulated as

$$\begin{aligned} \dot{q} &= G(q)v \\ \dot{v} &= a \end{aligned}$$

The first n equations are the kinematic model whereas the second m equations involve the dynamic extension. The torque nonlinear feedback input requires measurements of v, however, the measurements may not be directly available. Following the steps in[34], we can write the closed-loop state-space model compactly in the following form

$$\dot{x} = f(x) + g(x)u = \begin{bmatrix} G(q)v \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I_m \end{bmatrix} u \tag{3-6}$$

with $x = [q,v]^T$. The model in (3-6) is known as the *second-order kinematic model* of the constrained mechanical system.

Summarizing, for non-holonomic systems, it is possible to cancel dynamic parameters via nonlinear feedback, assuming[34]

- The dynamical model is exactly known.
- The complete state of the system is measurable.

3-1-2 Control of Non-Holonomic Systems

It is relevant to initially focus on the control of first-order kinematic models of non-holonomic systems and actually it is not a limitation. If the control problem is solved for the kinematic system in (C-7), extending the solution to the second-order kinematic model and subsequently at the dynamical level is straightforward[34].
In fact, note that any configuration q is an equilibrium configuration, q_e, of the unforced system[34]. It is evident from the analysis of non-holonomic systems that the kinematic model of a (completely) non-holonomic system is controllable, i.e. at least one convenient control input exists that drives the system from its initial configuration q_0 to a desired configuration q_f[34].
For robustness against perturbations, uncertainties in the model and initial condition errors, a feedback control strategy is desired by using the the *configuration error* $e = (q_f - q)$ in each step. However, feedback control of non-holonomic systems is confronted with a major difficulty as a result of *Brockett's* condition[5;52].

Theorem 1 (Brockett's Necessary Conditions[5]). *A beautiful general theorem on necessary conditions for feedback stabilization of nonlinear systems was given by Brockett.*
Consider the nonlinear system $\dot{x} = f(x, u)$ with $f(x_0, 0) = 0$ and f(.,.) continuously differentiable in a neighborhood of $(x_0, 0)$. Necessary conditions for the existence of a continuously differentiable control law for asymptotically stabilizing $(x_0, 0)$ are:

- *The linearized system has no uncontrollable modes associated with eigenvalues with positive real part.*
- *There exist a neighborhood N of $(x_0, 0)$ such that for each $\xi \in N$, there exist a control $u_\xi(t)$ defined for all $t > 0$ that drives the solution of $\dot{x} = f(x, u_\xi))$ from the point $x = \xi$ at t=0 to $x = x_0$ at $t = \infty$.*
- *The mapping $\gamma : N \times \mathbb{R}^m \to \mathbb{R}^n$, N a neighborhood of the origin, defined by $\gamma : (x, u) \to f(x, u)$ should be onto an open set of the origin.*

Proof. The interested reader might consult the proof of this theorem in[5]. □

As a consequence of Brockett's theorem, asymptotic stabilization of equilibria cannot be achieved using static continuous state feedback control laws, see proposition 3 in[34]. This limitation motivates the use of different approaches to solve the control problem, i.e. using *open-loop/feedforward control*, where the input does not depend on the state q nor on the error e, or by using a non-smooth feedback control and/or time-varying law. The design of this type of feedback control algorithms involve more difficulties, however, they are desired in real-time control under uncertain or perturbed conditions[34].

3-2 Nonlinear Control Theory

This section attempts to provide some relevant lemmas and theorems involving nonlinear stability theory concepts for time-varying (non-autonomous) systems that will be important in evaluating the collective stability of the flocking algorithms. Since it involves many stability concepts for different type of systems, it is chosen to not explicitly discuss them in detail in this preliminary section but to provide the reader with well-known literature in this field and focus on the specific tools needed for the problem to be solved.

3-2-1 Lyapunov-Like Analysis - Barbalat's Lemma

The nonlinear control theory concepts that will be used for stability analysis of the flock can be consulted from[53;25] and comprise two lemmas, *Barbalat's Lemma* and *Lyapunov-like Lemma*, respectively.

Barbalat's lemma is a purely mathematical result concerning the asymptotic properties of functions and their derivatives. When properly used for dynamic systems, particularly non-autonomous systems, it may lead to the satisfactory solution of many asymptotic stability problems[53].

Lemma 1 (Barbalat's Lemma[53])**.** *If the differentiable function f(t) has a finite limit as* $t \to \infty$*, and if* $\dot{f}$ *is uniformlly continuous, then* $\dot{f}(t) \to 0$ *as* $t \to \infty$*.*

Proof. The interested reader might consult the proof of this theorem in[53]. □

Then, a function $f : \mathbb{R} \to \mathbb{R}$ is said to be uniformly continuous if $\forall \epsilon > 0$, it exists $\delta(\epsilon) > 0$ such that

$$\forall |t_2 - t_1| \leq \delta \to |f(t_2) - f(t_1)| \leq \epsilon$$

Uniform continuity of a function is often awkward to assert directly from the above definition[53]. Hence, a more convenient method is to evaluate the derivative of the function. A sufficient condition is then that, for a differentiable function, uniform continuity can be implied if the derivative is bounded.
To apply Barbalat's lemma to the analysis of dynamic systems, one typically uses the following immediate corollary, which looks very much like an invariant set theorem in Lyapunov analysis:

Lemma 2 (Lyapunov-like Lemma[53])**.** *If a scalar function V(x,t) satisfies the following conditions*

- *V(x,t) is lower bounded*
- $\dot{V}(x,t)$ *is negative semi-definite*
- $\dot{V}(x,t)$ *is uniformly continuous in time*

then $\dot{V}(x,t) \to 0$ *as* $t \to \infty$

An example is illustrated in[53] (see example 4.13), to illustrate the procedure of the asymptotic stability analysis for non-autonomous systems.

Remark 1. *[53] It is important to note that, such an analysis based on Barbalat's lemma, is called a Lyapunov-like analysis. However, it presents two subtle, but important differences with Lyapunov analysis*

(i) The function $V(t,x)$ *can simply be a lower bounded function of* x *and* t *instead of a positive definite function.*

(ii) The derivative $\dot{V}$ *must in addition to being negative or zero, also shown to be uniformly continuous. A sufficient condition is then that, for a differentiable function, uniform continuity can be implied if the derivative is bounded. Generically, this is done by proving boundedness of* $\ddot{V}$*.*

3-3 Communication Structure

The communication structure often emerges as a rational behaviour in multi-robot systems (MRSs) cooperating in an environment and serves as a means for the coordination problem. The communication can be seen as a mode of the interaction between the robots. This mode of interaction allows the robots to share information about their position, the environment and sensor data with other robots in the system or in case of limited communication with their neighboring robots (e.g. in flocking).
In fact, this interaction between robots allows individual robots to get information about the other robots in the system in the perspective of their intentions, goals and actions. The structure of the communication network in networked MRSs can generally be classified in the following way:

- *Communication topology*: denotes the "physical" interconnections among the agents in a cooperative group. The topology of the system can either be fixed or dynamic. The first indicates that the topology is static whereas the latter means that the interconnection structure can change arbitrarily[59]. Moreover, the interconnection structure can be directed or undirected, i.e. unidirectional or bidirectional, respectively.
- *Communication range*: the maximum range between two agents where communication is still possible. This can be divided into three classes of communication. The case when there is no direct communication between the robots, i.e. the robots are communicating indirect with each other by observing the behaviour of other robots. Local communication includes that robot can only communicate with other robots that are in the neighborhood. Then, global communication is where robots can communicate with any robot in the system.
- *Communication bandwidth*: the amount of data that a communication link can transmit during a given time span.

In flocking algorithms, locality of communication and interaction is involved since the robots in the system only interact with neighboring robots in proximity in a dynamic topology within a limited sensing field. This is illustrated in Figure 3-1. The studies that require global information may suffer from lack of scalability, but allow more accurate forming of a wider range of formations. On the other hand, studies using only local communication and sensor data tend to be more scalable, more robust, and easier to build, however, they are also limited in variety and precision of formations[24].

3-4 Graph Theory

Graph theory is widely used as a theoretical basis for establishing communication and control structures in decentralized swarm robotics (SR) systems, as this makes it possible to abstract away the complex sensing and communication characteristics of individual robots so that research efforts can focus on the underlying interaction topologies that lead to the desired global behaviors. Consensus algorithms are often used to enable individual units in SR systems to reach a common perspective of objectives and state of the world so they can agree on the direction of the movement, where to meet or the location of the intruders. Consensus algorithms have proved to work well in combination with graph theory and are of significant importance in flocking algorithms[2].

3-4-1 Formation Control Graphs

The theoretical framework that will be discussed in this section relies on some basic concepts in graph theory, algebraic graph theory and spatially induced graphs (i.e. proximity nets), see ([41] and its references [43], [45-48]).

Topology of Flocks: Proximity Nets

Typically, a graph is used that characterizes the interaction network between the agents in the flock. A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a pair that consists of a set of nodes/vertices $\mathcal{V} = \{1, \dots, N\}$ representing the agents and an edge set $\mathcal{E} \subseteq \{(i, j) : i, j \in \mathcal{V}, j \neq i\}$ denoting the links between the agents to share information (the graph is generally directed and has no self-loops)[41]. Moreover, $|\mathcal{V}|$ and $|\mathcal{E}|$ denote the *order* and *size* of the graph, respectively. In networked dynamic systems, the quantity $|\mathcal{E}|$ is called the *communication* complexity[26].
For a directed graph, an edge $(j, i) \in \mathcal{E}$ specifies that node i has the capability to obtain information from node j, which is not necessarily vice versa. In this interaction setting, the node j is a neighbor of node i. A directed path in a directed graph is an ordered sequence of edges of the form (i_1, i_2),$(i_2, i_3), \dots$, where $i_j \in \mathcal{V}$. A subgraph of $\mathcal{G}$ denotes a node and edge set that are subsets of those

in $\mathcal{G}$. The *adjacency matrix* of a graph[16], denoted by $A = [a_{ij}]$, is a matrix that has nonzero elements satisfying the property of $a_{ij} \neq 0 \iff (i,j) \in \mathcal{E}$. Furthermore, the graph is weighted if it has also other elements than just 0-1 elements. In general, in the flocking phenomenon, undirected weighted graphs with position-dependent adjacency components are exploited. Note that an undirected graph results in a symmetric adjacency matrix A, i.e. $A^T = A$. More formally, in an undirected graph, it holds that $(i,j) \in \mathcal{E} \iff (j,i) \in \mathcal{E}$. The *neighbor set* of node i can be defined as[41]

$$\mathcal{N}_i = \{j \in \mathcal{V} : a_{ij} \neq 0\} = \{j \in \mathcal{V} : (i,j) \in \mathcal{E}\} \tag{3-7}$$

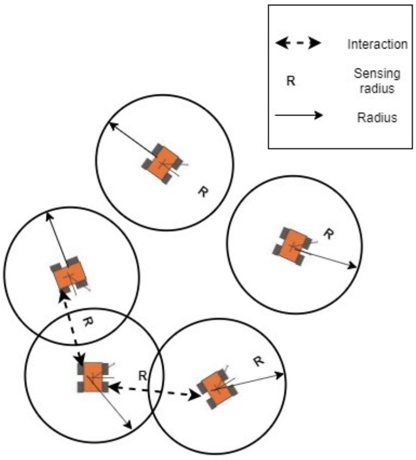


Figure 3-1: Proximity communication structure in flocking algorithms. The robots in the MRSs involve locality of communication and interaction, i.e. robots only communicate with neighboring robots within the limited sensing radius R.

Then, define $q_i \in \mathbb{R}^m$ to be the position of node i $\forall i \in \mathcal{V}$. The configuration of all nodes of the graph $\mathcal{G}$ can be described as a vector $q = col(q_1, \dots, q_n) \in \mathcal{Q} = \mathbb{R}^{mn}$. A *framework* is a pair $(\mathcal{G}, q)$ that consists of a graph and the configuration of its nodes[41].
The interaction range between agents is denoted by $R > 0$. The set of *spatial neighbors* of agent i, see Figure 3-1, is defined by

$$\mathcal{N}_i = \{j \ \in \mathcal{V} : ||q_j - q_i|| < R\} \tag{3-8}$$

with $||.||$ the Euclidian norm in $\mathbb{R}^m$. The *proximity net* $\mathcal{G}(q) = (\mathcal{V}, \mathcal{E}(q))$ is defined by the set of vertices $\mathcal{V}$ and the set of edges[41]

$$\mathcal{E}(q) = \{(i,j) \in \mathcal{V} \times \mathcal{V} : ||q_j - q_i|| < R, i \neq j\} \tag{3-9}$$

which has position-dependency on q. Then, the framework $\mathcal{G}(q), q$ is called the *proximity structure*[41]. In fact, note that an example of a proximity net is the topology of a wireless sensor network with a radio range denoted by R. The proximity net $\mathcal{G}(q)$ will be undirected if the interaction range is identical for all agents in the networked system.

Consensus on Proximity Nets & Graph Laplacians

The Laplacian of a graph[16] is used for analysis of velocity vector matching of agents in the flock structure. The *degree matrix* of the graph $\mathcal{G}$ is a diagonal matrix $\Delta = \Delta(A)$ with diagonal elements $\sum_{j=1}^{n} a_{ij}$, i.e. column sums of the adjacency matrix A[41]. Furthermore, the graph *Laplacian* $L = [l_{ij}]$ is an $n \times n$ matrix that is related to the graph $\mathcal{G}$ and described by

$$L = \Delta(A) - A \tag{3-10}$$

The Laplacian matrix L has the property that the right eigenvector $\mathbf{1}_n = (1, \dots, 1)^T$ is always associated with an eigenvalue $\lambda_1 = 0$. Some relevant properties of the graph Laplacian L are summarized in the folowing lemma:

Lemma 3 (Properties of graph Laplacian). *Let $\mathcal{G}= \mathcal{V}, \mathcal{E}$ be an undirected graph of order n with a nonnegative adjacency matrix $A = A^T$. Then, the following statements hold*[16]

- *L is a positive semi-definite matrix that satisfies the following sum-of-squares (SOS) property:*

$$z^T Lz = \frac{1}{2} \sum_{(i,j)\in\mathcal{E}} a_{ij}(z_j - z_i)^2, \qquad z \in \mathbb{R}^n \tag{3-11}$$

- *The graph $\mathcal{G}$ has $c \geq 1$ connected components iff rank(L) = n-c. Particularly, $\mathcal{G}$ is connected iff rank(L) = n-1.*

- *Let $\mathcal{G}$ be a connected graph, then*

$$\lambda_2(L) = \min_{z\perp \mathbf{1}_n} \frac{z^T Lz}{||z||^2} > 0 \tag{3-12}$$

Proof. The statements of the above lemma are well-known results in the domain of algebraic graph theory and the proofs can be consulted from[16]. □

The eigenvalue $\lambda_2(L)$ is also known as the *algebraic connectivity* of a graph[16]. The speed of convergence of a linear consensus protocol is equal to $\lambda_2 > 0$[41]. The consensus protocol is relevant for synchronization purposes in flocking algorithms and results in *velocity matching*. The multi-dimensional graph Laplacian is denoted by

$$\hat{L} = L \otimes I_m$$

and it satisfies the SOS property defined as

$$z^T \hat{L}z = \frac{1}{2} \sum_{(i,j)\in\mathcal{E}} a_{ij}(z_j - z_i)^2, \qquad z \in \mathbb{R}^{mn} \tag{3-13}$$

with $z = col(z_1, z_2, \ldots, z_n)$ ajd $z_i \in \mathbb{R}^m \ \forall i$. Note that the SOS property also holds for the proximity net $\mathcal{G}(q)$[41].

3-5 Mathematical Definitions

This section discusses some important mathematical concepts and notations that will be used in the theoretical analysis of the flocking algorithms.

3-5-1 Euclidean Norm ||.||

The Euclidean norm of a vector x is represented by $||x||_2 = \sqrt{x_1^2 + x_2^2 + \ldots + x_N^2}$, where $x = [x_1, x_2, \ldots, x_N]^T$, a column vector. The norm is a scalar value. Note that a derivative of a scalar with respect to the vector x must result in a vector. To estimate the derivative of a scalar with respect to a vector, we estimate the partial derivatives of the scalar with respect to each element in the vector and arrange the partial derivatives to form a vector. As a result, we get

$$\Delta_x ||x||_2 = [\frac{\partial}{\partial x_1}||x||_2, \frac{\partial}{\partial x_2}||x||_2, \ldots, \frac{\partial}{\partial x_N}||x||_2]^T \tag{3-14}$$

The $i-th$ component of the derivative is given by

$$\frac{\partial}{\partial x_i}||x|| = \frac{\partial}{\partial x_i}(\sqrt{x_1^2 + x_2^2 + \ldots x_N^2}) = \frac{1}{2}\frac{2x_i}{(x_1^2 + x_2^2 + \ldots + x_N^2)} = \frac{x_i}{(x_1^2 + x_2^2 + \ldots + x_N^2)} \tag{3-15}$$

By arranging all the partial derivatives together, we obtain that the derivative of the Euclidean norm is given by[46]

$$\Delta_x ||x||_2 = \frac{x}{||x||_2} \tag{3-16}$$

3-5-2 Properties/Facts of Norms of Vectors and Matrices

Let $||.||$ denote the norm of a vector or matrix without loss of generality, since matrix and vector norms are equivalent, we consider the 2-norm, i.e.

$$||x|| = |x^T x| = \sqrt{\sum_{i=1}^{n} x_i^2}, \quad x \in \mathbb{R}^n \tag{3-17}$$

and

$$||A|| = \sigma_{max}(A) = \sqrt{\lambda_{max}(A^T A)}, \quad A \in \mathbb{R}^{n \times n} \tag{3-18}$$

where σ_{max} denotes the largest singular value and λ_{max} denotes the largest eigenvalue of the corresponding matrix, respectively. If A is symmetric, then

$$||A|| = max\{|\lambda_1|, |\lambda_2|, \ldots, |\lambda_n|\} \tag{3-19}$$

Furthermore, the following properties hold for vectors and matrices

- $||AB|| \leq ||A||\ ||B||$, for any A and B of appropriate dimension.
- $||x + y|| \leq ||x|| + ||y||$, for any vectors x and y (*triangle inequality*)
- The scalar $ax + by$= $[ab]\ [xy]^T \leq ||[ab]||\ ||[xy]^T$
- The scalar $|xy| \leq cx^2 + \frac{1}{4c}y^2$, for any $c > 0$
- $x^T Px \leq ||x^T Px|| \leq ||P||\ ||x||^2 = \lambda_{max}(P)||x||^2$, for any vector x and positive semi-definite matrix $P = P^T$, where λ_{max} is the largest eigenvalue of P.
- $x^T Px \geq \lambda_{min}(P)||x||^2$, for any vector x and positive semi-definite matrix $P = P^T$, where λ_{min} is the smallest eigenvalue of P.

3-5-3 Notations

The sets of the reals and $N \times N$ matrices are denoted by $\mathbb{R}$ and $\mathbb{R}^{N \times N}$, respectively. Moreover, let $\mathbf{1}_N$ ($\mathbf{0}_N$) denote the $N \times 1$ column vector of all ones (zeros). Throughout this study, we use $||.||$ and $\otimes$ to denote the Euclidean norm and Kronecker product, respectively. For a real symmetric matrix, $\lambda_{min}(.)$ and $\lambda_{max}(.)$ denote its minimum and maximum eigenvalues, respectively. Furthermore, I_m is used to denote the $m \times m$ identity matrix. Then, for a vector function $x(t)$: $\mathbb{R} \mapsto \mathbb{R}^n$, it is said that $x(t) \in \mathcal{L}_p$ if $(\int_0^\infty ||x(t)||^p\ dt)^{\frac{1}{p}} < \infty$ and $x(t) \in \mathcal{L}_\infty$ if for each element of $x(t)$, noted as $x_i(t)$, $\sup\limits_{t \geq 0} |x_i(t)| < \infty$, $i = 1, \ldots, n$.

Chapter 4

Flocking Algorithm for Formation Control

Problem Formulation & Preliminary Results

This chapter introduces the problem formulation through a mathematical framework and discusses some preliminary results and definitions concerning the mathematical modelling of the dynamic non-holonomic mobile multi-agent system (MAS), proximity network communication architecture and the general structure of the flocking control algorithm to be designed later.

4-1 Non-Holonomic Networked Euler-Lagrange Multi-Agent System

Consider a group of N homogeneous non-holonomic Euler-Lagrange (EL) agents subject to holonomic and non-holonomic constraints, respectively. The EL framework is used to derive the motion dynamics of the constrained mechanical system[34;19]. Generically, the agents are mathematically governed by the following EL dynamics

$$\begin{cases} \mathcal{M}_i(q_i)\ddot{q}_i + \mathcal{V}_i(q_i,\dot{q}_i)\dot{q}_i + \mathcal{G}_i(q_i) + \mathcal{B}_i(q_i)\tau_d = \mathcal{B}_i(q_i)\tau_i - \mathcal{A}_i^T(q_i)\lambda_i, & i = 1,\dots,N \\ \mathcal{A}_i(q_i)\dot{q}_i = 0 \end{cases} \tag{4-1}$$

where $q_i \in \mathbb{R}^n$ denotes the generalized coordinates vector, $\mathcal{M}_i(q_i) \in \mathbb{R}^{n\times n}$ is a symmetric positive definite inertia matrix, $\mathcal{V}_i(q_i,\dot{q}_i) \in \mathbb{R}^{n\times n}$ is the centripetal and coriolis matrix, $\mathcal{G}_i(q_i) \in \mathbb{R}^n$ is the gravitational vector, $\mathcal{B}_i(q_i)\tau_i \in \mathbb{R}^n$ is the generalized force vector with $\mathcal{B}_i(q_i) \in \mathbb{R}^{n\times p}$ the input matrix mapping the external inputs $\tau_i \in \mathbb{R}^p$, where $p = n - k$, to forces and torques that execute work on the generalized coordinates q_i.
Then, $\tau_i \in \mathbb{R}^p$ denotes the input vector of external torques and forces provided by the actuators. Moreover, $\tau_d \in \mathbb{R}^p$ denotes the bounded and unknown disturbances including unstructured uncertainties (or unmodelled dynamics), $\mathcal{A}_i^T(q_i)\lambda_i \in \mathbb{R}^n$ is the set of generalized constraint forces where $\mathcal{A}_i(q_i) \in \mathbb{R}^{m\times n}$ and $\lambda_i \in \mathbb{R}^m$ denote the constraint matrix and the vector of unknown Lagrange multipliers, respectively.
It is assumed that we have k holonomic constraints and $m-k$ non-holonomic independent constraints and they can all be written in the form of $\mathcal{A}_i(q_i)\dot{q}_i = 0$ with $\mathcal{A}_i(q_i)$ being a full-rank matrix. As a consequence, it is always possible to find a full rank matrix $\mathcal{S}_i(q_i) = [s_1(q)\dots s_{n-m}(q)] \in \mathbb{R}^{n\times m}$, that is formed by a set of smooth and linearly independent vector fields spanning the null space of $\mathcal{A}_i(q_i)$, such that $\mathcal{A}_i(q_i)\mathcal{S}_i(q_i) = 0$[34;39]. The dynamics of the reduced dynamic model of the non-holonomic mobile EL agents, that will be discussed next in Subsection 4-1-2, satisfies the following properties[53;14;15]:

Property 4-1.1. *$\mathcal{M}_i(q_i)$ is symmetric and positive definite, $\forall q_i$.*

Property 4-1.2. *There exist positive constants $k_{\underline{\mathcal{M}}}$, $k_{\overline{\mathcal{M}}}$, $k_{\overline{\mathcal{V}}}$, $k_{\overline{\mathcal{G}}}$, such that $k_{\underline{\mathcal{M}}} I_n \leq \mathcal{M}_i(q_i) \leq k_{\overline{\mathcal{M}}} I_n$, $||\mathcal{V}_i(q_i, \dot{q}_i)|| \leq k_{\overline{\mathcal{V}}} ||\dot{q}_i||$ and $||\mathcal{G}_i(q_i)|| \leq k_{\overline{\mathcal{G}}}$, $\forall q_i$.*

Property 4-1.3. *$\dot{\mathcal{M}}_i(q_i)$ - $2\mathcal{V}_i(\dot{q}_i, q_i)$ is skew symmetric.*

Remark 2. *In nonlinear control design, exploitation of the physical properties can provide relevant insights into facilitating the control design for complex nonlinear plants to a more simple issue, or may easily solve an otherwise intractable design problem.*

Remark 3. *As a complement to Property 4-1.3, an interesting fact is that a quadratic function associated with a skew-symmetric matrix with zero trace is always zero. This is an essential property. Note that, in principle, a matrix can be skew-symmetric, yet its associated quadratic form is not zero due to the aforementioned reason. More formally, skew-symmetry is a property which can be interpreted as a matrix expression reflecting conservation of energy*[53]*.*

4-1-1 Dynamic Reduction Procedure

The non-holonomic EL system equations can be transformed into an equivalent form to obtain a reduced dynamic model that is more appropriate for control purposes and numerical simulation studies[34;19]. It is possible to define $n-m$ velocities $v_{p,i} = [v_{p,i}(1) \ldots v_{p,i}(n-m)]$ such that the kinematic model of the system is given by

$$\dot{q}_i = \mathcal{S}_i(q_i) v_{p,i}(t) \tag{4-2}$$

The main objective is to eliminate the unknown Lagrange multipliers $\lambda_i \in \mathbb{R}^m$ by using the kinematics matrix $\mathcal{S}_i(q_i)$ in (4-2) of the system as described in Section 4-1 and Subsection 3-1-1. The unknown Lagrange multipliers are usually functions of the generalized coordinates and time, and are obtained as an element of the solution[19].
Then, as in[34], a method is proposed to eliminate the Lagrange multipliers by exploiting the properties of the nullspace of the constraint matrix $\mathcal{A}_i(q_i)$ and obtaining a reduced dynamic model to facilitate the theoretical analysis of the system and control design:

- Eliminating the unknown Lagrange multipliers λ_i by pre-multiplying the dynamic EL model (4-1) with $\mathcal{S}_i^T$ from the left such that $\mathcal{S}_i^T \mathcal{A}^T = 0$.
- Then, differentiating (4-2) and substituting it into (4-1).

As a result, the dynamical equations of the reduced dynamic model are obtained by

$$\begin{aligned} &\widetilde{\mathcal{M}}_i \dot{v}_{p,i} + \widetilde{\mathcal{V}}_i v_{p,i} + \widetilde{\mathcal{G}}_i + \widetilde{\mathcal{B}}_i \tau_{di} = \widetilde{\mathcal{B}}_i \tau_i \\ &\text{with} \end{aligned} \tag{4-3}$$

$$\begin{aligned} \widetilde{\mathcal{M}}_i &= \mathcal{S}_i^T \mathcal{M}_i \mathcal{S}_i \\ \widetilde{\mathcal{V}}_i &= \mathcal{S}_i^T \mathcal{M}_i \dot{\mathcal{S}}_i + \mathcal{S}_i^T \mathcal{V}_i \mathcal{S}_i \\ \widetilde{\mathcal{G}}_i &= \mathcal{S}_i^T \mathcal{G}_i \\ \widetilde{\mathcal{B}}_i &= \mathcal{S}_i^T \mathcal{B}_i \end{aligned} \tag{4-4}$$

where $v_{p,i} = [v_i, \omega_i]^T \in \mathbb{R}^{n-m}$ is the vector of pseudo-velocities and $\tau_i \in \mathbb{R}^p$ denotes the flocking control algorithm for formation control purposes that will be designed later in Chapter 5 and is of the form of (4-14).

Assumption 1. [34] *It is assumed that $\det[\mathcal{S}_i^T \mathcal{B}_i] \neq 0$, which is a realistic hypothesis in most practical cases. This assumption is of fundamental importance for the design and implementation of the flocking control algorithm for non-holonomic networked EL multi-agent systems, which will be evident later.*

Remark 4. *It is important to note that, although some information might be lost in the reduced dynamical model, this simplification of the dynamic model comes with minor losses of accuracy and facilitates the design of control algorithms*[34;19]*. Note that more accurate or more advanced models are not always better in performance, because they may require unnecessarily complex control design and analysis and more demanding computation*[53]*. The key here is to keep "essential" effects and discard insignificant effects in the system dynamics in the operating range of interest.*

The kinematic and dynamical model can conveniently be merged into an $(n+m)$-dimensional reduced state-space model

$$\begin{cases} \dot{q}_i = \mathcal{S}_i v_{p,i} \\ \dot{v}_{p,i} = [\widetilde{\mathcal{M}}_i]^{-1}(\widetilde{\mathcal{B}}_i \tau_i - \widetilde{\mathcal{V}}_i v_{p,i}) \end{cases} \tag{4-5}$$

4-1-2 Modelling of a Networked Non-Nolonomic DDWMR System

A detailed and accurate dynamic model is developed to describe the motion of the system that will be used in the theoretical analysis of the flocking formation control algorithm and numerical simulation studies. The mathematical dynamic modelling of differential-drive WMRs, taking the non-holonomic constraints into consideration, is evaluated by exploiting the EL methodology [34;19]. The configuration of a non-holonomic differential-drive wheeled mobile robot (WMR) is represented in Figure 4-1.
A differential-drive WMR is a mobile robot where the motion consists of two separately driven fixed wheels by two independent actuators (DC motors) [39]. The center of mass of the mobile robot is located at C. The local coordinate frame (A, x_r, y_r) is body-fixed to the robot at the actuation point A, which is the intersection of the wheel's axis of rotation and the axis of symmetry of the robot's body, respectively. This point is located at a distance d from C. In addition, R and L represent the radius of each wheel and the half distance between two wheels. The robot can change its orientation by varying the relative rate of rotation between the left- and right wheel, respectively. The passive castor wheel has the ability to rotate in any direction and is placed in front of the mobile robot to preserve its equilibrium. Therefore, an additional steering motion is not required for the system.
In contrast to most of the other flocking algorithms that consider particle agents, as discussed in Chapter 1, this work considers a multi-robot system of non-holonomic EL modelled mobile agents that are isomorphic to a rigid body with a fixed circular area denoting the proximity radius R_s in which the proximity neighbors can communicate, as illustrated in Figure 4-2. Furthermore, compared with the existing work, that assume the mobile robots are points, we also consider the effect of the robot actuation (i.e. the actuation point A has a skewing d to te center of mass C, see Figure 4-1).

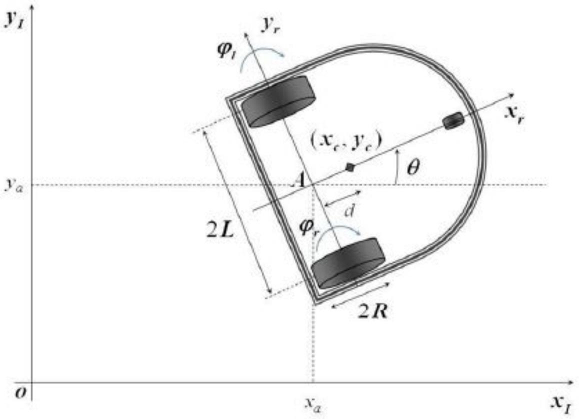


Figure 4-1: Scheme of a differential-drive WMR[19].

The configuration of the differential-drive WMR can be described by using the following generalized coordinates

$$q_i = [x_{i,C}\,, y_{i,C}\,, \theta_i, \phi_{i,R}\,, \phi_{i,L}\,]^T$$

where $(x_{i,C}, y_{i,C})$ denote the coordinates of the center of mass C of mobile robot i in the global coordinate frame at (O, x_I, y_I), θ_i is the orientation angle and $\phi_{i,R}, \phi_{i,L}$ denote the angular positions of the left- and right driving wheels, respectively. The kinematic modelling of the robotic system requires the consideration of the following assumptions[39;19]

- *No lateral slip motion*; the definition of this constraint is simply that the robot is not able to move in a sideward direction. As a result, the WMR can only move in a curved motion. This can simply mathematically be formulated for point A in the robot frame as

  $$\dot{y}^r_{A,i} = 0$$

  i.e., the point A has no sideway velocity, in order to maintain the no skidding condition.

- *Pure rolling*; the pure rolling constraints represents the fact that each wheel maintains one contact point with the ground. More formally, there is no slipping of the wheels in the x_r axis and no skidding in the y_r axis.

The following constraint equations can mathematically be concluded from the above assumptions

$$\begin{cases} \dot{y}_{i,C}\cos\theta_i - \dot{x}_{i,C}\sin\theta_i - d\dot{\theta}_i = 0 \\ \dot{x}_{i,C}\cos\theta_i + \dot{y}_{i,C}\sin\theta_i + L\dot{\theta}_i = R\dot{\phi}_{i,R} \\ \dot{x}_{i,C}\cos\theta_i + \dot{y}_{i,C}\sin\theta_i - L\dot{\theta}_i = R\dot{\phi}_{i,L} \end{cases} \tag{4-6}$$

The first constraint in (4-6) is non-integrable and thus considered as the non-holonomic constraint[39]. Then, note that the constraint equations can be represented as in (4-1) by the following constraint matrix form

$$\mathcal{A}_i(q_i)\dot{q}_i = \underbrace{\begin{bmatrix} -\sin\theta_i & \cos\theta_i & -d & 0 & 0 \\ -\cos\theta_i & -\sin\theta_i & -L & R & 0 \\ -\cos\theta_i & -\sin\theta_i & L & 0 & R \end{bmatrix}}_{\mathcal{A}_i(q_i)} \underbrace{\begin{bmatrix} \dot{x}_{i,C} \\ \dot{y}_{i,C} \\ \dot{\theta}_i \\ \dot{\phi}_{i,R} \\ \dot{\phi}_{i,L} \end{bmatrix}}_{\dot{q}_i} = 0$$

Then, by (4-2), and defining the pseudo-velocities vector as $v_{p,i} = [v_{A,i}, \omega_i]$, where $v_{A,i}$ and ω_i are the linear velocity of the robot actuation point A and the angular velocity of the differential-drive WMR i, respectively, the following so-called kinematics model of the considered non-holonomic mobile agent can be formulated[39]

$$\dot{q}_i = \mathcal{S}_i(q_i)v_{p,i}(t) = \underbrace{\begin{bmatrix} \cos\theta_i & -d\sin\theta_i \\ \sin\theta_i & d\cos\theta_i \\ 0 & 1 \\ \frac{1}{R} & \frac{L}{R} \\ \frac{1}{R} & -\frac{L}{R} \end{bmatrix}}_{\mathcal{S}_i(q_i)} \underbrace{\begin{bmatrix} v_{A,i} \\ \omega_i \end{bmatrix}}_{v_{p,i}(t)} \tag{4-7}$$

Modelling of a Differential-Drive WMR The mathematical modelling of a constrained mechanical system is important to study the motion. Dynamic modelling takes into consideration the forces that influence the motion whereas kinematic modelling does not account for these forces. The dynamic model of a differential-drive WMR is essential for simulation analysis and for the design of various motion control algorithms.
By exploiting the relationship between the robot actuation point A and the mass center C, and

by neglecting the angular positions of the driving wheels ($\phi_{i,R}$ and $\phi_{i,L}$) the kinematic model can alternatively be written into the following well-known kinematic model

$$\begin{bmatrix} \dot{x}_{A,i} \\ \dot{y}_{A,i} \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \cos\theta_i & 0 \\ \sin\theta_i & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{A,i} \\ \omega_i \end{bmatrix}$$

The EL framework is used to derive the dynamics of the wheeled mobile robot. The dynamical model of a non-holonomic differential-drive WMR can be described by the following equations of motion

$$\begin{cases} \mathcal{M}_i(q_i)\ddot{q}_i + \mathcal{V}_i(q_i,\dot{q}_i)\dot{q}_i = \mathcal{B}_i(q_i)\tau_i - \mathcal{A}_i^T(q_i)\lambda_i \\ \mathcal{A}_i(q_i)\dot{q}_i = 0 \end{cases}$$

by assuming no surface friction, no bounded unknown disturbances and unstructured uncertainties, and no gravitational influences, since we are dealing with wheeled mobile robots (WMRs) moving in a 2D plane, we can cancel out and neglect some terms present in the general mathematical formulation in (4-1).

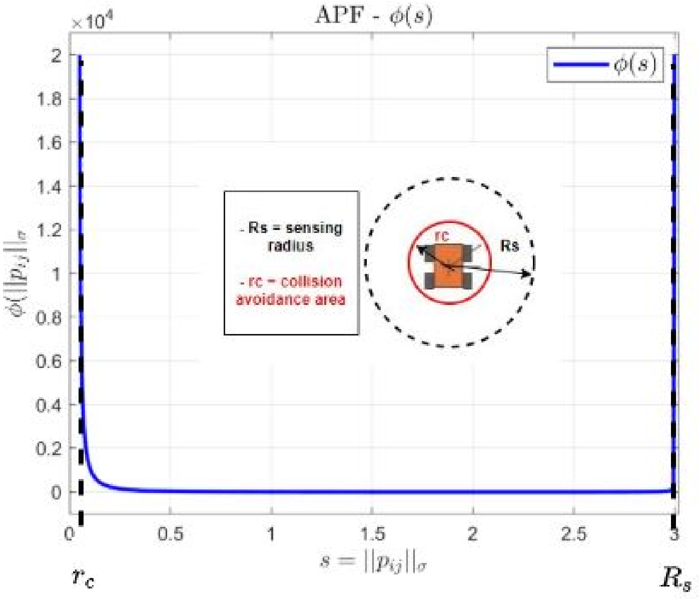


Figure 4-2: The non-holonomic EL agents are isomorphic to a rigid body with specific dimensions and a fixed circular area R_s denoting the sensing radius. The r_c can be intuitively interpreted as the avoidance "comfort zone" of the mobile robot. The artificial potential function (APF) is also illustrated alongside the agent setup for intuitive and explanatory purposes, see Chapter 5 for more details on the APF.

Dynamic Reduced Model The non-holonomic EL system equations can be transformed into an equivalent form to obtain a model that is more appropriate for control purposes and simulations, since we are only interested in the kinetic motion of the system. The main objective is to eliminate the unknown Lagrange multipliers $\lambda_i \in \mathbb{R}^m$ by using the kinematics matrix $\mathcal{S}_i(q_i)$ in (4-7) of the system. By following the dynamic reduction procedure described in Subsection 4-1-1, the corresponding reduced dynamic model is obtained

$$\widetilde{\mathcal{M}}_i(q_i)\dot{v}_{p,i} + \widetilde{\mathcal{V}}_i(q_i,\dot{q}_i)v_{p,i} = \widetilde{\mathcal{B}}_i(q_i)\tau_i \tag{4-8}$$

with

$$\widetilde{\mathcal{M}}_i(q_i) = \mathcal{S}_i(q_i)^T\mathcal{M}_i(q_i)\mathcal{S}_i(q_i)$$

$$\widetilde{\mathcal{V}}_i(q_i,\dot{q}_i) = \mathcal{S}_i(q_i)^T\mathcal{M}_i(q_i)\dot{\mathcal{S}}_{\rangle}(q_i) + \mathcal{S}_i(q_i)^T\mathcal{V}_i(q_i,\dot{q}_i)\mathcal{S}_i(q_i)$$

$$\widetilde{\mathcal{B}}_i(q_i) = \mathcal{S}_i(q_i)^T\mathcal{B}_i(q_i)$$

with $v_{p,i} = \, [v_{A,i}\,, \omega_i]^T$ known as the pseudo-velocities vector. The system matrices of the non-holonomic EL reduced dynamic differential-drive WMR model are governed by

$$\widetilde{\mathcal{M}}_i(q_i) = \begin{bmatrix} m + \frac{2I_w}{R^2} & 0 \\ 0 & I + md^2 + \frac{2I_w L^2}{R^2} \end{bmatrix}, \quad \widetilde{\mathcal{V}}_i(q_i, \dot{q}_i) = \begin{bmatrix} 0 & -md\omega_i \\ md\omega_i & 0 \end{bmatrix} \tag{4-9}$$
$$\widetilde{\mathcal{B}}_i(q_i) = \frac{1}{R}\begin{bmatrix} 1 & 1 \\ L & -L \end{bmatrix}, \quad \widetilde{\mathcal{G}}_i(q_i) = 0$$

where m denotes the total mass of the mobile robot (i.e. platform and driving wheels), I is the total moment of inertia of the system about a vertical axis passing through the center of mass C and I_w is the moment of inertia of both the right and left driving wheel about the axis of rotation of the wheels, respectively. Then, we can conveniently formulate the $(n + m)$-dimensional reduced state-space dynamical model of the differential-drive WMR as

$$\begin{cases} \dot{q}_i = \mathcal{S}_i(q_i) v_{p,i} \\ \widetilde{\mathcal{M}}_i(q_i)\dot{v}_{p,i} + \widetilde{\mathcal{V}}_i(q_i, \dot{q}_i) v_{p,i} = \widetilde{\mathcal{B}}_i(q_i)\tau_i \end{cases} \tag{4-10}$$

Remark 5. [39] *The control of differential-drive WMRs is a challenging research area because of the fact that these systems typically are under-actuated nonlinear systems subject to non-holonomic constraints. From a theoretical perspective, the uncontrollability of the linearized models of such systems cause the necessary conditions of the Brockett's theorem (see Theorem 1) to the existence of stabilizing smooth state feedback laws not satisfied*[39;34]*. The key feature of many of the problems that arise for a large class of under-actuated systems is the existence of nonlinear coupling between the actuated and under-actuated degrees of freedom*[68]*.*

4-2 Proximity Network Architecture

Consider a network of N non-holonomic EL mobile agents with dynamical model (4-10). In this study, it is assumed that the dynamic interaction topology that defines the neighbor relationship among the agents is dependent on their relative maximum distance. Therefore, the graph of the flocking algorithm is a *proximity graph*. In the flocking framework, typically, weighted undirected graphs with position-dependent adjacency components are exploited. Note that an undirected graph results in a symmetric adjacency matrix A, i.e. $A^T = A$.

More formally, in an undirected graph, it holds that $(i, j) \in \mathcal{E} \iff (j, i) \in \mathcal{E}$. The adjacency matrix $A(t) = [a_{ij}(t)] \in \mathbb{R}^{N\times N}$ of the graph $\mathcal{G}(t)$ is defined such that $a_{ij}(t) = [\ ||p_{ij}||_\sigma] > 0$ if $(j, i) \in \mathcal{E}(t)$ and $a_{ij}(t) = 0$ otherwise. The definition of $||p_{ij}||_\sigma$ is given in Definition 4-5, to obtain a smooth adjacency matrix with position-dependent adjacency elements $a_{ij}(t)$. It is important to note that this is an effective and convenient choice to the control of the system. Since the adjacency elements smoothly depend on the relative position between the agents in the proximity network, agents further away from each other get more penalized in the flocking control algorithm to obtain the desired result. This motivation adds some kind of autonomy to the choice of control gains for the control of the flock, which will be more evident in Chapter 5. For an undirected graph, it holds that $a_{ji} = a_{ij}$. The Laplacian matrix $\mathcal{L} = [l_{ij}] \in \mathbb{R}^{N\times N}$ associated with the adjacency matrix $A(t)$ is defined as $l_{ii} = \sum_{j\neq i} a_{ij}$ and $l_{ij} = -a_{ij}$, where $i \neq j$. It is important to note that the graph $\mathcal{G}(t)$ is undirected by definition where the Laplacian matrix $\mathcal{L}$ is symmetric positive semi-definite as a consequence.

The interaction range (i.e. sensing radius) between the agents is defined by $R_s > 0$ and the set of spatial neighbors is denoted by

$$\mathcal{N}_i(t) = \{j \in \mathcal{V} : ||p_j - p_i|| < R_s, j \neq i\} \tag{4-11}$$

where $||.||$ denotes the Euclidean norm, $p_i = [x_i, y_i]^T$ is the position of agent i, $\mathcal{N}_i$ the set of spatial neighbors of agent i and $\mathcal{V}$ the set of vertices. To obtain an algorithm with obstacle avoidance capabilities that has more practical relevance, no distinction is made between agents ($j = 1, \dots, N$) and obstacles ($j = N+1, , \dots, N_{obs}$) in the communication architecture. As a result, the set of spatial neighbors $\mathcal{N}_i$ involves both agents and static/dynamic obstacles, respectively. A distributed flocking control algorithm with obstacle avoidance capabilities with no distinction between neighboring agents and obstacles will be designed in Chapter 7. Furthermore, once the distance between the agents in the flock falls below a threshold $r_t \in (0, R_s)$, inter-agent and agent-obstacle collision avoidance maneuvers must be studied. In specific, this is mathematically formulated in Definition 5 and Lemma 10. The proximity net $G(q) = (\mathcal{V}, \mathcal{E}(q))$ is defined by the set of unordered vertices $\mathcal{V}$ and the set of unordered edges

$$\mathcal{E}(q) = \{(i,j) \in \mathcal{V} \times \mathcal{V} : ||q_j - q_i|| < R_s, i \neq j\} \tag{4-12}$$

The proximity graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ represents the communication network and is defined to be an undirected graph with a set of vertices $\mathcal{V} = \{1, 2, \dots, N\}$ that represents the mobile robots in the flock.

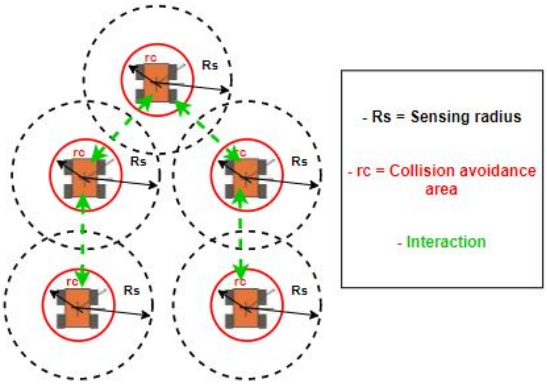


Figure 4-3: Proximity communication structure in the flocking algorithm. The mobile robots in the multi-agent network involve locality of communication and interaction, i.e. robots only communicate with neighboring robots within the proximity radius R_s.

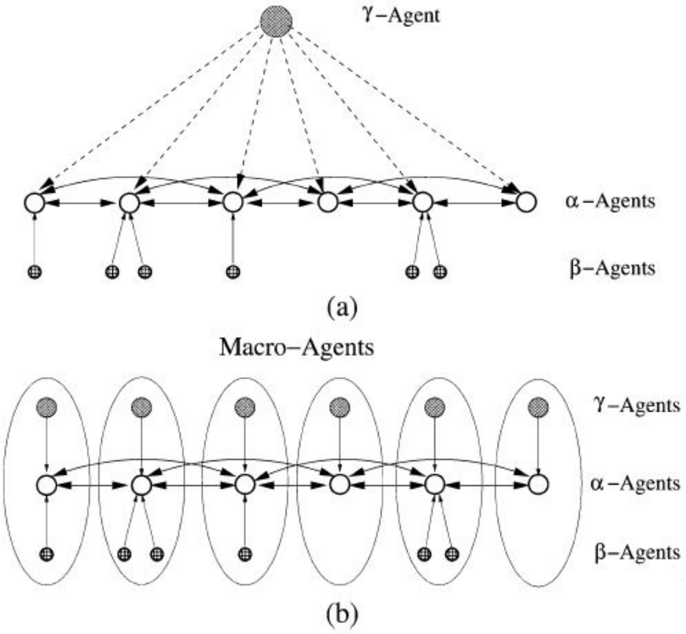


Figure 4-4: The virtual-leader follower information flow in constrained flocking. (a) Virtual-leader following hierarchical architecture. (b) Peer-to-peer architecture[41].

Assumption 2. *It is assumed that the robots are only equipped with limited on-board sensors, i.e. laser range finders (to detect obstacles and robots) without the ability to distinguish between them and a gyro sensor (to measure its position $[x_{A,i}, y_{A,i}]$ and orientation θ_i), respectively.*

Then, let $x_0(t) = [q_0(t), \dot{q}_0(t)]^T$ denote the virtual leader (in the set $\mathcal{V}_0$), which encapsulates a desired reference trajectory. The (virtual) leader-following proximity network is denoted as $\widetilde{\mathcal{G}}(t) = (\widetilde{\mathcal{V}}, \widetilde{\mathcal{E}}(t))$ with $\widetilde{\mathcal{V}} = \{0, 1, 2, \ldots, N\}$ where the node 0 is associated with the virtual leader. The set of edges that define the interaction network between the virtual leader and the mobile robots in the multi-robot system is defined as $\widetilde{\mathcal{E}} \subseteq \widetilde{\mathcal{V}} \times \widetilde{\mathcal{V}}$. It is assumed that the virtual leader has no neighboring agents and its motion is independent of the following robots. There exists a directed path between a virtual leader and the robot in proximity when there is a path from the leader to the follower. More formally, $(i, j) \in \widetilde{\mathcal{E}}(t)$ if and only if agent j can use the information of agent i for control. In specific, followers i, j are neighbors of each other if $||p_i - p_j|| < R_s$ and the leader is a neighbor of follower i if $||p_i - p_0|| < R_s$, respectively. As a result, the neighbor set of agent i in the (virtual) leader-following proximity network $\widetilde{\mathcal{G}}(t)$ at time t can be formulated as

$$\widetilde{\mathcal{N}_i}(t) = \{(j, i) \in \widetilde{\mathcal{E}}(t), j \neq i\} \tag{4-13}$$

In Chapter 6, a distributed (virtual) leader-following flocking control algorithm using this proximity architecture is designed. Evidently, note that the proximity graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ is a subgraph of $\widetilde{\mathcal{G}}(t)$ and it can be obtained by removing all the edges between the virtual leader in $\mathcal{V}_0$ and the nodes in $\widetilde{\mathcal{V}}$. Furthermore, the edge weights are defined as $a_{i0} = [\ ||p_{i0}||_\sigma] > 0$ if the leader is in proximity (i.e. a neighbor) of follower i, otherwise $a_{i0} = 0$. The leader-follower interaction topology matrix is defined as $\mathcal{H} = \mathcal{L} + \Lambda$, where $\Lambda = diag\{a_{10}, \ldots, a_{N0}\}$. It is evident that $\mathcal{H}$ is symmetric positive definite when there is at least one nonzero a_{i0}.
The visual interpretation of the communication topology is illustrated in Figure 4-3 to give the reader a more intuitive understanding of the proximity network concept. The communication structure often emerges as a rational behaviour in MRSs cooperating in an environment and serves as a means for the coordination problem. The communication can be seen as a mode of the interaction between the robots. This mode of interaction allows the robots to share information about their position, the environment and sensor data with other robots in the multi-robot system or in case of limited communication with their neighboring robots (e.g. in flocking). The interaction topology between the neighboring robots is dynamically changing with time. In general, proximity graphs might be the proper tool to mathematically handle the problem of a limited and time-varying communication topology among the robots. The following lemmas are important to obtain analytical insights into the flocking control algorithm, which will be used later in the theoretical analysis of the considered system.

Lemma 4 ([16]). *The eigenvalues of the Laplacian matrix $\mathcal{L}$ satisfy 0=$\lambda_1(\mathcal{L}) \leq \ldots \leq \lambda_N(\mathcal{L})$, and the algebraic connectivity $\lambda_2(\mathcal{L}) > 0$, if and only if the proximity graph $\mathcal{G}$ is connected.*

Lemma 5. *[16] The Laplacian matrix $\mathcal{L}$ of the proximity graph $\mathcal{G}$ has at least one zero eigenvalue associated with eigenvector $\mathbf{1} = (1, \ldots, 1)^T \mathbb{R}^N$, and all non-zero eigenvalues are positive. Furthermore, the Laplacian matrix $\mathcal{L}$ has a simple zero eigenvalue if and only if the proximity graph $\mathcal{G}$ is connected.*

Lemma 6 ([66;16]). *Suppose $\mathcal{G}$ to be an undirected graph of order N, and $\mathcal{G}_1$ be the undirected graph by adding some edge(s) into the graph $\mathcal{G}$. Then, $\lambda_i(\mathcal{L}_1) \geq \lambda_i(\mathcal{L})$, $\forall i = 1, 2, \ldots, N$, where $\mathcal{L}$ and $\mathcal{L}_1$ are the Laplacian matrices of $\mathcal{G}$ and $\mathcal{G}_1$, respectively.*

Lemma 7 ([15]). *If the leader has directed paths to all the following robots in the flock, the matrix $\mathcal{H}$ is symmetric positive definite.*

Lemma 8 ([15]). *Let $\mathcal{H}^a$ and $\mathcal{H}^b$ be the leader-follower topology matrix associated with the proximity graphs $\widetilde{\mathcal{G}}^a$ and $\widetilde{\mathcal{G}}^b$, respectively. If $\widetilde{\mathcal{G}}^a$ is a subgraph of $\widetilde{\mathcal{G}}^b$, then it holds that $\mathcal{H}^a \leq \mathcal{H}^b$.*

Proof. The proof can be consulted from [15]. □

4-3 Control Objective

The control and communication architecture are of fundamental importance in the design of scalable and robust flocking algorithms for swarm robotics (SR) systems. The ability to compute locally and share information has facilitated the development of multi-robot systems. The robustness of multi-robot systems is strictly related to the control and communication structure that is implemented to coordinate the robots into a desired formation.

SR systems must be scalable and robust to operate efficiently in dynamic and unpredictable environments. To be scalable, a SR system should be able to operate under a group size ranging from a small number to a big number of individuals. A common way of supporting scalability in the system is by exploiting distributed communication and control schemes. Such type of systems promise increased performance, efficiency and robustness, utilizing local coordinated sensing and actuation. A distributed dynamic feedback control algorithm of the following form is considered

$$\tau_i = h_i(u_i, q_i, \dot{q}_i), \ j \in \mathcal{N}_i(t) \quad i = 1, \dots, N \tag{4-14}$$

where h_i is a sufficiently smooth function to be designed later as control algorithm. To achieve cohesive flocking as defined below, the distributed control algorithm u_i has the following form

$$u_i = \alpha_i + \beta_i + \gamma_i \tag{4-15}$$

where $\alpha_i \in \mathbb{R}^p$, $\beta_i \in \mathbb{R}^p$ and $\gamma_i \in \mathbb{R}^p$ denote the connectivity-preserving APF-gradient based term to enforce inter-agent stabilization while avoiding inter-agent and agent-obstacle collisions, a velocity vector and orientation angle consensus term for alignment and a navigational (nonlinear) feedback term for flock guidance encapsulated in a virtual leader, respectively. This is graphically illustrated in Figure 4-5. The flock behavior inside the network is determined by the interplay between attraction and repulsive forces, respectively. There exists a distance that is called "equilibrium distance" in biology, in which attraction and repulsive forces between two agents are in balance[23]. Moreover, individual agents adjust their velocity vector and orientation according to those of their neighbors in proximity. More formally, a feedback mechanism is to be constructed based on the error between agents and their proximity neighbors. A consensus mechanism is such a feedback mechanism, that has been widely applied to a variety of multi-agent cooperative tasks[36;47].

In this work, being the first author, our major effort is to initialize and contribute to the design of a flocking control algorithm τ_i, $i = 1, \dots, N$, to make the non-holonomic networked EL dynamic system in (4-10) converge to a stable flocking motion while avoiding collisions and preserving the connectivity of the dynamic proximity network, respectively.

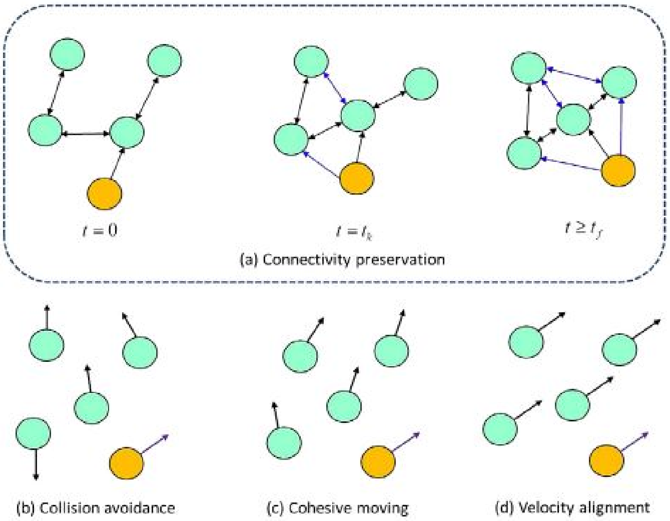


Figure 4-5: Graphical representation of connectivity-preserving flocking for explanatory purposes. The yellow and green circles represent the virtual leader and following robots, respectively. (a-d) describe the desired properties[14].

Definition 1 (Flocking)**.** *A group of non-holonomic networked Euler-Lagrange (EL) systems is said to be (asymptotically) flocking, when all the agents reach a consensus velocity vector and orientation, inter-agent stabilization is obtained by minimizing the APF and no collisions among the agents occur.*

Definition 2 (Graph connectivity)**.** *A proximity graph $\mathcal{G}(t)$ is connected at time instant t if there exists a path, i.e. a sequence of distinct vertices such that consecutive vertices are adjacent, between any two vertices in $\mathcal{G}(t)$.*

Definition 3 (Spanning tree)**.** *In the mathematical field of graph theory, a spanning tree of a graph is a subgraph that is a tree which includes all of the vertices of $\mathcal{G}(t)$, with a minimum possible number of edges. In general, a graph may have several spanning trees, but a graph that is not connected will not contain a spanning tree*[67]*. See also Figure 4-5a for illustration of a spanning tree with node 0 (the virtual leader) as root.*

Remark 6. *It is important to note that convergence and stability of the flocking algorithms strongly depend on the connectivity of the underlying communication topology that is time-varying. This will be more evident in the theoretical analysis exploiting concepts from nonlinear control theory.*

Remark 7. *Formally, proximity graphs are being exploited to include the local communication and local mobile robot interaction into the flocking algorithms. This is of significant importance for information sharing between the neighboring agents to ensure reliable and efficient coordination in the flocking control algorithm. Proximity graphs are viewed as a useful and decent mathematical tool to incorporate the time-varying communication topology of multi-robot systems (MRSs) in flocking algorithms. In addition, the convergence and stability of the flocking algorithms firmly depend on the connectivity of the communication network. Hence, the connectivity is of fundamental importance to explicitly be included in flocking algorithms as it allows the flock of robots to move as a stable cohesive unit.*

Remark 8. *The flocking control algorithms discussed are (fully) distributed in the sense that the algorithm relies on only local information interactions, i.e. each agent only uses its own position and velocity and the relative position and relative velocity between itself and its neighbors in proximity, respectively. In literature, this is commonly called as one-hop information*[15;7]*.*

Remark 9. *The cohesive flock guidance feature can be obtained by including the virtual leader in the neighboring set $\widetilde{\mathcal{N}}_i(t)$ of the multi-robot system. In addition, the hierarchical structure of the networked system, see Figure 4-4a, is useful in understanding the reason of a virtual leader (γ-agent) playing the role of a unifying objective bringing all the non-holonomic agents together to assemble a connected network of mobile robot agents (α-agents). However, since the computation required for implementation of virtual agents has to be carried out by embedded computers of a physical agent, for a more realistic picture of the computational virtual leader-follower architecture, see Figure 4-4b.*

Chapter 5

Flocking Algorithm for Non-Holonomic Networked Euler-Lagrange Systems

This chapter discusses a distributed connectivity-preserving artificial potential function (APF)-based flocking algorithm for formation control of a networked non-holonomic Euler-Lagrange (EL) mobile robot system. The control algorithm is initially analyzed through a theoretical framework by exploiting nonlinear control theory concepts for time-varying (non-autonomous) systems. Then, a numerical simulation study is performed that is guided by theoretical findings to verify the obtained results from the theoretical analysis of the control system.

5-1 State-Space Model of The Closed-Loop System

It is possible to perform a partial linearization of the dynamic reduced state-space model defined below

$$\begin{cases} \dot{q}_i = \mathcal{S}_i(q_i)v_{p,i} \\ \widetilde{\mathcal{M}}_i(q_i)\dot{v}_{p,i} + \widetilde{\mathcal{V}}_i(q_i, \dot{q}_i)v_{p,i} = \widetilde{\mathcal{B}}_i(q_i)\tau_i \end{cases} \tag{5-1}$$

via a computed torque nonlinear feedback to obtain a more convenient system for theoretical analysis and control design[34]. Consider the following dynamic computed torque nonlinear feedback control algorithm

$$\tau_i = [\widetilde{\mathcal{B}}_i]^{-1}(\widetilde{M}_i a_i + \widetilde{V}_i v_{p,i}) \tag{5-2}$$

Now by defining the state vector $x_i = [q_i, v_{p,i}]^T \in \mathbb{R}^{n+m}$, the control input vector $a_i = u_i \in \mathbb{R}^m$ and substituting (5-2) into (5-1), the state-space model of the closed-loop system can compactly be written as

$$\dot{x}_i = f_i(x_i) + g_i(x_i)u_i = \begin{bmatrix} \mathcal{S}_i(q_i)v_{p,i} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I_m \end{bmatrix} u_i \tag{5-3}$$

Note that the system results in a nonlinear control system with drift where $u_i \in \mathbb{R}^m$ denotes the flocking protocol. The system in (5-3) is also known as the second-order kinematic model of the constrained mechanical system[34]. In summary, nonlinear feedback control allows dynamic non-holonomic systems the possibility to cancel dynamic parameters by assuming[34]:

(i) the dynamical model is exactly known, and

(ii) the complete system state is measurable or computable from measurements

By assuming the above, the control problem can be directly solved at the velocity level, in a way where $v_{p,i}$ is synthesized such that the resulting system in (5-3) is regulated as desired. The computed torque nonlinear feedback requires the pseudo-velocities vector $v_{p,i}$ to be available to obtain the control input τ_i as defined in (5-2). The remarks below are of fundamental importance for the design and motivation of the flocking control algorithm.

Remark 10 ([34]). *The computed torque nonlinear feedback requires the measure of the pseudo-velocities vector $v_{p,i}$. When this is not directly accessible, it can be computed by pseudo-inverting the kinematic model $v_{p,i} = (\mathcal{S}_i^T \mathcal{S}_i)^{-1} \mathcal{S}_i^T \dot{q}_i$, provided that $q_i, \dot{q}_i$ are measurable.*

Remark 11 ([34]). *When $v_{p,i}$ is available, the dynamic computed torque nonlinear feedback equation can be used to obtain the control input τ_i at the generalized force level, provided that $v_{p,i}$ is sufficiently smooth (at least differentiable, since an acceleration term a_i appears in the torque nonlinear feedback). This motivates the design of smooth stabilizing flocking control algorithms.*

Remark 12 ([53]). *Note that, a conceptually simple nonlinear controller, commonly called computed torque controller in the robotics literature, can fully compensate the nonlinear forces in the robot motion and lead to high accuracy control for a very large range of robot speeds and a large workspace.*

Remark 13 ([53]). *In nonlinear control, the concept of feedback plays a fundamental role in controller design, as it does in linear control. However, the importance of feedforward is much more conspicuous than in linear control. Feedforward is used to cancel the effects of known disturbances and provide anticipative actions in tracking tasks. Very often it is impossible to control a nonlinear system stably without incorporating feedforward action in the control law. Note that a model of the plant is always required for feedforward compensation (although the model need not to be very accurate).*

Remark 14 ([53]). *Although the computed torque nonlinear feedback algorithm used for flocking control has a lot of beneficial contributions, it is also important to carefully highlight its limitations:*

- *There is no robustness guaranteed in the presence of parameter uncertainty or unmodelled dynamics (or unstructured uncertainties) due to the fact that the exact model of the nonlinear system is not available in performing feedback linearization. The sensitivity to modeling errors may be particularly severe when the linearizing transformation is poorly conditioned. This might be solved by exploiting an adaptive or robust control scheme to provide feedback linearizable systems with robustness to parametric uncertainties.*

- *The full state measurements are required in the feedback linearization algorithm. This limitation is due to the difficulty in finding convergent observers for nonlinear systems, and when they might be found, it is the lack of a general separation principle (analogous to that in linear systems), that guarantees the stability of the closed-loop system in case of the straightforward combination of both a stable feedback controller and stable observer.*

Lemma 9. *The partial linearized dynamic model of the networked differential-drive wheeled mobile robot (WMR) system that is obtained by simply plugging in the vector fields of the WMRs in (5-3) satisfies both the accessibility and the small-time local controllability conditions.*

Proof. Note that, by letting the computed torque nonlinear feedback control algorithm be

$$\tau_i = [\widetilde{\mathcal{B}}_i]^{-1}(\widetilde{M}_i u_i + \widetilde{V}_i v_{p,i})$$

the general partially linearized dynamical model in (5-3) is obtained. By plugging in the dynamics of the networked differential-drive WMR system, this can equivalently be written as

$$\dot{x}_i = \underbrace{\begin{bmatrix} v_i \cos\theta_i \\ v_i \sin\theta_i \\ \omega_i \\ 0 \\ 0 \end{bmatrix}}_{f_i(x_i)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}}_{g_1} u_{1,i} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{g_2} u_{2,i} = f_i(x_i) + g_1 u_{1,i} + g_2 u_{2,i}$$

with state vector $x_i = [x_i, y_i, \theta_i, v_i, \omega_i]^T \in \mathbb{R}^5$, which refers to the generalized coordinates of robot actuation point A. By computing the Lie brackets as in[34]

$$[g_1, f_i] = \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad [g_2, f_i] = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad [g_2, [f_i, [g_1, f_i]]] = \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

it is evident that the partial linearized dynamical model of the networked differential-drive WMR in (5-3) satisfies both the accessibility and the small-time local controllability conditions in Theorem 6 and Theorem 7, respectively. More formally, in fact, the vector fields

$$\Big(g_1, g_2, [g_1, f_i], [g_2, f_i], [g_2, [f_i, [g_1, f_i]]]\Big)$$

span a dimension of five at each state vector x_i and satisfy the conditions in Theorem 7. □

5-2 Flocking Algorithm for Non-Holonomic Networked Euler-Lagrange Systems

In this section, we study the distributed flocking problem based on proximity graphs for networked non-holonomic EL mobile robot systems.

Problem 1 (Flocking based on proximity graphs)**.** *Given a group of networked non-holonomic Euler-Lagrange mobile robots governed by (5-1), and consider the sensing radius $R_s > 0$ with $r_t \in (0, R_s)$. Assume that for any initial conditions $[q_i(0), v_{p,i}(0), \dot{v}_{p,i}(0)]^T$, $i = 1, \dots, N$, that makes $\mathcal{G}(0)$ an undirected connected graph, and in addition, suppose that at the initial time $t = 0$, no collisions occur among agents. Then, find a fully distributed APF-based connectivity-preserving flocking control algorithm τ_i of the form*

$$\tau_i = h_i(u_i, q_i, \dot{q}_i), \;\; j \in \mathcal{N}_i(t) \qquad i = 1, \dots, N \tag{5-4}$$

where h_i is a sufficiently smooth function to be designed as control algorithm with u_i as the flocking protocol, respectively, such that the closed-loop system has the following properties:

(i) connectivity preservation of the proximity interaction topology: $\mathcal{G}(t)$ is connected for all $t \geq 0$;

(ii) collision avoidance between agents;

(iii) velocity vector and heading angle alignment, i.e. the robots in the flock asymptotically move with the same velocity vector and heading angle;

- $\lim_{t\to\infty}(v_i - v_j) = 0,\ i = 1, \dots, N,\ \ j = 1, \dots, N.$
- $\lim_{x\to\infty}(\omega_i - \omega_j) = 0,\ i = 1, \dots, N,\ \ j = 1, \dots, N.$
- $\lim_{x\to\infty}(\theta_i - \theta_j) = 0,\ i = 1, \dots, N,\ \ j = 1, \dots, N.$

The objective is to design τ_i and u_i for each mobile agent to achieve flocking under a proximity network. This includes connectivity preservation, collision avoidance for all time, (asymptotically) velocity vector and orientation matching of the agents in the flock, respectively. The following distributed control algorithm is proposed

$$\begin{cases} u_{1i} = \underbrace{-\gamma_\phi \sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \vec{t}_i}_{\text{APF gradient term}} \underbrace{- \gamma_v \sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(v_i - v_j)}_{\text{Consensus term}} \\ u_{2i} = \underbrace{-\gamma_\phi \sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \vec{n}_i}_{\text{APF gradient term}} \underbrace{- \gamma_\omega \sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(\omega_i - \omega_j) - \gamma_\theta \sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(\theta_i - \theta_j)}_{\text{Consensus term}} \end{cases} \tag{5-5}$$

This can equivalently be rewritten into

$$u_i = \underbrace{-\gamma_\phi H_i \sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}}_{\text{APF gradient term}} \underbrace{- \gamma_{v_p} \sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(v_{p,i} - v_{p,j}) - \gamma_\theta \sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(\theta_{p,i} - \theta_{p,j})}_{\text{Consensus term}} \tag{5-6}$$

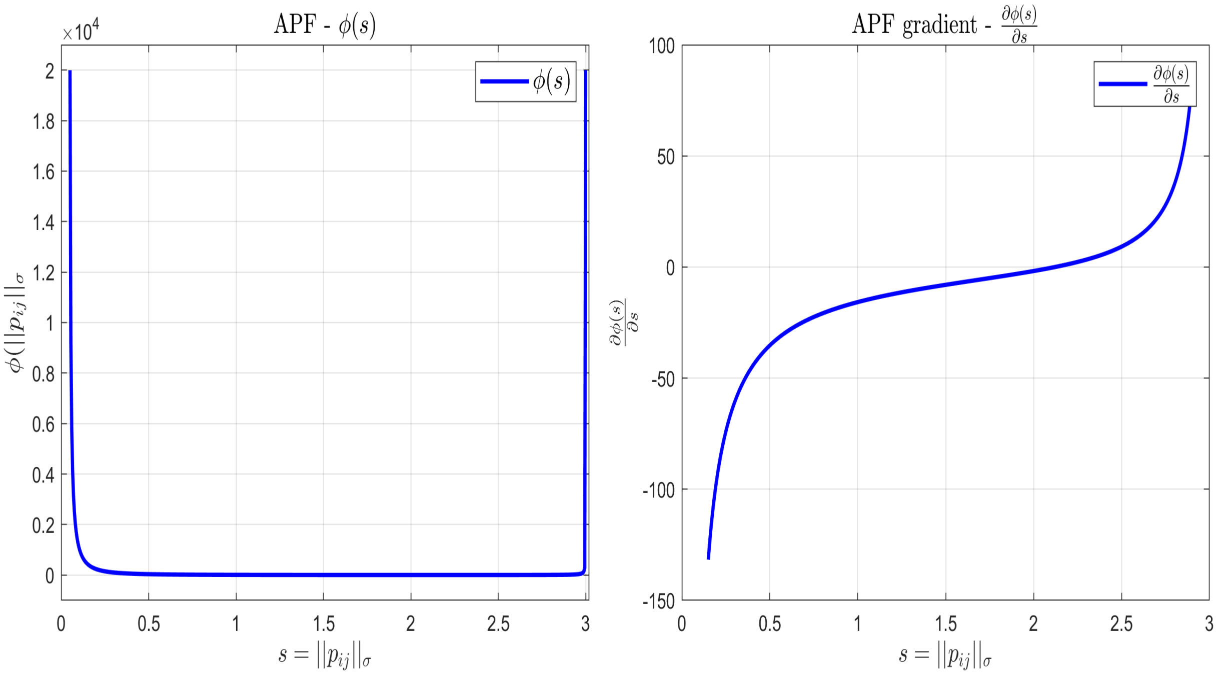
with

$$H_i = \begin{bmatrix} \vec{t}_i & \vec{n}_i \end{bmatrix}^T = \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix}$$

where $\theta_{p,i} = [0, \theta_i]^T$, $\gamma_\alpha > 0$ are control tuning parameters for $\alpha = \phi, v, \omega$ and θ, $p_{ij} = p_i - p_j$, with $p_i = [x_i, y_i]^T$ denoting the position of agent i. Moreover, $\vec{t}_i = [\cos\theta_i, \sin\theta_i]^T$ and $\vec{n}_i = [-\sin\theta_i, \cos\theta_i]^T$ are unit vectors orthogonal to each other in the tangential and normal direction, respectively. $\phi(||p_{ij}||_\sigma)$ denotes the APF between agents i and j, that is defined in Definition 5 and Lemma 10 with its derivative $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}$ in Lemma 11. Furthermore, $\mathcal{N}_i(t)$ denotes the neighboring set of the agents in proximity and $a_{ij}(t)$ denote the adjacency elements defining the edge weights associated with the proximity graph $\mathcal{G}(t)$, both defined in Subsection 4-2.

Remark 15. *In fact, note that the matrix H_i is just the inverse of the orthogonal rotation matrix (i.e. its transpose due to properties of orthogonal matrices) that maps the global coordinate frame (O, x_I, y_I) to the local coordinate frame (A, x_r, y_r), respectively.*

Definition 4 (σ-norm). *[14;41]* $||\mathcal{Z}||_\sigma = \frac{1}{\epsilon}(\sqrt{1+\epsilon||\mathcal{Z}||^2} - 1)$
The σ-norm of a vector is a map $\mathbb{R}^m \to \mathbb{R}_{\geq 0}$ with parameter $\epsilon > 0$ that remains fixed. Throughout this study, $\epsilon = 0.2$ will be used. The gradient $\sigma_\epsilon(\mathcal{Z}) = \Delta||\mathcal{Z}||_\sigma$ of the σ-norm is given by $\sigma_\epsilon(\mathcal{Z}) = \frac{\mathcal{Z}}{\sqrt{1+\epsilon||\mathcal{Z}||^2}} = \frac{\mathcal{Z}}{1+\epsilon||\mathcal{Z}||_\sigma}$. Note that the mapping $||\mathcal{Z}||_\sigma$ is differentiable everywhere, where $||\mathcal{Z}||$ is not differentiable at $\mathcal{Z} = 0$. Later on, this property of the σ-norm is exploited to construct smooth artificial potential functions and spatial adjacency matrices of a proximity graph, for a vector $\mathcal{Z}$ and a scalar $\epsilon > 0$. The relevance of a smooth stabilizing flocking control algorithm is already mentioned in Remark 11.



(a) The artificial potential function $\phi(s)$ with $s = ||p_{ij}||_\sigma$, for $r_t = 0.05$, $R_s = 3$ and $Q = 20000$.

(b) The gradient of the artificial potential function $\phi(s)$ with $s = ||p_{ij}||_\sigma$, for $r_t = 0.05$, $R_s = 3$ and $Q = 20000$.

Figure 5-1: Visualization of the artificial potential function $\phi(s)$ (a) with its derivative $\frac{\partial \phi(s)}{\partial s}$ (b) for intuitive and explanatory purposes.

Definition 5 ([14]). *[Artificial Potential Function (APF)] Motivated by the importance of connectivity preservation of the proximity interaction network and collision avoidance features to the flocking control algorithm, the following bounded APF is defined*

$$\phi(s) = \frac{s^2 - r_t^2}{R_s^2 - s^2 + \frac{R_s^2}{Q}} + \frac{R_s^2 - s^2}{s^2 - r_t^2 + \frac{R_s^2}{Q}}, \qquad r_t \leq s \leq R_s$$

where $Q > 0$, $s = ||p_{ij}||_\sigma$, $p_{ij} = p_i - p_j$ and the σ-norm is exploited to have a smooth potential function for its implementation in the flocking control algorithm. This APF $\phi(s)$ illustrated in Figure 5-1a.

Lemma 10 ([14]). *The APF $\phi(s)$ is a nonnegative function of s=$||p_{ij}||_\sigma$, and it is differentiable with respect to $||p_{ij}||_\sigma \in (r_t, R_s)$ so that*

(i) $\frac{d\phi(s)}{ds}$ *has a unique zero at* $r_0 = \sqrt{(R_s^2 + r_t^2)/2}$;

(ii) $\phi(s) \to (1 - r_t^2/R_s^2)Q = \overline{Q} \in [Q_{max}, +\infty)$ *as* $s \to R_s$ *or* $s \to r_t$.

Proof. The proof can be done by simple calculations and is thus omitted. □

Lemma 11 (Derivative of APF). *The derivative of the APF, denoted as $\frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i}$, is used in the flocking control algorithm for connectivity-preserving and collision avoidance purposes by introducing (virtual) attractive and repulsive forces, respectively. The derivative of the APF is defined as*

$$\Delta_{p_{ij}}\phi(||p_{ij}||_\sigma) = \frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i} = \Big(\frac{2||p_{ij}||_\sigma(R_s^2 - ||p_{ij}||_\sigma + \frac{R_s^2}{Q}) + 2||p_{ij}||_\sigma(||p_{ij}||_\sigma^2 - r_t^2)}{(R_s^2 - ||p_{ij}||_\sigma^2 + \frac{R_s^2}{Q})^2} - \frac{2||p_{ij}||_\sigma(||p_{ij}||_\sigma^2 - r_t^2 + \frac{R_s^2}{Q}) + 2||p_{ij}||_\sigma(R_s^2 - ||p_{ij}||_\sigma^2)}{(||p_{ij}||_\sigma^2 - r_t^2 + \frac{R_s^2}{Q})^2}\Big)\Big[\Delta_{p_{ij}\{1\}}||p_{ij}||_\sigma \;\; \Delta_{p_{ij}\{2\}}||p_{ij}||_\sigma\Big]$$

where

$$\Delta_{p_{ij}\{1\}}||p_{ij}||_\sigma = \frac{p_{ij}\{1\}}{1+\epsilon||p_{ij}||_\sigma}$$
$$\Delta_{p_{ij}\{2\}}||p_{ij}||_\sigma = \frac{p_{ij}\{2\}}{1+\epsilon||p_{ij}||_\sigma}$$

and the relative position vector p_{ij} *is defined as* $p_{ij} = [x_i - x_j, y_i - y_j]^T$*. In addition, the curly brackets* $\{.\}$ *indicate the selection of the corresponding element in that vector. The gradient-based APF* $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}$ *is illustrated in Figure 5-1b.*

Proof. The proof of this Lemma can easily be proven by using mathematical insights into the derivative of the Euclidian norm $||.||$ discussed in Section 3-5-1 and properties of the σ-norm discussed in Definition 4. □

Theorem 2. *[Flocking of non-holonomic networked Euler-Lagrange mobile agents] Consider a networked non-holonomic Euler-Lagrange dynamic mobile robot system (5-1) under the proximity interaction network* $\mathcal{G}(t)$ *and the assumption in Remark 16. Suppose that at the initial time* $t = 0$*, for any finite initial conditions* $[q_i(0), v_{p,i}(0), \dot{v}_{p,i}(0)]^T$*,* $i = 1, \ldots, N$*, such that* $V(.\ ,\ 0) \leq Q_{max} < \bar{Q}$*, the proximity graph* $\mathcal{G}(0)$ *is a connected graph and no collision occurs among the agents. Each agent is steered by the fully distributed computed torque nonlinear feedback control algorithm* τ_i *(5-2) and flocking control algorithm* u_i *(5-6), respectively. Consider the non-negative "energy-like" Lyapunov function candidate in (5-8). Then, for sufficiently large control parameters* γ_α *for* $\alpha = \phi, v_p, \theta$ *Problem 1 is solvable under the proposed controller, i.e.*

(i) the connectivity of the interaction network $\mathcal{G}(t)$ *is preserved for all* $t \geq 0$*;*

(ii) inter-agent collision avoidance is obtained;

(iii) the pseudo-velocities vector errors and orientation angle errors between the agents in the flock asymptotically converge to zero, i.e.

- $\lim_{t\to\infty}(v_{p,i} - v_{p,j}) = 0_m, \ \forall i \in \mathcal{V} \ \ \forall j \in \mathcal{N}_i(t)$
- $\lim_{t\to\infty}(\theta_i - \theta_j) = 0, \ \forall i \in \mathcal{V} \ \ \forall j \in \mathcal{N}_i(t)$

(iv) the system approaches a cohesive configuration and the agents attain a relative invariable distance by stabilizing the system and minimizing the global potential.

where for the non-negative Lyapunov function candidate in (5-8), it satisfies for all $t \in [0, t^*)$*,*

$$V(\tilde{p}, v_p, \theta_p, t) \leq V(\tilde{p}, v_p, \theta_p, 0) - \int_0^t \gamma_{vp}\tilde{x}_s^T(\tau)\tilde{\mathcal{L}}(t)\tilde{x}_s(\tau)d\tau \leq V(\tilde{p}, v_p, \theta_p, 0) - V_\infty \tag{5-7}$$

with $t^* = \infty$ *and* $\lim_{t\to\infty} V(\tilde{p}, v_p, \theta_p, t) = V_\infty < \infty$*.*

Proof. Using the computed torque nonlinear feedback control algorithm (5-2) into (5-1), a partially linearized state-space model of the closed-loop system is obtained and can compactly be written as in (5-3). Now, consider the following lower bounded non-negative "energy-like" function as a Lyapunov function candidate $V(t) = V_1(t) + V_2(t) + V_3(t)$, where

$$V(\tilde{p}, v_p, \theta_p, t) = \underbrace{\frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}\phi(||p_{ij}||_\sigma)}_{V_1(t)} + \underbrace{\frac{1}{2}\sum_{i=1}^{N} v_p^T{}_{,i}\, v_{p,i}}_{V_2(t)} + \underbrace{\frac{1}{2}\gamma_\theta\theta_p^T[\mathcal{L}(t)\otimes I_m]\theta_p}_{V_3(t)} \tag{5-8}$$

with $\widetilde{p} = p_{ij} = p_i - p_j$, θ_p is a column stack vector of $\theta_{p,i} = [0, \theta_i]^T$ for $i = 1, \ldots, N$ and $v_{p,i} = [v_i, d\omega_i]^T$, respectively. For the importance of maintaining the connectivity of the proximity graph $\mathcal{G}(t)$ and to avoid collision between neighboring agents, the non-negative potential function $V_1(t)$ is defined by the combination of the potential functions. The derivative of $V_1(t)$ is given by

$$\begin{aligned}\dot{V}_1 &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}\Big(\dot{p}_i^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \dot{p}_j^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}\Big)\\ &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}\Big(\dot{p}_i^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \dot{p}_j^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_j}\Big)\\ &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}\dot{p}_i^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \frac{1}{2}\sum_{j\in\mathcal{N}_i(t)}\sum_{i=1}^{N}\dot{p}_i^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}\\ &= \sum_{i=1}^{N}\dot{p}_i^T\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}\end{aligned} \tag{5-9}$$

by using Lemma 3.1 in[7] and the fact that $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = -\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_j}$. Then, the time derivative of the non-negative potential function $V_2(t)$ can be formulated as

$$\begin{aligned}\dot{V}_2 &= \sum_{i=1}^{N} v_{p},_{i}^{T}\,\dot{v}_{p,i}\\ &= \sum_{i=1}^{N} v_{p},_{i}^{T}\,u_i\end{aligned} \tag{5-10}$$

by using the fact that $\dot{v}_{p,i} = a_i = u_i \in \mathbb{R}^m$. Moreover, the derivative of $V_3(t)$ can be written by

$$\dot{V}_3 = \gamma_\theta\dot{\theta}_p^T[\mathcal{L}(t)\otimes I_m]\theta_p + \frac{1}{2}\gamma_\theta\theta_p^T[\dot{\mathcal{L}}(t)\otimes I_m]\theta_p \tag{5-11}$$

Now, the derivative of the Lyapunov function candidate $V(t)$ is given as

$$\begin{aligned}\dot{V} &= \underbrace{\sum_{i=1}^{N}\dot{p}_i^T\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}}_{\dot{V}_1} + \underbrace{\sum_{i=1}^{N} v_{p},_{i}^{T}\,u_i}_{\dot{V}_2} + \underbrace{\gamma_\theta\dot{\theta}_p^T[\mathcal{L}(t)\otimes I_m]\theta_p + \frac{1}{2}\gamma_\theta\theta_p^T[\dot{\mathcal{L}}(t)\otimes I_m]\theta_p}_{\dot{V}_3}\\ &= \sum_{i=1}^{N}\dot{p}_i^T\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \sum_{i=1}^{N} v_{p},_{i}^{T}\Big(-\gamma_\phi H_i\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \gamma_{v_p}\sum_{j\in\mathcal{N}_i(t)}a_{ij}(t)(v_{p},_{i} - v_{p},_{j}) -\\ &\quad \gamma_\theta\sum_{j\in\mathcal{N}_i(t)}a_{ij}(t)(\theta_{p},_{i} - \theta_{p},_{j})\Big) + \gamma_\theta\dot{\theta}_p^T[\mathcal{L}(t)\otimes I_m]\theta_p + \frac{1}{2}\gamma_\theta\theta_p^T[\dot{\mathcal{L}}(t)\otimes I_m]\theta_p\\ &= \sum_{i=1}^{N}\Big(\dot{p}_i^T - \gamma_\phi v_p^T{,}_i\,H_i\Big)\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp}\sum_{i=1}^{N} v_p^T{,}_i\sum_{j\in\mathcal{N}_i(t)}a_{ij}(t)(v_{p},_{i} - v_{p},_{j}) -\\ &\quad \gamma_\theta\sum_{i=1}^{N} v_p^T{,}_i\sum_{j\in\mathcal{N}_i(t)}a_{ij}(t)(\theta_{p},_{i} - \theta_{p},_{j}) + \gamma_\theta\dot{\theta}_p^T[\mathcal{L}(t)\otimes I_m]\theta_p + \frac{1}{2}\gamma_\theta\theta_p^T[\dot{\mathcal{L}}(t)\otimes I_m]\theta_p\\ &= \sum_{i=1}^{N}\Big(\dot{p}_i^T - \gamma_\phi v_p^T{,}_i\,H_i\Big)\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp}\sum_{i=1}^{N} v_p^T{,}_i\sum_{j\in\mathcal{N}_i(t)}a_{ij}(t)(v_{p},_{i} - v_{p},_{j}) -\\ &\quad \gamma_\theta\sum_{i=1}^{N}[v_i, \omega_i]\sum_{j\in\mathcal{N}_i(t)}a_{ij}(t)([0,\theta_i]^T - [0,\theta_j]^T) + \gamma_\theta\dot{\theta}_p^T[\mathcal{L}(t)\otimes I_m]\theta_p + \frac{1}{2}\gamma_\theta\theta_p^T[\dot{\mathcal{L}}(t)\otimes I_m]\theta_p\end{aligned} \tag{5-12}$$

$\dot{\mathcal{L}}(t)$ denotes the time derivative of $\mathcal{L}(t)$. Using the fact that $\dot{p}_i = H_i^T v_{p,i}$ by denoting $p_i = [x_i, y_i]^T$ as the position of the center of mass (COM), and by choosing $\gamma_\phi = 1$, the derivative of the Lyapunov function candidate is given as

$$\begin{aligned}\dot{V} &= -\gamma_{vp} \sum_{i=1}^{N} v_{p}^T{}_{,i} \sum_{j \in \mathcal{N}_i(t)} a_{ij}(t)(v_{p,i} - v_{p,j}) - \gamma_\theta \sum_{i=1}^{N} v_{p}^T{}_{,i} \sum_{j \in \mathcal{N}_i(t)} a_{ij}(t)(\theta_{p,i} - \theta_{p,j}) \\ &\quad + \gamma_\theta \dot{\theta}_p^T [\mathcal{L}(t) \otimes I_m]\theta_p + \frac{1}{2}\gamma_\theta \theta_p^T [\dot{\mathcal{L}}(t) \otimes I_m]\theta_p \\ &= -\gamma_{vp} v_p^T [\mathcal{L}(t) \otimes I_m] v_p - \gamma_\theta v_p^T [\mathcal{L}(t) \otimes I_m]\theta_p + \gamma_\theta \dot{\theta}_p^T [\mathcal{L}(t) \otimes I_m]\theta_p + \frac{1}{2}\gamma_\theta \theta_p^T [\dot{\mathcal{L}}(t) \otimes I_m]\theta_p \qquad (5\text{-}13)\end{aligned}$$

Now, define an auxiliary variable $\widetilde{x}_s = [v_p, \theta_p]^T$. Also, note that the second and third last terms of (5-13) are equivalent and cancel out. By exploiting the structure of the derivative of the Lyapunov function candidate, we can mathematically equivalent formulate it as

$$\begin{aligned}\dot{V} &= -\gamma_{vp} \underbrace{\begin{bmatrix} v_p & \theta_p \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} [\mathcal{L}(t) \otimes I_m] & 0 \\ 0 & 0 \end{bmatrix}}_{\widetilde{\mathcal{L}}(t)} \underbrace{\begin{bmatrix} v_p \\ \theta_p \end{bmatrix}}_{\widetilde{x}_s} - \gamma_\theta \underbrace{\begin{bmatrix} v_p & \theta_p \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & -\frac{1}{2}[\dot{\mathcal{L}}(t) \otimes I_m] \end{bmatrix}}_{\widehat{\mathcal{L}}(t)} \underbrace{\begin{bmatrix} v_p \\ \theta_p \end{bmatrix}}_{\widetilde{x}_s} \\ &= -\gamma_{vp} \widetilde{x}_s^T \widetilde{\mathcal{L}}(t) \widetilde{x}_s - \gamma_\theta \widetilde{x}_s^T \widehat{\mathcal{L}}(t) \widetilde{x}_s \qquad (5\text{-}14)\end{aligned}$$

By using properties for norms and matrices (see Subsection 3-5-2), we can write the expression as

$$\begin{aligned}\dot{V} &= -\gamma_{vp} \widetilde{x}_s^T \widetilde{\mathcal{L}}(t) \widetilde{x}_s \qquad (5\text{-}15) \\ &\leq -\gamma_{vp} \lambda_{min}[\widetilde{\mathcal{L}}(t)] \; ||\widetilde{x}_s||^2\end{aligned}$$

where we have used the fact that the topology matrix $\widetilde{\mathcal{L}}(t)$ is a symmetric and positive semi-definite matrix, the discussion in Remark 16 on $\widehat{\mathcal{L}}(t)$, Lemma 4-5, and v_p, θ_p are column stack vectors of $v_{p,i}, \theta_{p,i}$, $i = 1, \ldots, N$. Then, evidently it follows that $\dot{V}$ is negative semi-definite, i.e. $\dot{V} \leq 0$. By integrating the above arguments and from $V \geq 0$ and $\dot{V} \leq 0$, it can be concluded that V is a bounded function with (finite) limit, i.e. $\lim_{t \to \infty} V(\widetilde{p}, v_p, \theta_p, t) = V_\infty < \infty$ and $V(\widetilde{p}, v_p, \theta_p, t) \leq V(\widetilde{p}, v_p, \theta_p, 0)$. From this, we can subsequently conclude that $\phi(||p_{ij}||_\sigma), \theta_i, v_{p,i} \in \mathcal{L}_\infty$.

Remark 16. *Note that the Laplacian matrix, denoted by $\mathcal{L}(t)$, is a piecewise continuous function, that is continuous except for some discontinuities at a (finite) number of switching intervals. However, the time derivative of $V(.,t)$ exists everywhere since the Lyapunov function candidate in the proof of Theorem 2 is (a.e.) continuously differentiable. In addition, note that for $\widehat{\mathcal{L}}(t)$, when $\mathcal{L}(t)$ is a fixed interaction network in each nonempty, bounded and contiguous time-interval $[t_{i-1}, t_i)$, where $i = 1, 2, \ldots$ is the switching time and $\dot{\mathcal{L}}(t) = 0$ as a result, it will be equal to $\widetilde{x}_s^T \widehat{\mathcal{L}}(t) \widetilde{x}_s = 0$ since its associated quadratic form is zero.*
Now, assume there are $m_i \in \mathbb{N}$ new links being added to the proximity interaction network $\mathcal{G}(t)$ at the corresponding switching time t_i, $i = 1, 2, \ldots$.. For the reason that the initial interaction network $\mathcal{G}(0)$ is a connected proximity graph, the proposed connectivity-preserving control algorithm can guarantee that the sequence of switching topologies $\mathcal{G}(t_i)$ within $[t_i, t_{i+1})$ consists of connected graphs, i.e. $\mathcal{G}(t_i) \in \mathcal{G}_c$, with $\mathcal{G}_c$ the set of all connected proximity graphs on the vertices. Note that the number of vertices (representing the agents) is finite and therefore $\mathcal{G}_c$ is a finite set. Then, assume that we have $M \in \mathbb{N}$ new links that can be added to the initial interaction network $\mathcal{G}(0)$. Evidently, it holds that $0 < m_i \leq M$ and $i \leq M$. As a result, the number of switching times is finite, and the proximity interaction network $\mathcal{G}(t)$ of the system eventually becomes a fixed graph. Therefore, by collecting the arguments, it is a reasonable and realistic assumption to neglect $\dot{\mathcal{L}}(t)$ in the theoretical proof of Theorem 2. Therefore, the proof of Theorem 2 is still considered to be valid under this assumption.

Part 1: Proof Theorem 6 (i-ii) - (Connectivity preservation and collision avoidance)

Now, it will be shown that the connectivity of the proximity interaction network $\mathcal{G}(t)$ is preserved. It will be proven that agents that are close at $t=0$ will stay close to each other for all future time, i.e. $\mathcal{G}(0) \subseteq \mathcal{G}(t)$, and collisions among agents are avoided for all $t \geq 0$ by exploiting a contradiction argumentation scheme. Based on the continuity of the solution of the closed-loop non-holonomic dynamic Euler-Lagrange system (5-3), there exists $0 < t_1 \leq \infty$, such that for $t \in [0, t_1)$, $\mathcal{G}(t) = \mathcal{G}(0)$. Then, note that if $t_1 = \infty$, it implies that $\mathcal{G}(t)$ is connected for all $t \geq 0$. Otherwise, there must exist some $t_1 \geq 0$ such that $\mathcal{G}(t) = \mathcal{G}(0)$, $t \in [0, t_1)$ and $\mathcal{G}(t_1) \neq \mathcal{G}(0)$. Now, claim that $\mathcal{G}(0) \subseteq \mathcal{G}(t_1)$. Assume that $\mathcal{G}(t)$ switches at time t_i, $i = 1, 2, \ldots$ and is fixed on $[t_i, t_{i+1})$. Note that since the proximity graph $\mathcal{G}(t)$ is connected for $t \in [0, t_1)$, by Lemma 10, and by collecting the boundedness arguments before, the choice of gains guarantees $V(\widetilde{p}, v_p, t) \leq V(\widetilde{p}, v_p, 0) \leq Q_{max} < \overline{Q}$ for all $t \in [0, t_1)$.
Then, assume for some edge that $(i,j) \in \mathcal{E}(0)$, $(i,j) \notin \mathcal{E}(t_1)$. This indicates $\lim_{t \to t_1^-} ||p_{ij}||_\sigma = R_s$ and thus implies $\lim_{t \to t_1} V(\widetilde{p}, v_p, \theta_p, t) \geq \bar{Q}$, which contradicts $V(\widetilde{p}, v_p, \theta_p, t) \leq V(\widetilde{p}, v_p, \theta_p, 0) \leq Q_{max} < \bar{Q}$. Therefore, no existing edge will be lost at time t_1. As a result, $\mathcal{G}(0) \subseteq \mathcal{G}(t_1)$. By using similar argumentation, it can also be shown that $\lim_{t \to t_1^-} ||p_{ij}||_\sigma \not\to r_t$, otherwise we would get that $\lim_{t \to t_1^-} V(\widetilde{p}, v_p, \theta_p, t) \geq \bar{Q}$, which also is in contradiction with $V(\widetilde{p}, v_p, \theta_p, t) \leq V(\widetilde{p}, v_p, \theta_p, 0) \leq Q_{max} < \bar{Q}$. This implies that no collisions can occur between $t \in [0, t_1)$.
The Lyapunov function candidate $V(\widetilde{p}, v_p, \theta_p, t)$ is bounded for all $t \in [0, t^\star)$ as long as the proximity graph $\mathcal{G}(t)$ is connected in $t \in [0, t^\star)$. For the reason that $\mathcal{G}(t)$ can only have a finite amount of edges (bounded by $\frac{N(N-1)}{2}$ from above, which is the number of the links of a complete graph[40]), the above synthesis can be repeated, and we can conclude that $t^\star = +\infty$, and there exist a $k > 0$ such that $\mathcal{G}(t) = \mathcal{G}(0)$, $t \in [0, t_1)$, $\mathcal{G}(t) = \mathcal{G}(t_i) \supseteq \mathcal{G}(t_{i-1})$, $t \in [t_i, t_{i+1})$, $i = 1, \ldots, k-1$, $\mathcal{G}(t) = \mathcal{G}(t_k) \supseteq \mathcal{G}(t_{k-1})$, $t \in [t_k, \infty)$. Equivalently, $V(\widetilde{p}, v_p, \theta_p, t) \leq V(\widetilde{p}, v_p, \theta_p, 0) \leq Q_{max} < \bar{Q}$, for all $t \geq 0$ (i.e. no existing edges are lost).
Therefore, by collection of the arguments, we can conclude that the connectivity of the interaction network $\mathcal{G}(t)$ is preserved where no (existing) edges are lost. This shows that agents close to each other stay close to each other. Moreover, no inter-agent collisions will occur, for all $t \geq 0$ and $i, j \in \mathcal{V}$, respectively. This concludes (i-ii) from Theorem 2.

Part 2: Proof Theorem 6 (iii) - (Velocity vector and orientation matching)

Then, by integrating both sides of the expression in (5-15)

$$V(\widetilde{p}, v_p, \theta_p, t) \leq V(\widetilde{p}, v_p, \theta_p, 0) - \int_0^\infty \gamma_{vp} \widetilde{x}_s^T(\tau) \widetilde{\mathcal{L}}(t) \widetilde{x}_s(\tau) d\tau \leq V(\widetilde{p}, v_p, \theta_p, 0) - V_\infty \tag{5-16}$$

Since this integral is finite, we get that $\widetilde{x}_s \in \mathcal{L}_2$ which implies that $v_{p,i} \in \mathcal{L}_2$ and $\theta_i \in \mathcal{L}_2$, respectively. Then, from Definition 5 and Lemma 10-11, and the fact $\phi(||p_{ij}||_\sigma)$ is continuously differentiable with respect to $||p_{ij}||_\sigma \in (r_t, R_s)$, we get that $\frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i} \in \mathcal{L}_\infty$. From the fact that $v_{p,i} \in \mathcal{L}_\infty$ is bounded, it is evident that $\dot{p}_i \in \mathcal{L}_\infty$. Moreover, from Property (4-1.1-4-1.2), the closed-loop dynamics of the system (5-3), collecting the above boundedness results and the flocking algorithm (5-5), it can be concluded that $\dot{v}_{p,i} = u_i \in \mathcal{L}_\infty$ and $\tau_i \in \mathcal{L}_\infty$. Integrating the boundedness arguments, overall we have that $\widetilde{x}_s \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\dot{\widetilde{x}}_s \in \mathcal{L}_\infty$, from which we can conclude that $\dot{V}$ is uniformly continuous. Then, from Barbalat's Lemma, (Lemma 1-2), we can conclude that velocity vector matching and orientation angle consensus is asymptotically achieved, i.e. $\lim_{t \to \infty}(v_{p,i} - v_{p,j}) = 0_m$ and $\lim_{t \to \infty}(\theta_i - \theta_j) = 0$, respectively. This concludes (iii) of Theorem 2.

Part 3: Proof Theorem 6 (iv) - (Minimization of global potential)

Now, consider the agents having reached a flocking stability configuration by having proven (i-iii) of Theorem 2. This implies that $\theta_i(t) = \theta_j(t)$, $v_{p,i}(t) = v_{p,j}(t) = v_{p,s}$, and as a result $H_i(t) = H_j(t)$, in steady state. Furthermore, since the gradient of the APF $\frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i}$ is a symmetric pairwise potential function, it holds that $\sum_{i=1}^N \sum_{j \in \mathcal{N}_i(t)} \frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i} = 0$. Thus, in steady state, from the flocking control algorithm (5-6), we obtain

$$
\begin{aligned}
\dot{\bar{v}}_p &= \frac{1}{N}\sum_{i=1}^{N}\underbrace{\dot{v}_{p,i}}_{u_i} \\
&= \frac{1}{N}\Big(-\gamma_\phi \sum_{i=1}^{N} H_i \sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \gamma_{v_p}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(v_{p,i}-v_{p,j}) - \gamma_\theta \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(\theta_{p,i}-\theta_{p,j})\Big) \\
&= 0 \qquad (5\text{-}17)
\end{aligned}
$$

which implies that the velocity vector, $v_{p,i} = [v_i, \omega_i]^T$ of the center of mass of the non-holonomic agents are invariant, by noticing that the rotation matrix $H_i(t)$ will be the same for each robot at steady state for every future time instant.

In other words, at steady state, the terms for each robot are weighted by the same for each $H_i(t)$, which results in cancellation of the terms. From above, we can conclude that $\dot{v}_{p,i} = 0$, which implies that $v_i(t) = v_j(t) = v_s$ and $\omega_i(t) = \omega_j(t) = \omega_s$ will remain constant, and $\theta_i(t) = \theta_j(t)$. As a result, from algorithm (5-5), we obtain

$$
\begin{aligned}
-\sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \vec{t}_i &= 0 \\
-\sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \vec{n}_i &= 0 \qquad (5\text{-}18)
\end{aligned}
$$

Due to the fact that $\vec{t}_i$ and $\vec{n}_i$ are unit vectors that are orthogonal to each other, the above is equivalent to $\sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = 0$. From this, it can be said that the global potential is minimized due to

$$
\dot{V}_1 = \sum_{i=1}^{N} \dot{p}_i^T \sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = 0 \qquad (5\text{-}19)
$$

This proves conclusion (iv) from Theorem 2. □

Remark 17. *Clearly, as proven above, by using the computed torque nonlinear feedback control law* (5-2) *with u_i as the flocking control algorithm in* (5-6)*, the non-holonomic Euler-Lagrange mobile agents preserve the connectivity of the proximity interaction network $\mathcal{G}(t)$, avoid inter-agent collisions, and achieve velocity vector and orientation angle consensus for all $t \geq 0$, as long as the assumption in Theorem 2 holds for the initial time $t = 0$ and by the discussion Remark 16. The designed flocking control law is a continuous time-varying and smooth algorithm.*

5-3 Numerical Simulation Study

In this section, numerical simulation results are given to illustrate the effectiveness of the theoretical results obtained in Section 5-2. We consider the connectivity-preserving flocking of 10 homogeneous mobile robots under a proximity interaction network. The relative non-holonomic EL dynamics of the differential-drive wheeled mobile robots (WMRs) with its system matrices are given in (4-9-4-10) where the proximity architecture is described in Section 4-2.

In the simulation, we consider the same system parameters as in[39], i.e. $m = 6.64$ $[kg]$, $I = 0.06363$ $[kg.m^2]$, $I_w = 0.001$ $[kg.m^2]$, $L = 0.2$ $[m]$, $R = 0.08$ $[m]$ and $d = 0.2$ $[m]$. The initial positions of the 10 mobile robots are, respectively, $p_1(0) = [1, 0]^T$ $[m]$, $p_2(0) = [3, 0.5]^T$ $[m]$, $p_3(0) = [4, 1]^T$ $[m]$, $p_4(0) = [4.5, 1.5]^T$ $[m]$, $p_5(0) = [2, 2.5]^T$ $[m]$, $p_6(0) = [3.5, 3.5]^T$ $[m]$, $p_7(0) = [3.9, 3.9]^T$ $[m]$, $p_8(0) = [4.5, 2.2]^T$ $[m]$, $p_9(0) = [5.0, 2.5]^T$ $[m]$, and $p_{10}(0) = [2.5, 1]^T$ $[m]$. The orientation of each mobile robot is randomly generated while the initial velocities of the robots are assumed to be zero. The sensing radius of the proximity interaction network that represents the communication view of the mobile robots is given to be $R_s = 3$ $[m]$. Moreover, the other design parameters used are $r_t = 0.05$ $[m]$, $Q = 20000$ and $\epsilon = 0.2$.

The numerical integration scheme fourth order Runge-Kutta (RK4) is used in this simulation. The step-size used is $T_s = 0.01$ $[s]$. The RK4 is an effective and widely used method for solving initial-value problems of differential equations and can be used to construct high order accurate numerical methods[73]. The step-size is conveniently chosen such that the RK4 algorithm preserves the stability. This is, it preserves stability provided that the step-size does not become too large. Thus, in practice, the use of high order Runge-Kutta methods allows us to increase the step-size while still obtaining good accuracy but the stability of the algorithms establishes limits to the value of the step-size[73].

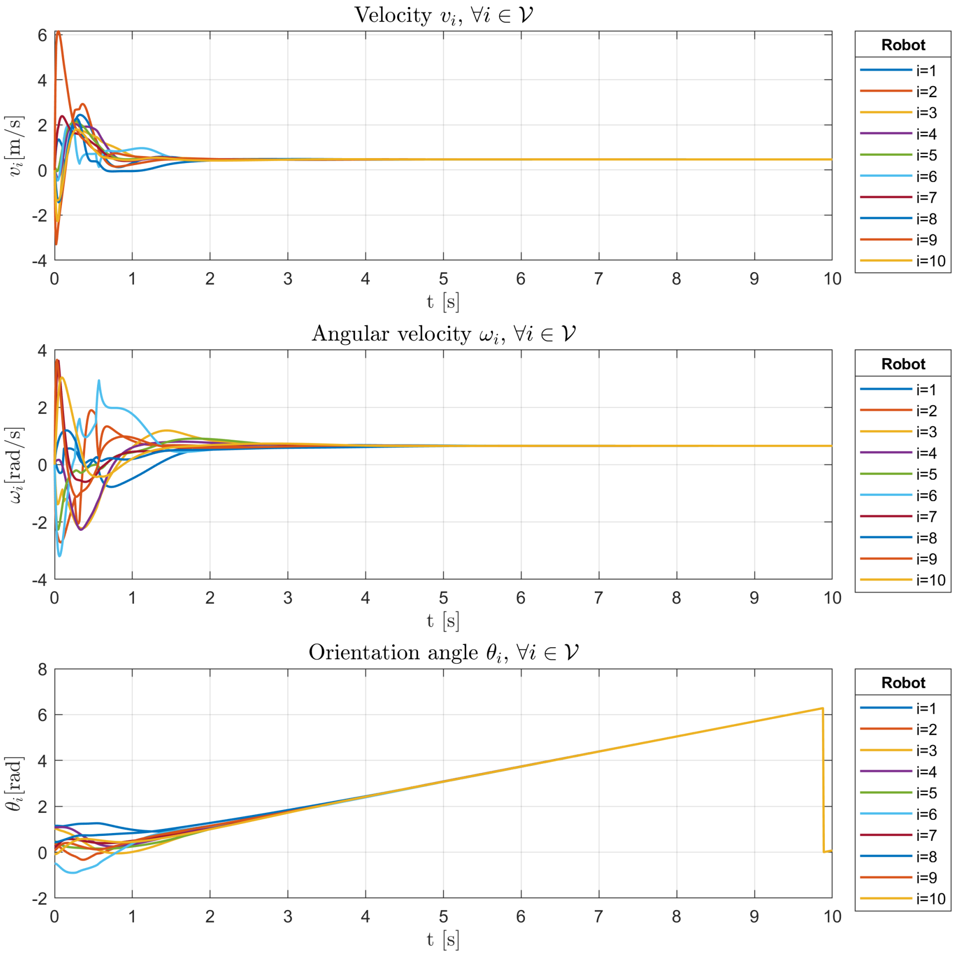
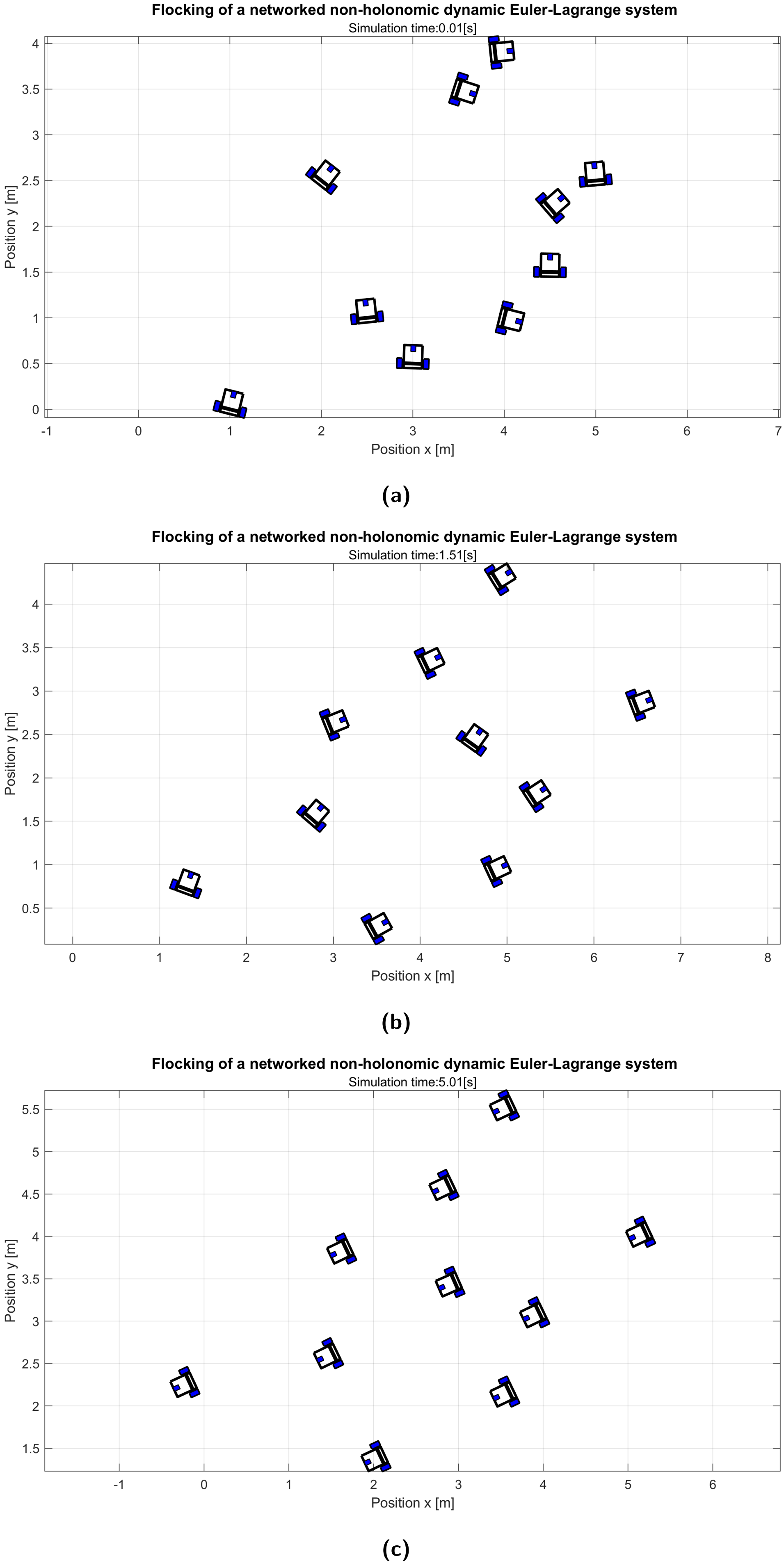


Figure 5-2: Simulation of the velocity vector $[v_i, \omega_i]^T$ and orientation angle θ_i consensus graphs for all the differential-drive WMRs in the networked system.

The control parameters are conveniently chosen as $\gamma_\phi = 1$, $\gamma_v = 2$, $\gamma_\omega = 2$ and $\gamma_\theta = 2$ guided by the theoretical analysis such that the best performance and a stable flocking system is obtained. An important remark on this concerning the stability is that the control parameters have to be carefully chosen such that the control bandwith of the system does not operate in the frequency region of the uncertainties and high frequency unmodelled dynamics which might lead to an unstable system.

From Figure 5-2, which shows the consensus graphs of the robots in the network, it is evident that the agents (asymptotically) move with the same velocity vector and orientation, respectively. From the simulation performed in Figure 5-3 and numerically, it is verified that no inter-agent collision occurs. Figure 5-4 shows the trajectories of the differential-drive WMRs. Evidently, from Figure 5-3 and Figure 5-4, it can be concluded that the agents move cohesively in a flock.



(a)

(b)

(c)

Figure 5-3: Simulation of the 10 differential-drive WMRs under the connectivity-preserving flocking algorithm (5-6) at (a) t = 0.01 [s], (b) t = 1.51 [s] and t = 5.01 [s].

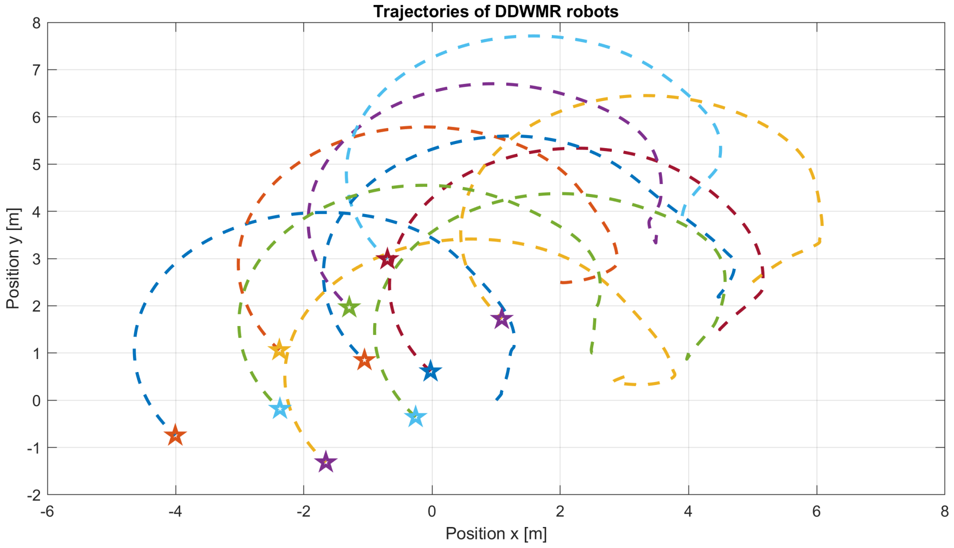
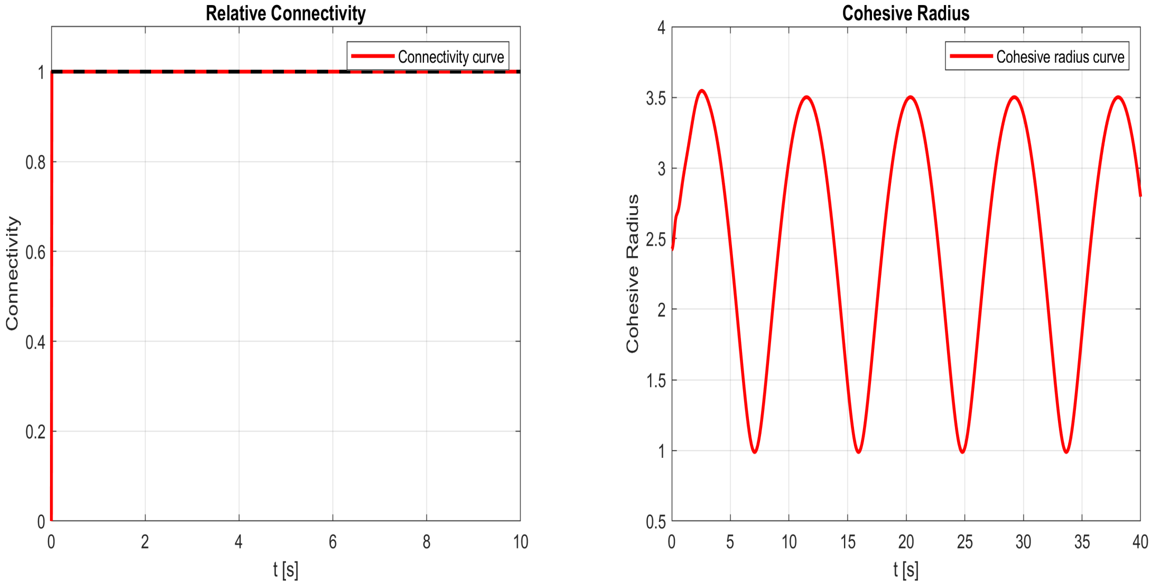


Figure 5-4: The trajectories of the mobile robots where the stars indicate the position of the robots at the end of the simulation.

Verification of Flocking To verify whether a group of particles performs flocking, we can calculate some performance quantities along the trajectory of the particles[41]. These quantities are defined in the following and can be used as performance metrics for flocking.

- *Relative Connectivity*: $C(t) = (1/N - 1)rank(A(q(t))) \in [0, 1]$
- *Cohesive Radius*: $R(t) = \max_{i \in \mathcal{V}} ||p_i(t) - p_c(t)||$, with $p_c(t) = \frac{\sum_{i=1}^{N} p_i}{N}$ denoting the geometrical center of mass of all agents in the flock.

In short, since the rank of a connected Laplacian and adjacency matrix of order N is at most $N - 1$, a relative connectivity $C(t)$ of 1 implies that the interaction graph $\mathcal{G}(t)$ is a connected graph. Moreover, a flock is cohesive when it has a finite cohesion radius. A more detailed discussion is given in[41].



(a) Relative connectivity **(b)** Cohesion radius

Figure 5-5: Visualization of relative connectivity and cohesive radius for numerical verification of flocking with connectivity-preserving and cohesive flock movement for intuitive and explanatory purposes.

The initial proximity interaction graph $\mathcal{G}(0)$ forms a connected graph. Figure 5-5a shows that the connectivity of the interaction network $\mathcal{G}(t)$ is preserved for all times $t \geq 0$. Furthermore, Figure 5-5b shows that the flock is a cohesive group by having a finite cohesive radius. Thus, by integrating all the arguments above, we can conclude that the flocking with connectivity-preservation of 10 non-holonomic EL networked mobile robots under a proximity graph is achieved.

Chapter 6

Virtual Leader-Following Flocking Algorithm for Non-Holonomic Networked Euler-Lagrange Systems

This chapter discusses a distributed (virtual) leader-following flocking algorithm for formation control of a networked non-holonomic Euler-Lagrange (EL) mobile robot system by including the virtual leader in the proximity network as a complement to the algorithm discussed in the Chapter 5. The control algorithm is initially analyzed through a theoretical framework by exploiting nonlinear control theory concepts for time-varying (non-autonomous) systems. Then, a numerical simulation study is performed that is guided by theoretical findings to support the obtained results from the theoretical analysis of the control system.

6-1 Virtual Leader-Following Flocking Algorithm for Non-Holonomic Networked Euler-Lagrange Systems

In this section, we study the distributed (virtual) leader-following flocking problem for networked non-holonomic EL mobile robot systems. In addition to the flocking algorithm discussed in Chapter 5, a virtual leader is included in the proximity interaction network to introduce the mission to the flock in practical situations.

Problem 2 (Flocking with a constant virtual leader based on proximity graphs)**.** *Given a group of networked non-holonomic Euler-Lagrange mobile robots governed by*

$$\begin{cases} \dot{q}_i = \mathcal{S}_i(q_i)v_{p,i} \\ \widetilde{\mathcal{M}}_i(q_i)\dot{v}_{p,i} + \widetilde{\mathcal{V}}_i(q_i,\dot{q}_i)v_{p,i} = \widetilde{\mathcal{B}}_i(q_i)\tau_i \end{cases} \tag{6-1}$$

and sensing radius $R_s > 0$ with $r_t \in (0, R_s)$. Assume that for any initial conditions $[q_i(0), v_{p,i}(0), \dot{v}_{p,i}(0)]^T$, $i = 0, \dots, N$, that makes $\widetilde{\mathcal{G}}(0)$ contain a spanning tree with node 0, i.e. the virtual leader, as the root. Assume that the virtual leader is a neighbor of at least one follower (i.e. node 0 $\in \widetilde{\mathcal{N}}_i(t)$ for at least one i) at each time instant t. In addition, suppose that at the initial time $t = 0$, no collisions occur among agents. Then, find a fully distributed artificial potential function (APF)-based connectivity-preserving flocking control algorithm τ_i of the form

$$\tau_i = h_i(u_i, q_i, \dot{q}_i), \;\; j \in \widetilde{\mathcal{N}}_i(t) \qquad i = 1, \dots, N \tag{6-2}$$

where h_i is a sufficiently smooth function to be designed as control algorithm with u_i as the flocking protocol, respectively, such that the closed-loop system is dissipative and has the following properties:

(i) *connectivity preservation of the interaction topology: $\widetilde{\mathcal{G}}(t)$ is a spanning tree for all $t \geq 0$;*

(ii) *collision avoidance and cohesive flock guidance: the networked multi-agent system moves cohesively with the virtual leader while collisions are avoided, i.e. formally written in mathematical terms as $||p_i - p_j|| > (r_t + 2L)$, $i = 1, \dots, N$, $j = 0, 1, \dots, N$;*

(iii) *velocity vector and heading angle alignment, i.e. the robots in the flock asymptotically move with the same velocity vector and heading angle as the virtual leader;*

- $\lim_{t\to\infty}(v_i - v_j) = 0$, $i = 1, \dots, N$ *and* $j = 0, \dots, N$.
- $\lim_{x\to\infty}(\omega_i - \omega_j) = 0$, $i = 1, \dots, N$ *and* $j = 0, \dots, N$.
- $\lim_{x\to\infty}(\theta_i - \theta_j) = 0$, $i = 1, \dots, N$ *and* $j = 0, \dots, N$.

A natural question is whether the networked control system obtained is dissipative. It might not be as predictable as the previous control system discussed in Chapter 5 in case of no virtual leader where only physical robots in proximity interact with each other. An intuitive reason for this is that in flocking without a virtual leader, each agent reciprocates the action of its neighboring agents under an undirected interaction network, whereas including a virtual leader in the proximity network induces a unidirectional graph (or digraph).

The (constant) virtual leader is a kinematic non-holonomic agent such that the generated reference trajectory is compatible with the permissible trajectories for the differential-drive wheeled mobile robots (WMRs) subject to non-holonomic constraints. A virtual leader has the role of a commander in charge of navigation and control of the behavior of the flock as a whole. The virtual leader is mathematically governed by

$$\dot{x}_0 = f_0(x_0) + g_0(x_0)u_0 = \begin{bmatrix} \mathcal{S}_0(q_0)v_{p,0} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I_m \end{bmatrix} u_0 \tag{6-3}$$

where $x_0 = [q_0, v_{p,0}]^T$, $\mathcal{S}_0(q_0)$ is the kinematics matrix of the virtual leader and $u_0 = \dot{v}_{p,0} = 0$, since we consider a constant virtual leader in this case. In fact, smoothness of the reference trajectory is a reasonable assumption in the case of mobile robots subject to non-holonomic constraints[37].

The objective is to design τ_i and u_i for each mobile agent to achieve leader-follower flocking under a proximity network. This includes cohesive flock guidance with connectivity preservation (i.e. followers move cohesively with the leader), collision avoidance for all time, (asymptotically) velocity vector and orientation matching of the agents in the flock, respectively. The following distributed control algorithm is proposed

$$\begin{cases} u_{1i} = \underbrace{-\gamma_\phi \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \dfrac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \vec{t}_i}_{\text{APF gradient term}} \underbrace{- \gamma_v \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(v_i - v_j)}_{\text{Consensus term}} \\ u_{2i} = \underbrace{-\gamma_\phi \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \dfrac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \vec{n}_i}_{\text{APF gradient term}} \underbrace{- \gamma_\omega \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\omega_i - \omega_j) - \gamma_\theta \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\theta_i - \theta_j)}_{\text{Consensus term}} \end{cases} \tag{6-4}$$

This can be rewritten into

$$u_i = \underbrace{-\gamma_\phi H_i \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}}_{\text{APF gradient term}} \underbrace{- \gamma_{v_p} \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(v_{p,i} - v_{p,j}) - \gamma_\theta \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\theta_{p,i} - \theta_{p,j})}_{\text{Consensus term}} \tag{6-5}$$

with

$$H_i = \left[\vec{t}_i \ \vec{n}_i\right]^T = \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix}$$

where $\theta_{p,i} = [0, \theta_i]$, $\gamma_\alpha > 0$ are control tuning parameters for $\alpha = \phi, v, \omega$ and θ , $p_{ij} = p_i - p_j$, with $p_i = [x_i, y_i]^T$ denoting the position of agent i. In addition, $\overrightarrow{t}_i = [\cos\theta_i, \sin\theta_i]^T$ and $\overrightarrow{n}_i = [-\sin\theta_i, \cos\theta_i]^T$ are unit vectors orthogonal to each other in the tangential and normal direction, respectively. $\phi(||p_{ij}||_\sigma)$ is the APF between agents i and j, that is defined in Definition 5 and Lemma 10 with its derivative $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}$ in Lemma 11. The neighboring set $\widetilde{\mathcal{N}}_i(t)$ includes the virtual leader in the proximity network. Furthermore, $p_0 = [x_0, y_0]^T$ and θ_0 denote the position and orientation angle of the virtual leader, respectively. The virtual leader is encapsulated as a reference trajectory formulated in the form of a consensus term in the control algorithm.

As discussed in Section 4-2, the virtual leader is only a neighbor of the following agents, i.e. there is a directed path from the virtual leader to the robots in proximity, and has itself no neighbors where its motion is not necessarily dependent on the follower robots in proximity. Moreover, $a_{ij}(t)$ denote the adjacency elements defining the edge weights associated with the proximity graph $\widetilde{\mathcal{G}}(t)$ defined in Section 4-2. It is important to note that the flocking control law is similar to that of the previous chapter, except for the proximity neighbor set, which now also includes the virtual leader as virtual agent.

Assumption 3. *The states of the virtual leader,* $q_0(t) = [x_0(t), y_0(t), \theta_0(t)]^T$ *and the velocity vector* $v_{p,0}\,(t) = [v_0(t), \omega_0(t)]^T$ *exist and are smooth, and are upper bounded by some unknown constants*[15;14].

Theorem 3 (Flocking of non-holonomic networked Euler-Lagrange mobile agents with a virtual leader)**.** *Consider a networked non-holonomic Euler-Lagrange dynamic system* (6-1) *and a constant virtual leader moving with equivalent non-holonomic kinematics (6-3) under the proximity graph* $\widetilde{\mathcal{G}}(t)$*, Assumption 3 and considering the assumption made in Remark 16. The virtual leader is a neighbor of at least one follower (i.e. node 0* $\in \widetilde{\mathcal{N}}_i(t)$ *for at least one i) at each time instant t and no collision occurs among the agents at the initial time.*

Consider the non-negative "energy-like" Lyapunov function candidate $V(.\ ,\ t)$ *given by* (6-9)*. Each agent is steered by the fully distributed computed torque nonlinear feedback control algorithm in* (6-6) *and flocking control algorithm* (6-5)*, respectively.*

$$\tau_i = [\widetilde{\mathcal{B}}_i]^{-1}(\widetilde{M}_i \underbrace{a_i}_{=u_i} + \widetilde{V}_i v_{p,i}) \tag{6-6}$$

Then, for sufficiently large control parameters γ_α *for* $\alpha = \phi, v_p, \theta$, $\dot{V} \leq 0$, *if*

$$-\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}(t)]\ ||\widetilde{x}_s||^2 + l_e||\widetilde{x}_s|| \leq 0 \tag{6-7}$$

where

$$\widetilde{\mathcal{H}}(t) = \begin{bmatrix} [\mathcal{H}(t) \otimes I_m] & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

More formally, if (6-7) *holds, we have that the energy of the system is non-increasing along the trajectory of the collective dynamics of the networked mobile robot control system.*

Proof. Using the computed torque nonlinear feedback control algorithm (6-6) into (6-1), a state-space model of the closed-loop system can compactly be written in companion form as

$$\dot{x}_i = f_i(x_i) + g_i(x_i)u_i = \begin{bmatrix} \mathcal{S}_i(q_i)v_{p,i} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I_m \end{bmatrix} u_i \tag{6-8}$$

Now, consider the following lower bounded "energy-like" function as a Lyapunov function candidate $V(t) = V_1(t) + V_2(t) + V_3(t)$, where

$$V(\widetilde{p}, \widetilde{\theta}_p, \widetilde{v}_p, t) = \underbrace{\frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)} \phi(||p_{ij}||_\sigma) + \sum_{i=1}^{N}\phi(||p_{i0}||_\sigma)}_{V_1(t)} + \underbrace{\frac{1}{2}\sum_{i=1}^{N}\widetilde{v}_p^T{}_{,i}\,\widetilde{v}_{p,i}}_{V_2(t)} + \underbrace{\frac{1}{2}\gamma_\theta\widetilde{\theta}_p^T[\mathcal{H}(t)\otimes I_m]\widetilde{\theta}_p}_{V_3(t)} \tag{6-9}$$

with $p_{ij} = p_i - p_j$, $\widetilde{\theta}_p$ is a column stack vector of $\widetilde{\theta}_{p,i} = [0, \widetilde{\theta}_i]^T$ with $\widetilde{\theta}_i = \theta_i - \theta_0$ for $i = 1, \ldots, N$ and $\widetilde{v}_{p,i} = v_{p,i} - v_{p,0}$, where we have defined $\widetilde{p}_i = p_i - p_0$.

For the importance of maintaining the connectivity of the directed proximity graph $\widetilde{\mathcal{G}}(t)$ (i.e. $\widetilde{\mathcal{G}}(t)$ is a spanning tree for all $t \geq 0$) and to avoid collision between neighboring agents, the non-negative potential function $V_1(t)$ is defined. The derivative of $V_1(t)$ is given by

$$\begin{aligned}
\dot{V}_1 &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}\Big(\dot{p}_i^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \dot{p}_j^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}\Big) + \sum_{i=1}^{N}\Big(\dot{p}_i^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i} - \dot{p}_0^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i}\Big) \\
&= \sum_{i=1}^{N}\dot{p}_i^T\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \sum_{i=1}^{N}\Big(\dot{p}_i^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i} - \dot{p}_0^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i}\Big) \\
&= \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}(\dot{p}_i - \dot{p}_0)^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \sum_{i=1}^{N}\dot{\widetilde{p}}_i^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i} \\
&= \sum_{i=1}^{N}\dot{\widetilde{p}}_i^T\sum_{j\in\widetilde{\mathcal{N}}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}
\end{aligned} \tag{6-10}$$

by using Lemma 3.1 in[7] and the fact that $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = -\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_j}$ to obtain the second equality term, and using the fact that $\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}\dot{p}_0^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = \dot{p}_0^T\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = 0$ to obtain the third equality term. Then, the time derivative of $V_2(t)$ can be formulated as

$$\begin{aligned}
\dot{V}_2 &= \sum_{i=1}^{N}\widetilde{v}_{p,i}^T\,\dot{\widetilde{v}}_{p,i} \\
&= \sum_{i=1}^{N}\widetilde{v}_{p,i}^T\,u_i
\end{aligned} \tag{6-11}$$

by using the fact that $\dot{\widetilde{v}}_{p,i} = \dot{v}_{p,i} - \underbrace{\dot{v}_{p,0}}_{=0} = u_i \in \mathbb{R}^m$, since the virtual leader has a constant velocity.

Note that the flocking protocol in (6-5) can mathematically equivalent be written as

$$u_i = \underbrace{-\gamma_\phi H_i\sum_{j\in\widetilde{\mathcal{N}}_i(t)}\frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i}}_{\text{APF gradient term}}\underbrace{-\gamma_{v_p}\sum_{j\in\widetilde{\mathcal{N}}_i(t)}a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j}) - \gamma_\theta\sum_{j\in\widetilde{\mathcal{N}}_i(t)}a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j})}_{\text{Consensus term}} \tag{6-12}$$

where $\widetilde{p}_{ij} = \widetilde{p}_i - \widetilde{p}_j = p_i - p_j$ with $\widetilde{p}_i = p_i - p_0$, $\widetilde{v}_{p,i} = v_{p,i} - v_{p,0}$ and $\widetilde{\theta}_{p,i} = \theta_{p,i} - \theta_{p,0}$. The time derivative of $V_3(t)$ can be written as

$$\dot{V}_3 = \gamma_\theta\dot{\widetilde{\theta}}_p^T[\mathcal{H}(t)\otimes I_m]\widetilde{\theta}_p + \frac{1}{2}\gamma_\theta\widetilde{\theta}_p^T[\dot{\mathcal{H}}(t)\otimes I_m]\widetilde{\theta}_p \tag{6-13}$$

Now, the derivative of the Lyapunov function candidate $V(t)$ is given as

$$
\begin{aligned}
\dot{V} &= \underbrace{\sum_{i=1}^{N} \dot{\widetilde{p}}_i^T \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i}}_{\dot{V}_1} + \underbrace{\sum_{i=1}^{N} \widetilde{v}_{p,i}^{\ T}\, u_i}_{\dot{V}_2} + \underbrace{\gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{\theta}_p + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p}_{\dot{V}_3} \\
&= \sum_{i=1}^{N} \dot{\widetilde{p}}_i^T \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} + \sum_{i=1}^{N} \widetilde{v}_{p,i}^{\ T} \Big(- \gamma_\phi H_i \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} - \gamma_{v_p} \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j}\,) - \\
&\gamma_\theta \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j}\,)\Big) + \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{\theta}_p + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p \\
&= \sum_{i=1}^{N} \Big(\dot{\widetilde{p}}_i^T - \gamma_\phi \widetilde{v}_p^T{}_{,i}\, H_i\Big) \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp} \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j}\,) - \\
&\gamma_\theta \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j}\,) + \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{\theta}_p + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p \\
&= \sum_{i=1}^{N} \Big(\dot{\widetilde{p}}_i^T - \gamma_\phi \widetilde{v}_p^T{}_{,i}\, H_i\Big) \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp} \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j}\,) - \\
&\gamma_\theta \sum_{i=1}^{N} [\widetilde{v}_i, \widetilde{\omega}_i] \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)([0, \widetilde{\theta}_i]^T - [0, \widetilde{\theta}_j]^T) + \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{\theta}_p + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p
\end{aligned}
\tag{6-14}
$$

$\dot{\mathcal{H}}(t)$ denotes the time derivative of $\mathcal{H}(t)$. Using the fact that $\dot{\widetilde{p}}_i^T = \dot{p}_i^T - \dot{p}_0^T = v_p^T{}_{,i}\, H_i - v_p^T{}_{,0}\, H_0$ and $\widetilde{v}_p^T{}_{,i}\, H_i = v_p^T{}_{,i}\, H_i - v_p^T{}_{,0}\, H_i$, where $p_i = [x_i, y_i]^T$ denotes the position of the center of mass (COM), we can formulate the derivative of the Lyapunov function candidate as

$$
\begin{aligned}
\dot{V} &= \sum_{i=1}^{N} \Big(\dot{\widetilde{p}}_i^T - \gamma_\phi \widetilde{v}_p^T{}_{,i}\, H_i\Big) \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp} \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j}\,) \\
&- \gamma_\theta \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j\in\widetilde{\mathcal{N}}_i(t)} a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j}\,) + \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{\theta}_p + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p \\
&= \sum_{i=1}^{N} \Big(v_p^T{}_{,i}\, H_i - v_p^T{}_{,0}\, H_0 - \gamma_\phi (v_p^T{}_{,i}\, H_i - v_p^T{}_{,0}\, H_i)\Big) \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp} \widetilde{v}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{v}_p \\
&- \gamma_\theta \widetilde{v}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{\theta}_p + \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{\theta}_p + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p
\end{aligned}
\tag{6-15}
$$

where $\widetilde{v}_p$ is a column stack vector of $\widetilde{v}_{p,i}\,$, $i = 1, \ldots, N$, $\mathcal{H}(t)$ denotes the interaction network of the virtual leader-following architecture and I_m is the $m \times m$ identity matrix. The leader-follower interaction topology matrix is defined as $\mathcal{H} = \mathcal{L} + \Lambda$, where $\mathcal{L}$ is the Laplacian matrix and $\Lambda = diag\{a_{10}, \ldots, a_{N0}\}$. It is evident that $\mathcal{H}$ is symmetric positive definite. A more detailed discussion is given in Section 4-2. Also, notice that the second last two terms in (6-15) are equivalent to each other and cancel out as a result. Now, by choosing $\gamma_\phi = 1$, the derivative of the Lyapunov function candidate is given as

$$
\dot{V} = \underbrace{\sum_{i=1}^{N} \Big(v_p^T{}_{,0}\,(H_i - H_0)\Big) \sum_{j\in\widetilde{\mathcal{N}}_i(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i}}_{(1)} - \underbrace{\gamma_{vp} \widetilde{v}_p^T [\mathcal{H}(t) \otimes I_m]\widetilde{v}_p}_{(2)} + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p \tag{6-16}
$$

Note that (1) is upper bounded by Assumption 3, the definition of the APF in Definition 5, Lemma 10-11, and from the fact that the orthogonal rotation matrix $||H_i|| = 1$ is always bounded.

Moreover, by the positive (semi)-definiteness of the interaction graph of the virtual leader-follower network $\mathcal{H}(t)$, it can be concluded that (2) is already negative (semi)-definite, i.e. $-\gamma_{vp}\widetilde{v}_p^T[\mathcal{H}(t) \otimes I_m]\widetilde{v}_p \leq 0$. In fact, note that this interaction network is actually positive definite in our case since we assume that there is at least one following robot in the proximity of the virtual leader at each time instant. Now, by algebraic manipulation, we can write the expression in (6-16) equivalently as

$$\dot{V} = \underbrace{\begin{bmatrix} v_p^T,_0(H_1 - H_0) & v_p^T,_0(H_2 - H_0) & \dots & v_p^T,_0(H_N - H_0) \end{bmatrix}}_{\widetilde{v}_e^T} \underbrace{\begin{bmatrix} \sum_{j \in \widetilde{\mathcal{N}}_1(t)} \frac{\partial \phi(||\widetilde{p}_{1j}||_\sigma)}{\partial p_1} \\ \sum_{j \in \widetilde{\mathcal{N}}_2(t)} \frac{\partial \phi(||\widetilde{p}_{2j}||_\sigma)}{\partial p_2} \\ \vdots \\ \sum_{j \in \widetilde{\mathcal{N}}_N(t)} \frac{\partial \phi(||\widetilde{p}_{Nj}||_\sigma)}{\partial p_N} \end{bmatrix}}_{\Delta_{p_i}\phi_s} - \gamma_{vp}\widetilde{v}_p^T[\mathcal{H}(t) \otimes I_m]\widetilde{v}_p$$

$$+ \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T[\dot{\mathcal{H}}(t) \otimes I_m]\widetilde{\theta}_p \tag{6-17}$$

Then, by conveniently introducing a new variable $\widetilde{x}_s = [\widetilde{v}_p, \Delta_{p_i}\phi_s, \widetilde{\theta}_p]^T$, we can formulate the expression (6-17) as

$$\begin{aligned} \dot{V} &= -\gamma_{vp} \underbrace{\begin{bmatrix} \widetilde{v}_p & \Delta_{p_i}\phi_s & \widetilde{\theta}_p \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} [\mathcal{H}(t) \otimes I_m] & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\widetilde{\mathcal{H}}(t)} \underbrace{\begin{bmatrix} \widetilde{v}_p \\ \Delta_{p_i}\phi_s \\ \widetilde{\theta}_p \end{bmatrix}}_{\widetilde{x}_s} \\ &\quad - \gamma_\theta \underbrace{\begin{bmatrix} \widetilde{v}_p & \Delta_{p_i}\phi_s & \widetilde{\theta}_p \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{2}[\dot{\mathcal{H}}(t) \otimes I_m] \end{bmatrix}}_{\widehat{\mathcal{H}}(t)} \underbrace{\begin{bmatrix} \widetilde{v}_p \\ \Delta_{p_i}\phi_s \\ \widetilde{\theta}_p \end{bmatrix}}_{\widetilde{x}_s} + \underbrace{\begin{bmatrix} \widetilde{v}_p & \Delta_{p_i}\phi_s & \widetilde{\theta}_p \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} 0 \\ I \\ 0 \end{bmatrix}}_{E} \widetilde{v}_e \\ &= \underbrace{-\gamma_{vp}\widetilde{x}_s^T\widetilde{\mathcal{H}}(t)\widetilde{x}_s}_{\leq 0} - \gamma_\theta\widetilde{x}_s^T\widehat{\mathcal{H}}(t)\widetilde{x}_s + \widetilde{x}_s^T E\widetilde{v}_e \end{aligned} \tag{6-18}$$

Clearly, the first part of this expression is negative semi-definite, due to the quadratic expression and the positive semi-definiteness of $\widetilde{\mathcal{H}}$. In fact, $\widetilde{\mathcal{H}}$ is a symmetric and positive semi-definite matrix, therefore, by using some properties of norms for vectors and matrices (see Subsection 3-5-2), and considering a similar discussion on the assumption made in Remark 16 on $\widehat{\mathcal{H}}(t)$, we can write the expression (6-18) as

$$\begin{aligned} \dot{V} &= -\gamma_{vp}\widetilde{x}_s^T\widetilde{\mathcal{H}}(t)\widetilde{x}_s + \widetilde{x}_s^T E\widetilde{v}_e \\ &\leq -\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}(t)]\ ||\widetilde{x}_s||^2 + \underbrace{||E\widetilde{v}_e||}_{\leq ||E||\ ||\widetilde{v}_e||}\ ||\widetilde{x}_s|| \\ &\leq -\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}(t)]\ ||\widetilde{x}_s||^2 + l_e||\widetilde{x}_s|| \end{aligned} \tag{6-19}$$

where we have assumed that $||\widetilde{v}_e|| < l_e$ and the fact that $||E|| = 1$. Therefore, with the introduction of a new auxiliary variable $\widetilde{x}_s$, defined as $\widetilde{x}_s = [\widetilde{v}_p, \Delta_{p_i}\phi_s, \widetilde{\theta}_p]$, we have that $\dot{V} \leq 0$, if

$$-\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}(t)]\ ||\widetilde{x}_s||^2 + l_e||\widetilde{x}_s|| \leq 0 \tag{6-20}$$

More formally, when this holds, the energy of the system is non-increasing (i.e. $\dot{V}(.\ ,\ t) \leq 0$) along the trajectory of the collective dynamics of the networked mobile multi-robot system.

□

The positive linear-in-$||\widetilde{x}_s||$ term makes the flocking stability more questionable since in the (virtual) leader-following case it is more difficult for $\dot{V}(.\ ,\ t) \leq 0$ to hold. Therefore, convenient tuning of the control parameters $(\gamma_{vp}, \gamma_\phi, \gamma_\theta)$ is required such that the negative part of $\dot{V}$ dominates the positive term in the expression. However, for small values of $||\tilde{x}_s||$ that are close to zero, the result in (6-7) is hard to satisfy, as in this case the linear terms dominate over the quadratic terms. Concluding, we can not say anything on the desired properties of Problem 2 and a numerical simulation study is performed in the next section as a complement to the theoretical results for verification.

Remark 18. *To design a more practical flocking algorithm, the virtual leader has to be included in the proximity network of the multi-robot system, which is done in this chapter. The virtual leader is not affected by the followers, since the virtual leader is only a neighbor of the following agents, which results in a directed graph (digraph). To clarify and interpret the idea of putting the virtual leader in the proximity network, note that we have a kind of tracking problem of the leader-following architecture, therefore, the virtual leader has to stay cohesive to the flock which is done by including the virtual leader in the proximity interaction network.*

Remark 19. *Note that Barbalat's Lemma (see Lemma 1-2) is not applicable in this case, since it implies an asymptotic stability result for non-autonomous systems. It is not possible to guarantee the negative semi-definiteness of $\dot{V}$ for all time instants, i.e. $\dot{V} \leq 0$ $\forall t$ due to the positive linear-in-$||\tilde{x}_s||$ term l_e $||\tilde{x}_s||$.*

Remark 20. *Note that the positive linear-in-$||\tilde{x}_s||$ term makes it more difficult for $\dot{V} \leq 0$ to hold and makes the flocking stability questionable. Furthermore, note that, even if V converges, it does not necessarily imply that velocity matching and other desired characteristics are achieved. Thus, we can not conclude anything more and a numerical simulation study is performed as a complement to the theoretical analysis guided by the obtained theoretical results for verification on the desired properties of Problem 2; connectivity preservation, collision avoidance, velocity vector and orientation matching for all time, respectively.*

Remark 21. *In addition to Remark 20, an interesting observation is that by simply including a virtual leader in the proximity network of the interacting system further complicates the theoretical proof of Theorem 3, as compared to the theoretical proof of Theorem 2 in Chapter 5. An intuitive explanation for this is that in leaderless flocking, each agent recompenses the behavior of the neighboring robots in the proximity network under an (undirected) interaction network.*
However, by including a virtual leader to the proximity network, the interaction network results in a digraph with the virtual leader node as root. As a consequence, the behavior of the virtual leader can not be affected by the follower robots and moves fully independently. In addition, different to the existing literature, which consider point masses, the virtual leader is a more complex agent, modelled as a non-holonomic kinematic agent, such that the reference trajectory is compatible with the non-holonomic constraints of the robots. This interaction between virtual leader and following robots leads to a more complex relation and eventually results in a positive linear-in-$||\tilde{x}_s||$ term, which complicates the theoretical proof.

6-2 Numerical Simulation Study

In this section, numerical simulation results are given to illustrate the effectiveness of the theoretical results obtained in Section 6-1. We consider the distributed virtual leader-follower flocking of 10 homogeneous mobile robots and a constant virtual leader under a proximity interaction network. The relative non-holonomic EL dynamics of the differential-drive WMRs with its system matrices are given in (4-9-4-10) where the virtual-leader follower proximity architecture is defined in Section 4-2.
In the simulation, we consider the same system parameters as in[39], i.e. $m = 6.64$ $[kg]$, $I = 0.06363$ $[kg.m^2]$, $I_w = 0.001$ $[kg.m^2]$, $L = 0.2$ $[m]$, $R = 0.08$ $[m]$ and $d = 0.2$ $[m]$. The initial positions of the 10 mobile robots and virtual leader are, respectively, $p_0(0) = [3.5, 2.5]^T$ $[m]$, $p_1(0) = [1, 0]^T$ $[m]$, $p_2(0) = [3, 0.5]^T$ $[m]$, $p_3(0) = [4, 1]^T$ $[m]$, $p_4(0) = [4.5, 1.5]^T$ $[m]$, $p_5(0) = [2, 2.5]^T$ $[m]$, $p_6(0) = [3.5, 3.5]^T$ $[m]$, $p_7(0) = [3.9, 3.9]^T$ $[m]$, $p_8(0) = [4.5, 2.2]^T$ $[m]$, $p_9(0) = [5, 2.5]^T$ $[m]$, and $p_{10}(0) = [2.5, 1]^T$ $[m]$. The orientation of each mobile robot is randomly generated while the initial

velocities of the robots are assumed to be zero. The leader starts under an orientation of $\theta_0 = 40$ $[deg]$. The constant velocity vector of the leader is assumed to be $v_{p,0} = [3.0, -0.3]^T$. The sensing radius of the proximity interaction network that represents the communication view of the mobile robots is given to be $R_s = 3$ $[m]$. Moreover, the other design parameters used are $r_t = 0.05$ $[m]$, $Q = 20000$ and $\epsilon = 0.2$.

The numerical integration scheme fourth order Runge-Kutta (RK4) is used in this simulation. Moreover, the step-size used is $T_s = 0.01$ $[s]$. The RK4 is an effective and widely used method for solving initial-value problems of differential equations and can be used to construct high order accurate numerical methods[73]. The step-size is conveniently chosen such that the RK4 algorithm preserves the stability. This is, it preserves stability provided that the step-size does not become too large. Thus, in practice, the use of high order Runge-Kutta methods allows us to increase the step-size while still obtaining good accuracy but the stability of the algorithms establishes limits to the value of the step-size[73].

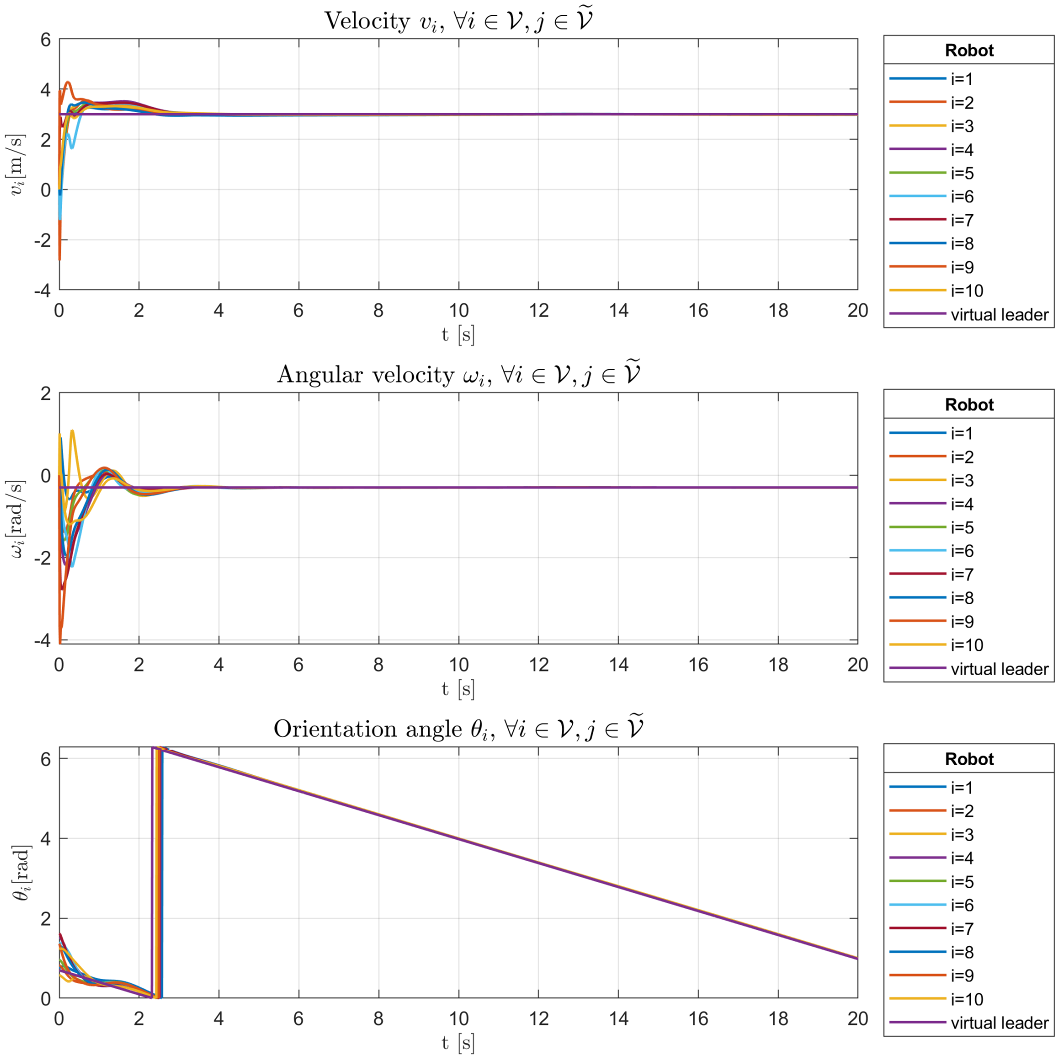
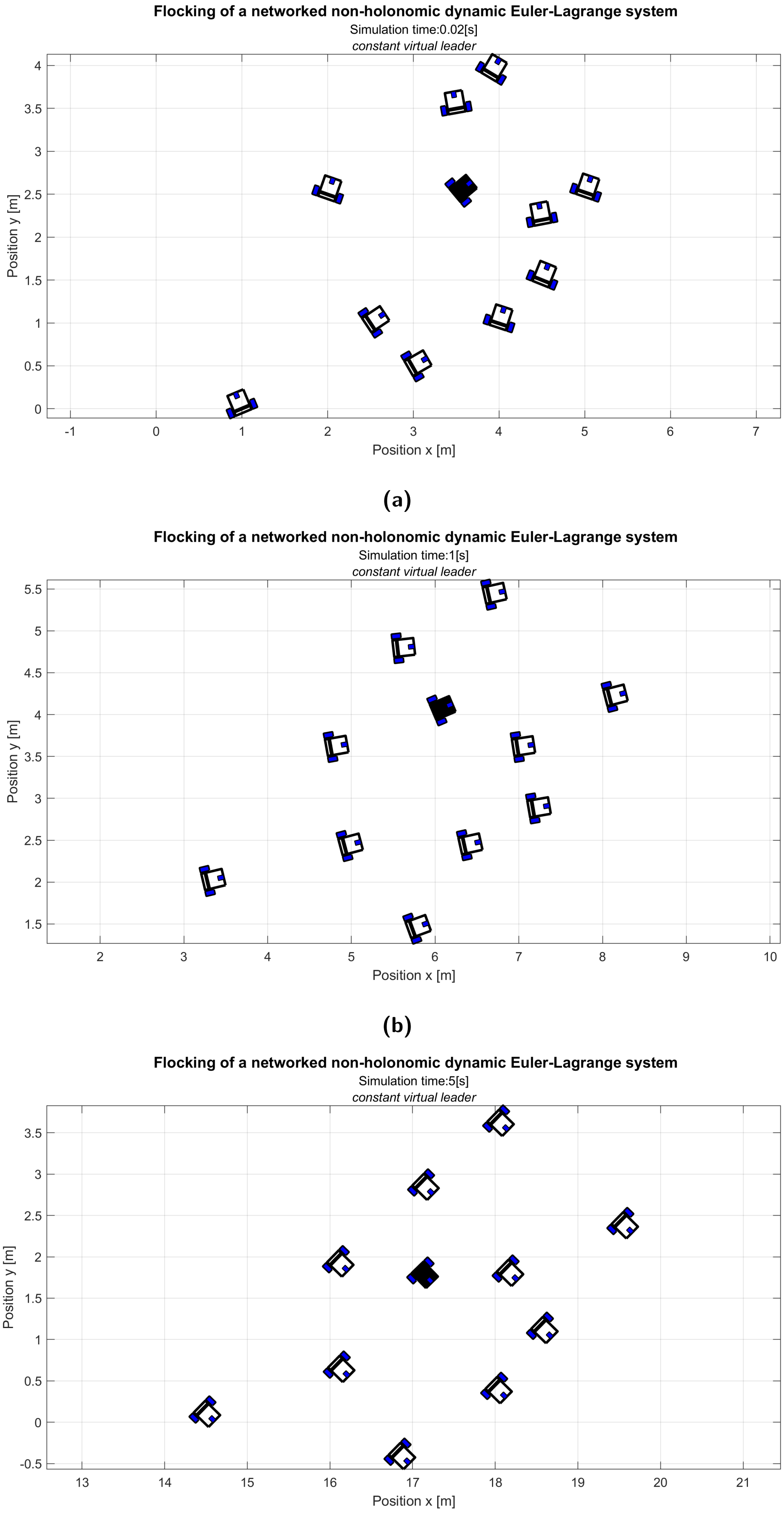


Figure 6-1: Simulation of the velocity vector $[v_i, \omega_i]^T$ and orientation angle θ_i consensus graphs for all the differential-drive WMRs and virtual leader in the networked system.



(a)

(b)

(c)

Figure 6-2: Simulation of the 10 differential-drive WMRs and virtual leader (black) under the distributed leader-following flocking algorithm (6-5) at (a) t = 0.02 [s], (b) t = 1.0 [s] and t = 5.0 [s].

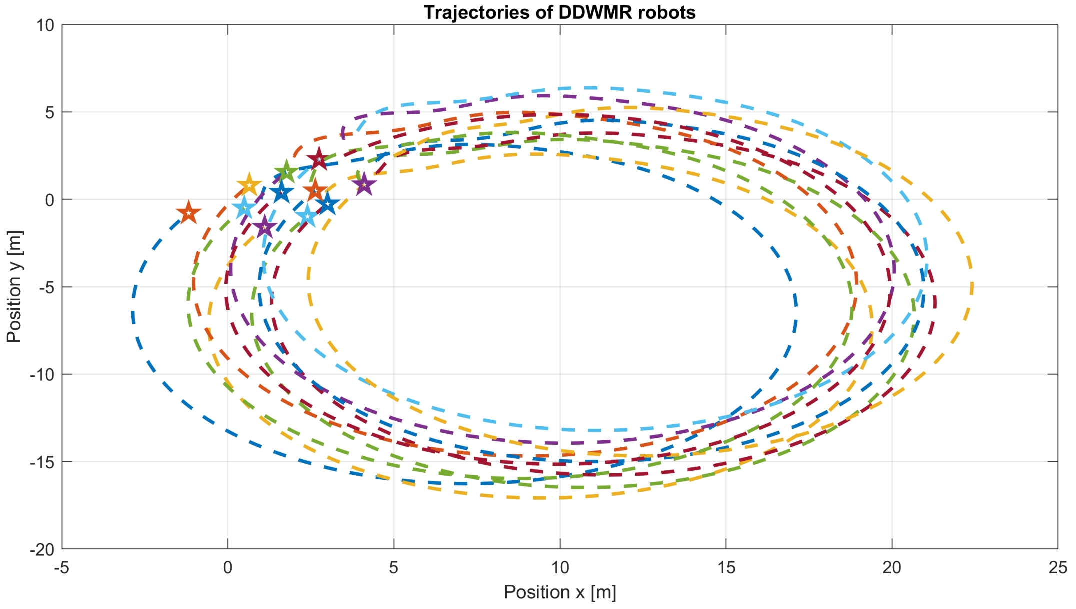


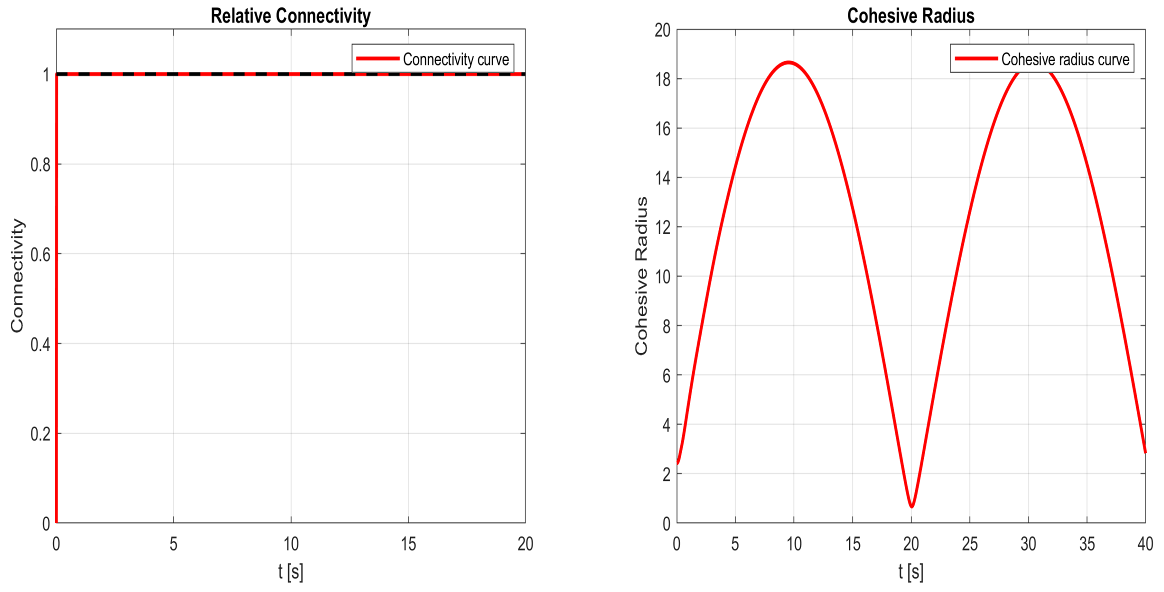
Figure 6-3: The trajectories of the mobile robots and the virtual leader, where the stars indicate the position of the robots at the end of the simulation.

The control parameters are appropriately chosen to be $\gamma_\phi = 1$, $\gamma_v = 5$, $\gamma_\omega = 5$ and $\gamma_\theta = 10$ guided by the theoretical results such that the best performance and a stable flocking system is obtained. An important remark on this concerning the stability is that the control parameters have to be carefully chosen such that the control bandwith of the system does not operate in the frequency region of the uncertainties and high frequency unmodelled dynamics which might lead to an unstable or more oscillating system.
From Figure 6-1, which shows the consensus graphs of the robots in the network, it is evident that the agents (asymptotically) move with the same velocity vector and orientation as the virtual leader (node 0), respectively. It is verified that no inter-agent collisions occur by the simulation performed in Figure 6-2 and from a numerical analysis. Figure 6-3 shows the trajectories of the differential-drive WMRs and the virtual leader, respectively. Evidently, from Figure 6-2 and Figure 6-3, respectively, it can be concluded that the agents move cohesively in a flock.

Verification of Flocking To verify whether a group of particles performs leader-follower flocking, similar to Section 5-3, we can calculate some performance quantities along the trajectory of the particles[41]. These quantities are defined in the following and can be used as performance metrics for flocking.

- *Relative Connectivity*: $C(t) = (1/N-1)rank(A(q(t))) \in [0,1]$
- *Cohesive Radius*: $R(t) = \max_{i \in \mathcal{V}} ||p_i(t) - p_c(t)||$, with $p_c(t) = \frac{\sum_{i=1}^{N} p_i}{N}$ denoting the geometrical center of mass of all agents in the flock.

In short, since the rank of a connected Laplacian and adjacency matrix of order N is at most $N-1$, a relative connectivity $C(t)$ of 1 implies that the interaction graph $\widetilde{\mathcal{G}}(t)$ is a spanning tree. Moreover, a flock is cohesive when it has a finite cohesion radius. A more detailed discussion is given in[41].



(a) **(b)**

Figure 6-4: Visualization of relative connectivity and cohesive radius for numerical verification of flocking with connectivity-preserving and leader-follower cohesive flock guidance for intuitive and explanatory purposes.

The initial proximity interaction graph $\widetilde{\mathcal{G}}(0)$ forms a spanning tree. Figure 6-4a shows that the interaction network $\widetilde{\mathcal{G}}(t)$ is a spanning tree for all time $t \geq 0$. Furthermore, Figure 6-4b shows that the leader-follower flock is a cohesive group by having a finite cohesive radius. By integrating all the arguments above, we can conclude that the distributed leader-follower flocking with connectivity-preservation of 10 non-holonomic EL networked mobile robots under a proximity graph is achieved, which could not directly be concluded from the theoretical analysis of Theorem 3.

Chapter 7

Flocking Algorithm for Non-Holonomic Networked Euler-Lagrange Systems With Obstacle Avoidance

This chapter discusses a distributed (virtual) leader-following flocking algorithm for formation control of a networked non-holonomic Euler-Lagrange (EL) mobile robot system by including obstacle avoidance capabilities as a complement to the algorithm discussed in Chapter 6. The control algorithm is initially attempted to be analyzed through a theoretical framework by exploiting nonlinear control theory concepts for time-varying (non-autonomous) systems.
However, this is a very challenging task and extensive theoretical results on flocking with obstacle avoidance is not yet achieved in the flocking literature[41;51]. Therefore, the performance of the control algorithm discussed in this chapter will mainly be evaluated and verified through a numerical simulation study guided by the obtained theoretical results.

7-1 Flocking Algorithm for Non-Nolonomic Networked Euler-Lagrange Systems With Obstacle Avoidance

In this section, we complement to the study on the distributed (virtual) leader-following flocking problem in the previous chapter by including obstacle avoidance capabilities that do not distinguish between mobile robots and obstacles in proximity, respectively. More formally, all detected objects are regarded as obstacles in the proposed algorithm, inspired by[51].
The ability to distinguish between a robot neighbor and an obstacle is not as easy for mobile robots as it is for animals, such as birds in their flocking behavior. A reason for this is that robots are commonly equipped with laser range finders, that cannot distinguish between obstacles and other robots. This is due to the fact that they only return range data[51].

Problem 3 (Flocking with obstacle avoidance based on proximity graphs)**.** *Given a group of networked non-holonomic Euler-Lagrange mobile robots governed by*

$$\begin{cases} \dot{q}_i = \mathcal{S}_i(q_i)v_{p,i} \\ \widetilde{\mathcal{M}}_i(q_i)\dot{v}_{p,i} + \widetilde{\mathcal{V}}_i(q_i,\dot{q}_i)v_{p,i} = \widetilde{\mathcal{B}}_i(q_i)\tau_i \end{cases} \tag{7-1}$$

and sensing radius $R_s > 0$ with $r_t \in (0, R_s)$. Assume that for any initial conditions $[q_i(0), v_{p,i}(0), \dot{v}_{p,i}(0)]^T$, $i = 0, \ldots, N$, that makes $\widetilde{\mathcal{G}}_t(0)$ contain a spanning tree with node 0, i.e. the virtual leader, as the root. Assume that the virtual leader is a neighbor of at least one follower (i.e. node 0 $\in \widetilde{\mathcal{N}}_i^o(t)$ for at least *one i) at each time instant t. In addition, suppose that at the initial time $t = 0$, no inter-agent and agent-obstacle collisions occur. Then, find a fully distributed artificial potential function (APF)-based flocking control algorithm τ_i of the form*

$$\tau_i = h_i(u_i, q_i, \dot{q}_i),\ j \in \widetilde{\mathcal{N}}_i^o(t) \qquad i = 1, \ldots, N \tag{7-2}$$

where h_i is a sufficiently smooth function to be designed as control algorithm with u_i as the flocking protocol, respectively, such that the closed-loop system is dissipative and has the following properties:

(i) collision avoidance and cohesive flock guidance: the networked multi-agent system moves cohesively with the virtual leader while both inter-agent and agent-obstacle collisions are avoided;

(ii) velocity vector and heading angle alignment, i.e. the robots in the flock asymptotically move with the same velocity vector and heading angle as the virtual leader;

- $\lim_{t\to\infty}(v_i - v_j) = 0$, *$i = 1, \ldots, N$ and $j \in \widetilde{\mathcal{N}}_i^o(t)$.*
- $\lim_{x\to\infty}(\omega_i - \omega_j) = 0$, *$i = 1, \ldots, N$ and $j \in \widetilde{\mathcal{N}}_i^o(t)$.*
- $\lim_{x\to\infty}(\theta_i - \theta_j) = 0$, *$i = 1, \ldots, N$ and $j \in \widetilde{\mathcal{N}}_i^o(t)$.*

The objective is to design τ_i and u_i for each mobile agent to achieve leader-follower flocking under a proximity network with obstacle avoidance, where no distinction is made between neighboring robots and obstacles, respectively. This includes cohesive flock guidance (i.e. followers move cohesively with the leader) when all obstacles are avoided, collision avoidance with both agents and obstacles for all time, (asymptotically) velocity vector and orientation matching of the agents in the flock, respectively. The following distributed control algorithm is proposed

$$\begin{cases} u_{1i} = \underbrace{-\gamma_\phi \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \overrightarrow{t}_i}_{\text{APF gradient term}} \underbrace{- \gamma_v \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(v_i - v_j) - \gamma_{rt} \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(p_i - p_0)^T \overrightarrow{t}_i}_{\text{Consensus term}} \\ u_{2i} = \underbrace{-\gamma_\phi \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}^T \overrightarrow{n}_i}_{\text{APF gradient term}} \underbrace{- \gamma_\omega \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\omega_i - \omega_j) - \gamma_\theta \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\theta_i - \theta_j) - \gamma_{rn} \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(p_i - p_0)^T \overrightarrow{n}_i}_{\text{Consensus term}} \end{cases} \tag{7-3}$$

This can be rewritten into

$$u_i = -\underbrace{-\gamma_\phi H_i \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}}_{\text{APF gradient term}} \underbrace{- \gamma_{v_p} \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(v_{p,i} - v_{p,j}) - \gamma_\theta \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\theta_{p,i} - \theta_{p,j})}_{\text{Consensus term}} \tag{7-4}$$
$$\underbrace{- \gamma_r H_i \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(p_i - p_0)}_{\text{Consensus term}}$$

with

$$H_i = \left[\overrightarrow{t}_i \ \overrightarrow{n}_i\right]^T = \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix}$$

$$\widehat{p}_{ij} = R_{obs} \begin{bmatrix} \cos\widehat{\theta}_{ij} \\ \sin\widehat{\theta}_{ij} \end{bmatrix} + O_k(x_k, y_k)$$

$$\widehat{\theta}_{ij} = atan2(y_i - y_j, x_i - x_j) \text{ for } j \in \mathcal{N}_i^o(t)$$

where $\theta_{p,i} = [0, \theta_i]$, $\gamma_\alpha > 0$ are control tuning parameters for $\alpha = \phi, v, \omega, r_t, r_n$ and θ, $p_{ij} = p_i - p_j$, with $p_i = [x_i, y_i]^T$ denoting the position of agent i. Furthermore, $p_0 = [x_0, y_0]^T$ and θ_0 denote the position and orientation angle of the virtual leader, respectively. In addition, $\overrightarrow{t}_i = [\cos\theta_i, \sin\theta_i]^T$

and $\overrightarrow{n}_i = [-\sin\theta_i, \cos\theta_i]^T$ are unit vectors orthogonal to each other in the tangential and normal direction, respectively. $\phi(||p_{ij}||_\sigma)$ is the revised APF (see Remark 22) between agents i and j, that is defined in Definition 6 with its derivative denoted as $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}$ defined in Lemma 12.
We restrict our study to obstacles that are connected convex surfaces. Specifically, we focus on obstacles that are spherical. In the following, we formally define the notion of a virtual robot (i.e. β-agent) and specify the interaction between the different agents. When the mobile robots detect an obstacle, a "virtual robot" is induced on the surface of the obstacles. In fact, the kinematic virtual robot induced on the obstacle surface simply represents the shortest distance from agent i to obstacle j, respectively.
The geometric and kinematic relationship between the mobile robots and neighboring obstacles is illustrated in Figure 7-2, where $O_k(x_k, y_k)$ denotes the position of the obstacle center, $\widehat{q}_{ij}(\widehat{x}_{ij}, \widehat{y}_{ij})$ represents the projection point coordinates of the mobile robot onto the surface of the obstacle, where $\widehat{q}_{ij}$ is $q_i \in \mathcal{N}_i^o(t)$ with $\mathcal{N}_i^o(t)$ the set of obstacles in proximity. $\widehat{p}_{ij}$ (i.e. $p_i \in \mathcal{N}_i^o(t)$) and $\widehat{\theta}_{ij}$ are the position of the generated kinematic virtual robot ($\beta-$agent ij) and the angle measured from the orientation of robot i to the straight line which connects the robot to the obstacle, when an obstacle enters the proximity view of the mobile robots. The total neighboring set of the system $\widetilde{\mathcal{N}}_i^o(t)$ is defined as the following set of detected objects $\widetilde{\mathcal{N}}_i(t) \cup \mathcal{N}_i^o(t)$ with $\widetilde{\mathcal{N}}_i(t) = \{j \in \{0, 1, \dots, N\}\backslash\{i\} : ||p_i - p_j||\} < R_s\}$ that includes the virtual leader and mobile robots and $\mathcal{N}_i^o(t) = \{j \in \{1, \dots, N_{obs}\}\backslash\{i\} : ||p_i - p_j||\} < R_s\}$ includes the obstacles in proximity, respectively. The gradient of the APF is now given as $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}$ for $j \in \widetilde{\mathcal{N}}_i^o(t)$, where the potential between the physical mobile robots and virtual robots is also synthesized into the argument of the total potential of the real robot. As a result, the mobile robot should be capable to steer safely from the virtual robot, and therefore also from the detected obstacles.
It is important to highlight that the difference between the previous flocking algorithms is that we added obstacles to the proximity set where the potential between the mobile robots and obstacles is also synthesized into the argument of the APF. In terms of practical sensing requirements, it is assumed that every mobile robot is equipped with range sensors in order to measure its relative position between the closest point on the obstacle and itself. For this purpose, both radars and laser radars (or ladars) can be used as range sensors[41;51]. Moreover, an extra consensus term is introduced to track the virtual leader in the system such that the flock is attracted to the virtual leader and minimize the chance that the mobile robots get stuck into local minimas and/or deadlocks, respectively (see Appendix B).

Definition 6 ([72])**.** *[Revised Artificial Potential Function (APF)] Motivated by the discussion in Remark 22, the following revised APF is defined*

$$\phi(s) = \frac{1}{s^2} - \ln(R_s^2 - s^2), \qquad 0 \le s \le R_s$$

where $s = ||p_{ij}||_\sigma$, $p_{ij} = p_i - p_j$ and the σ-norm is exploited to have a smooth potential function for its implementation in the flocking control algorithm. This APF $\phi(s)$ illustrated in Figure 7-1a.

Lemma 12 (Derivative of revised APF)**.** *The derivative of the revised APF in this chapter, denoted as $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}$, is used in the flocking control algorithm for obstacle avoidance purposes by introducing (virtual) attractive and repulsive forces, respectively, on the detected objects. The derivative of the APF is defined as*

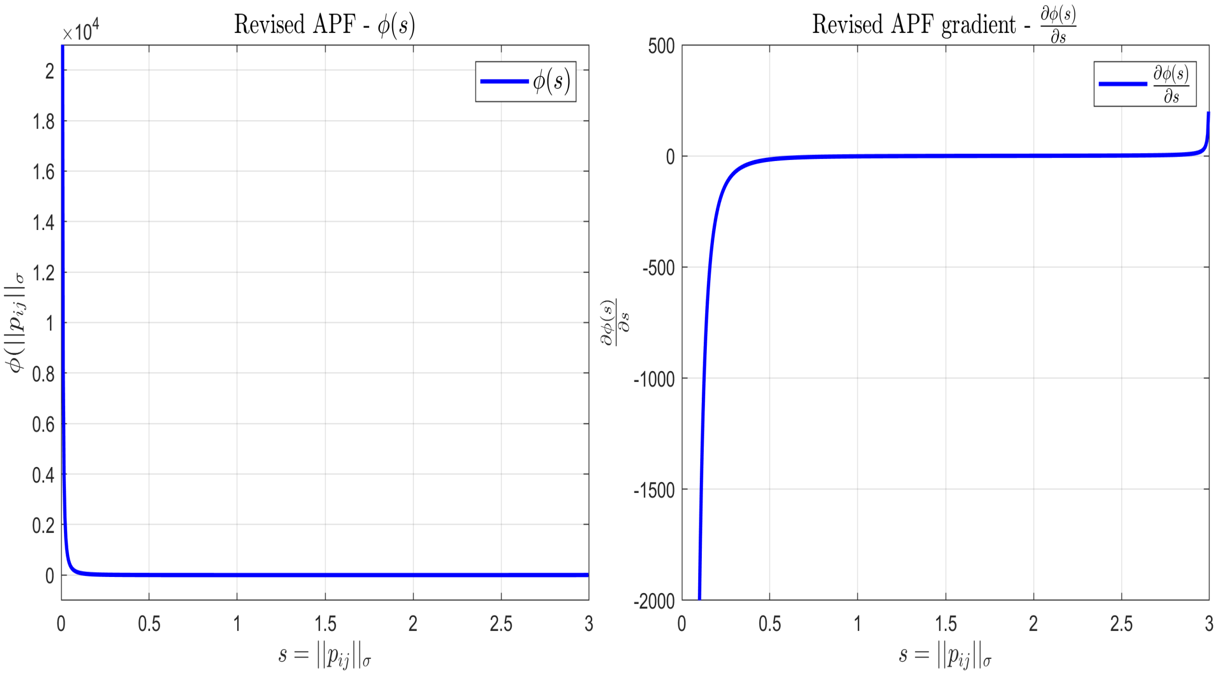
$$\Delta_{p_{ij}}\phi(||p_{ij}||_\sigma) = \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = \Big(\frac{-2}{||p_{ij}||_\sigma^3} + \frac{2||p_{ij}||_\sigma}{R_s^2 - ||p_{ij}||_\sigma^2}\Big)\Big[\Delta_{p_{ij}\{1\}}||p_{ij}||_\sigma \ \Delta_{p_{ij}\{2\}}||p_{ij}||_\sigma\Big]^T$$

where

$$\Delta_{p_{ij}\{1\}}||p_{ij}||_\sigma = \frac{p_{ij}\{1\}}{1 + \epsilon||p_{ij}||_\sigma}$$
$$\Delta_{p_{ij}\{2\}}||p_{ij}||_\sigma = \frac{p_{ij}\{2\}}{1 + \epsilon||p_{ij}||_\sigma}$$

and the relative position vector p_{ij} is defined as $p_{ij} = [x_i - x_j, y_i - y_j]^T$. In addition, the curly brackets $\{.\}$ indicate the selection of the corresponding element in that vector. The revised APF gradient is illustrated in Figure 7-1b.

Proof. The proof of this Lemma can easily be proven by using mathematical insights into the derivative of the Euclidian norm $||.||$ discussed in Section 3-5-1 and properties of the σ-norm discussed in Definition 4. □



(a) The artificial potential function $\phi(s)$ with $s = ||p_{ij}||_\sigma$ for $R_s = 3$.

(b) The gradient of the artificial potential function $\phi(s)$ with $s = ||p_{ij}||_\sigma$ for $R_s = 3$.

Figure 7-1: Visualization of the revised artificial potential function $\phi(s)$ (a) with its derivative $\frac{\partial\phi(s)}{\partial s}$ (b) for intuitive and explanatory purposes.

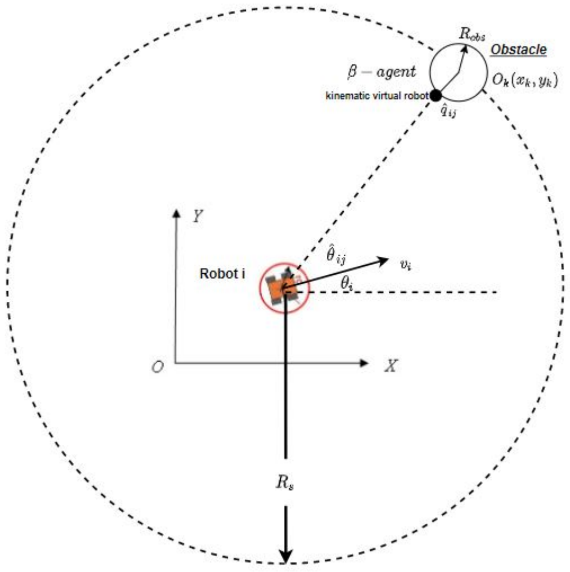


Figure 7-2: The geometric and kinematic relationship between a mobile robot and an obstacle by introducing a virtual robot ($\beta-$agent) as a projection of the robot on the surface of the obstacle when the obstacle enters the limited interaction view of the mobile robot.

A natural question is whether the networked control system obtained is dissipative. It might not be as predictable as the previously discussed control system in case of no virtual leader and without obstacles where only physical robots in proximity interact with each other. An intuitive reason for this is that in free flocking without a virtual leader and/or obstacles, each agent reciprocates the action of its neighboring agents under an undirected interaction network, whereas including a virtual leader and agent-obstacle interactions in the proximity network induces a unidirectional graph (or digraph) as illustrated in Figure 4-4.

Remark 22. *In this scenario, flocking with connectivity preservation is very difficult to be explicitly included to the APF since we do not distinguish between obstacles and mobile robots in the neighborhood. A reason for this is that the obstacles will interfere with the connectivity of the system as they might have to perform split-rejoin maneuvers and this can lead to serious stability problems (especially for large obstacles) as this is strictly related to the connectivity of the interaction network. Therefore, we have to reconsider the APF and the one in*[72] *is used. As a consequence, it is not possible to guarantee connectivity preservation theoretically due to the revising of the APF. However, since the virtual leader contributes to an unifying objective of the flock, we assume that the virtual leader will add robustness in this perspective.*

Remark 23. *In addition to the previous remark, it is important to note that (large) static and/or moving obstacles can form an interference for the connectivity preservation of the proximity graph. When the obstacle is big, in the worst case bigger than the sensing radius, serious conflict of interest between connectivity preserving and obstacle avoidance can occur. This motivates the design of a convenient APF-based flocking algorithm with obstacle avoidance capabilities and highlights its practical limitations.*

Theorem 4 (Flocking of non-holonomic networked Euler-Lagrange mobile agents with obstacle avoidance)**.** *Consider a networked non-holonomic Euler-Lagrange dynamic system* (7-1) *and a constant virtual leader moving with equivalent non-holonomic kinematics (6-3) under the interaction network* $\widetilde{\mathcal{G}}_t(t)$, *Assumption 3 and considering the assumption made in Remark 16. The virtual leader is a neighbor of at least one follower (i.e. node 0* $\in \widetilde{\mathcal{N}}_i^o(t)$ *for at least one i) at each time instant t and no collision occurs among the agents at the initial time.*
Define the "energy-like" Lyapunov function candidate to be $V(. \ , \ t)$ *given by* (D-2)*. Each agent is steered by the fully distributed computed torque nonlinear feedback control algorithm in* (7-5) *and flocking control algorithm* (7-4)*, respectively.*

$$\tau_i = [\widetilde{\mathcal{B}}_i]^{-1}(\widetilde{M}_i \underbrace{a_i}_{=u_i} + \widetilde{V}_i v_{p,i}\,) \tag{7-5}$$

Then, for sufficiently large control parameters γ_α *for* $\alpha = \phi, v_p, \theta, r$, $\dot{V} \leq 0$, *if*

$$-\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}_t(t)]\ ||\widetilde{x}_s||^2 - \gamma_r \widetilde{x}_s^T \overline{\mathcal{H}}_t(t)\widetilde{x}_s + l_e||\widetilde{x}_s|| + l_0||\widetilde{x}_s|| \leq 0 \tag{7-6}$$

where

$$\widetilde{\mathcal{H}}_t(t) = \begin{bmatrix} [\mathcal{H}_t(t) \otimes I_m] & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \overline{\mathcal{H}}_t(t) \begin{bmatrix} 0 & 0 & 0 & 0 & H[\mathcal{H}_t(t) \otimes I_m] \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and H is the column stack vector of H_i, $i = 1, \ldots, N$. *More formally, if* (7-6) *holds, we have that the energy of the system is non-increasing along the trajectory of the collective dynamics of the networked mobile robot control system.*

Proof. See Appendix D.

□

The positive linear-in-$||\widetilde{x}_s||$ terms and the indefinite term $-\gamma_r \widetilde{x}_s^T \overline{\mathcal{H}}_t(t)\widetilde{x}_s$ make it more difficult for $\dot{V} \leq 0$ to hold, where the flocking stability becomes even more questionable. From the theoretical analysis, it can be seen that convenient tuning of the control parameters $(\gamma_{vp}, \gamma_r, \gamma_\phi, \gamma_\theta)$ is required such that the negative part of the expression of $\dot{V}$ dominates the positive linear-in-$||\widetilde{x}_s||$ terms. However, for small values of $||\widetilde{x}_s||$ that are close to zero, the result in (7-6) is hard to satisfy, as in this case the linear terms dominate over the quadratic terms. Furthermore, note that even if V converges, it does not necessarily imply that velocity matching and other desired characteristics are achieved.

Thus, we can not conclude anything from the theoretical analysis and a numerical simulation study is performed in the next section as a complement guided by the obtained theoretical results for verification on the desired properties collision avoidance, velocity vector and orientation matching respectively, as stated in Problem 3.

7-2 Numerical Simulation Study

In this section, numerical simulation results are given to illustrate the effectiveness of the theoretical results obtained in Section 7-1. We consider the flocking of 10 homogeneous mobile robots under a proximity interaction network with obstacle avoidance capabilities without distinguishing between robots and obstacles. The relative non-holonomic EL dynamics of the differential-drive wheeled mobile robots (WMRs) with its system matrices are given in (4-9-4-10) where the proximity architecture is described in Section 4-2.

In the simulation, we consider the same system parameters as in[39], i.e. $m = 6.64$ $[kg]$, $I = 0.06363$ $[kg.m^2]$, $I_w = 0.001$ $[kg.m^2]$, $L = 0.2$ $[m]$, $R = 0.08$ $[m]$ and $d = 0.2$ $[m]$. The initial positions of the 10 mobile robots are, respectively, $p_1(0) = [1, 0]^T$ $[m]$, $p_2(0) = [3, 0.5]^T$ $[m]$, $p_3(0) = [4, 1]^T$ $[m]$, $p_4(0) = [4.5, 1.5]^T$ $[m]$, $p_5(0) = [2, 2.5]^T$ $[m]$, $p_6(0) = [3.5, 3.5]^T$ $[m]$, $p_7(0) = [3.9, 3.9]^T$ $[m]$, $p_8(0) = [4.5, 2.2]^T$ $[m]$, $p_9(0) = [5.0, 2.5]^T$ $[m]$, and $p_{10}(0) = [2.5, 1]^T$ $[m]$. The orientation of each mobile robot is randomly generated by repeating the experiment with different initial conditions until the obstacle avoidance algorithm succeeds for some orientation $\theta_1(0) = 1.93[rad]$, $\theta_2(0) = 1.24[rad]$, $\theta_3(0) = 0.89[rad]$, $\theta_4(0) = 0.82[rad]$, $\theta_5(0) = 1.87[rad]$, $\theta_6(0) = 1.69[rad]$, $\theta_7(0) = 1.57[rad]$, $\theta_8(0) = 0.68[rad]$, $\theta_9(0) = 1.85[rad]$, $\theta_{10}(0) = 1.24[rad]$. The initial velocities of the robots are assumed to be zero. The leader moves under an orientation of $\theta_0(0) = 0.61[rad]$. The constant velocity vector of the leader is assumed to be $v_{p,0} = [3.0, 0]^T$. The sensing radius of the proximity interaction network that represents the communication view of the mobile robots to detect the objects (both obstacles and mobile robots) is given to be $R_s = 3$ $[m]$. The radius of the spherical obstacle is 0.4 [m] and is centered at the coordinates $x = 13$ $[m]$, $y = 10$ $[m]$.

Moreover, the other design parameters used are $\epsilon = 0.2$. The numerical integration scheme fourth order Runge-Kutta (RK4) is used in this simulation. Moreover, the step-size used is $T_s = 0.01$ $[s]$. The RK4 is an effective and widely used method for solving initial-value problems of differential equations and can be used to construct high order accurate numerical methods[73]. The step-size is conveniently chosen such that the RK4 algorithm preserves the stability. This is, it preserves stability provided that the step-size does not become too large. Thus, in practice, the use of high order Runge-Kutta methods allows us to increase the step-size while still obtaining good accuracy but the stability of the algorithms establishes limits to the value of the step-size[73].

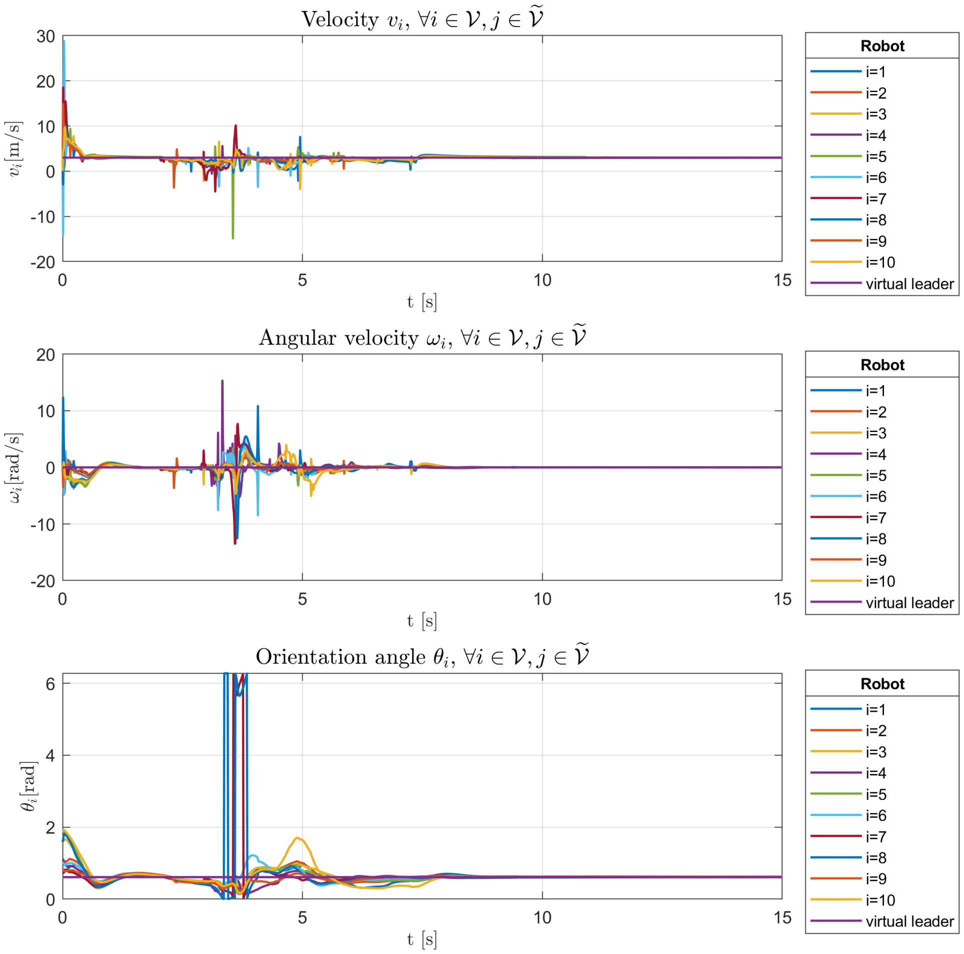
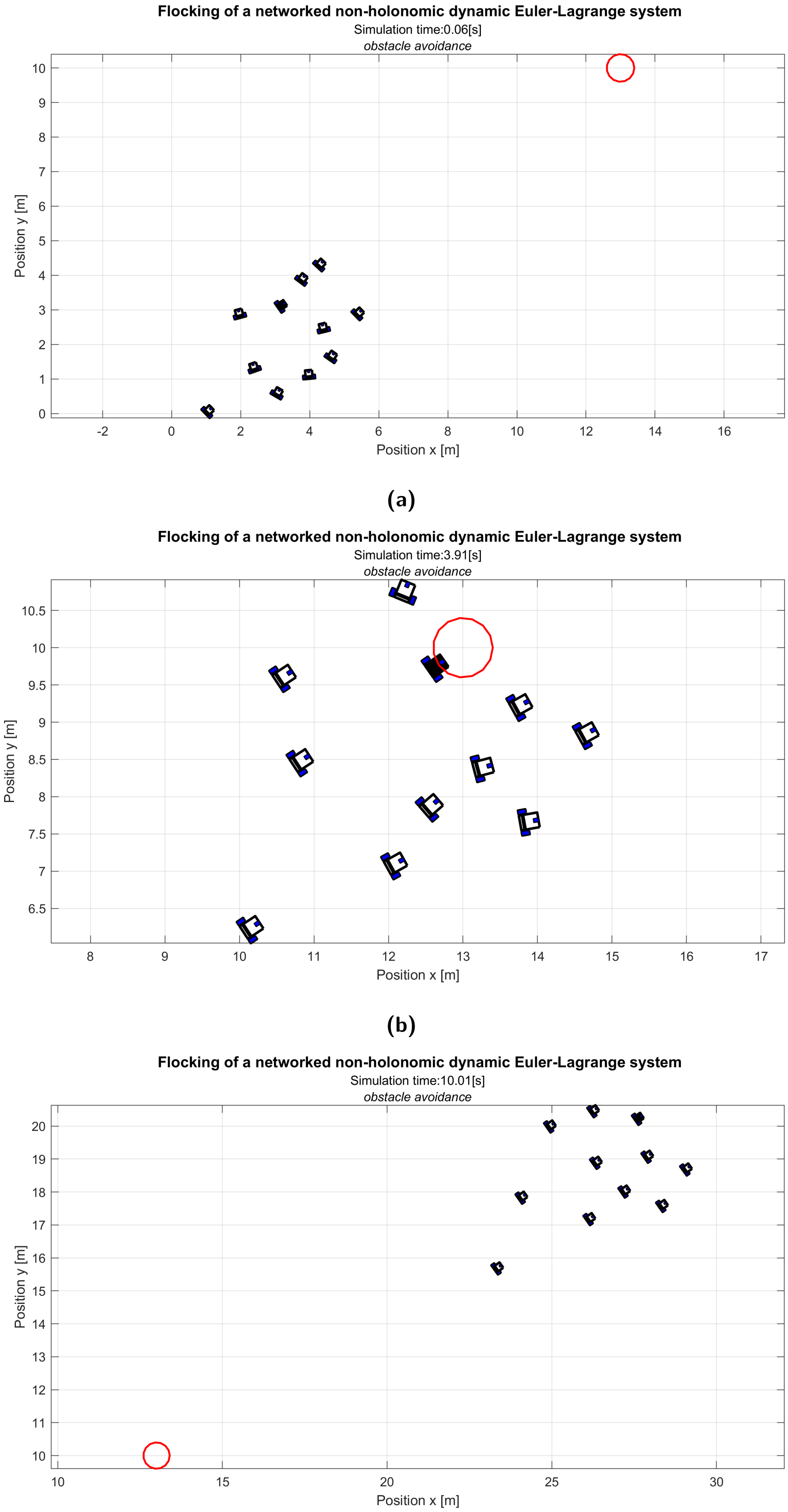


Figure 7-3: Simulation of the velocity vector $[v_i, \omega_i]^T$ and orientation angle θ_i consensus graphs for all the differential-drive WMRs in the networked system.

The control parameters are conveniently chosen as $\gamma_\phi = 1$, $\gamma_v = 5$, $\gamma_\omega = 5$, $\gamma_\theta = 15$, $\gamma_{rt} = 15$ and $\gamma_{rn} = 10$, guided by the theoretical analysis, such that the best performance and a stable flocking system is obtained. An important remark on this concerning the stability is that the control parameters have to be carefully chosen such that the control bandwith of the system does not operate in the frequency region of the uncertainties and high frequency unmodelled dynamics which might lead to an unstable or more oscillating system.

From Figure 7-3, which shows the consensus graphs of the robots in the network, it is evident that the agents (asymptotically) move with the same velocity vector and orientation as the virtual leader, respectively, after avoiding the obstacle. During the interval 3 to 8 seconds, the flocking stability is perturbed as the mobile robots encounter an obstacle on their path and the control algorithm maneuvers the robots around the obstacle, after which flocking is achieved.

Clearly, the obstacles in the environment will cause disturbance to the flocking stability in terms of connectivity, which is of fundamental importance for the stability and consensus properties of the system. Therefore, no flocking is achieved until the obstacles are avoided due to the perturbations on the system as a result of encountering obstacles. From the simulation performed in Figure 7-4 and numerically, it is verified that no inter-agent and agent-obstacle collisions occur. Figure 7-5 shows the trajectories of the differential-drive WMRs. Evidently, from Figure 7-4 and Figure 7-5, it can be concluded that the agents move cohesively in a flock with the virtual leader.



Flocking of a networked non-holonomic dynamic Euler-Lagrange system
Simulation time:0.06[s]
obstacle avoidance

(a)

Flocking of a networked non-holonomic dynamic Euler-Lagrange system
Simulation time:3.91[s]
obstacle avoidance

(b)

Flocking of a networked non-holonomic dynamic Euler-Lagrange system
Simulation time:10.01[s]
obstacle avoidance

(c)

Figure 7-4: Simulation of the 10 differential-drive WMRs and virtual leader (black) under the distributed leader-following obstacle avoidance flocking algorithm (7-4) at (a) t = 0.02 [s], (b) t = 1.0 [s] and t = 5.0 [s].

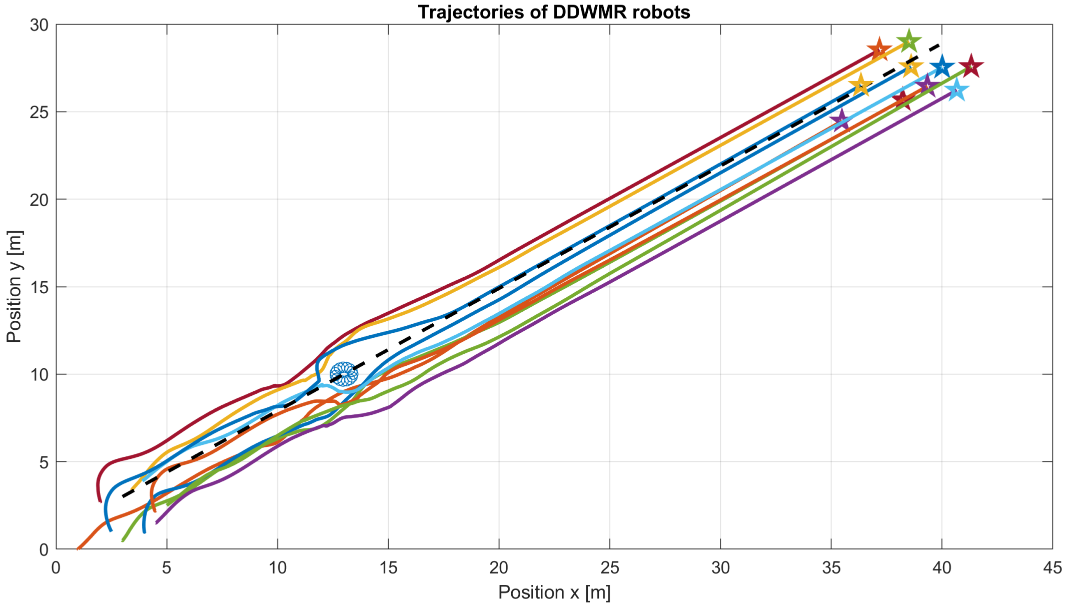


Figure 7-5: The trajectories of the mobile robots where the stars indicate the position of the robots at the end of the simulation and the dotted line is the trajectory of the virtual leader.

The challenging aspect of the flocking algorithm with obstacle avoidance is the case in which the value of the repulsive force from the obstacles might be bigger than the flocking control term. This can result in possible collision between agents in the flock. Therefore, we highlight that proper tuning of the control parameters is of fundamental importance.

When a group of agents faces an obstacle in front of it, the stability will be disturbed at this time instant. The potential function also increases where the gradient of the Lyapunov function candidate $\dot{V}$ can become positive. In this critical situation, however, the multi-robot system will avoid the obstacle in this case and then after the orientation to its goal is free from the obstacle, the group will show formation reconfiguration and self-organizing flocking behavior. Thus, by integrating all the arguments above, we can conclude that the flocking with obstacle avoidance of 10 non-holonomic EL networked mobile robots under a proximity graph is achieved.

It might be interesting to examine whether we can extrapolate the obtained results in this chapter to the aftermath of the algorithm in case of a larger group of mobile robots and/or multiple (including dynamic) obstacles, from a theoretical and computational perspective. Note that by increasing the number of mobile robots and/or obstacles in the flocking algorithm, the positive terms in the theoretical proof in Appendix D will increase and the stability will be even more questionable and challenging to achieve. Also, more challenging features include; the communication complexity and the number of interactions of the proximity network increase, high system dimensionality, more incomplete information and uncertainties, where the sampling time has to be revised for the stability of the system. By including a dense constrained environment with many obstacles, the system is much more likely to get stuck into local minimas. For example, when the distance between two obstacles is too small, the mobile robots get trapped by obstacles forming one concave obstacle, see Appendix B.

To overcome this problem, we need to design a more advanced obstacle avoidance algorithm by adding an emergency algorithm to detect and escape from such a deadlock situation (see e.g.[49]). Moreover, including dynamic obstacles in the flocking algorithm puts more complexities and challenges in the algorithm and its theoretical proof as we have to define the (unknown) complex interaction between the mobile robots and the dynamic obstacles. In addition, a prediction algorithm is required that must be integrated into the flocking algorithm that makes it possible to estimate the states and predict the dynamics of the moving obstacles.

Chapter 8

Conclusion & Future Work

Initially, this chapter gives concluding remarks on the effectiveness of the proposed flocking algorithms. Then, some future work recommendations are made towards the implementation of a practical swarm intelligent networked mobile multi-robot systems (MRSs).

8-1 Conclusion

The study of flocking algorithms for formation control of MRSs in perspective of nonlinear control theory can lead to interesting theoretical results to get insights into the development of swarm intelligent robotics concepts motivated by nature-inspired flocking algorithms. Flocking algorithms have beneficial contributions to solutions of a wide variety of practical applications and support great advantages such as robustness, flexibility, adaptability, self-organizing and self-reconfigurable features to the system with reduced operational costs.

In this study, we primarily focused on the distributed flocking with connectivity preservation of homogeneous networked non-holonomic Euler-Lagrange (EL) systems under a proximity graph with a limited sensing radius and local interactions. It is assumed that the mobile robots are only equipped with limited on-board sensors, i.e. laser range finders (to detect obstacles and robots) and a gyro sensor to measure its position and orientation, respectively. More specifically, three flocking algorithms were designed and evaluated from the perspective of nonlinear control theory, analysis and mathematical modelling of non-holonomic EL systems and the understanding of the heuristic flocking concept in nature.

The first algorithm designed, discussed in Chapter 5 is a connectivity-preserving artificial potential function (APF)-based flocking control protocol for formation control of a networked non-holonomic EL mobile multi-robot system. It is both theoretically and numerically verified that the system obeys the flocking principles: collision avoidance, cohesion and velocity matching, under this algorithm.

Then, a more practical (virtual) leader-follower flocking algorithm is designed and evaluated in Chapter 6. This algorithm builds further on the flocking algorithm in Chapter 5 by including a (virtual) leader that is encapsulated as a reference trajectory for the flock, such that the trajectory is compatible with the non-holonomic kinematic constraints of the following robots. The purpose of the virtual leader is to introduce the mission, i.e. providing guidance to the flock, and it is of fundamental importance for the robustness and fault-tolerance of the flock. From the theoretical analysis, the flocking principles could not directly be concluded. An interesting observation made from the theoretical analysis is that by simply including a virtual leader in the proximity network of the interacting system further complicates the theoretical proof. Intuitively, by interpreting the theoretical results, this can be due to the reason that in virtual leader-follower flocking, the interaction network results in a directed graph with the virtual leader node as root. The interaction between the non-holonomic kinematic virtual leader and the following robots, as a consequence of including a virtual leader in the proximity network, leads to a complex relation between these two agents, while in leaderless flocking each agent reciprocates the behavior of the robots in proximity under an (undirected) interaction network. A numerical simulation study was performed as a complement to verify that the system meets the flocking principles by using some performance metrics in addition to the simulation.

In practice, the multi-robot system moves in a constrained flocking environment where the system also interacts with obstacles in proximity. Therefore, obstacles can not be neglected in the design of the flocking algorithm. In Chapter 7, obstacle avoidance capabilities are added to the control algorithm. In practice, mobile robots are commonly equipped with laser range finders that do not have the capacity to distinguish between robots and obstacles in proximity. For this reason, the algorithm is conveniently designed such that it does not distinguish between robots and obstacles. Note that the proximity network becomes more complex since the mobile robots now also have a (directed) interacting term with the obstacles. This further complicates the theoretical proof for this system and the stability becomes even more questionable. From the theoretical results, the flocking principles can not directly be concluded due to similar reasons as the virtual leader-follower flocking system. The numerical simulation study shows that the flocking principles can be achieved for the chosen parameters and initial conditions. From these results, the effectiveness of the proposed algorithms has been verified except for in the cases where the robots get stuck in local minima where robots are trapped by e.g. multiple obstacles. To overcome this problem, an emergency algorithm is required to detect and escape from such a deadlock situation, however, this was outside the scope of this research.

Concluding, three flocking algorithms have been designed that are both analyzed and validated through a theoretical and computational framework where interesting innovative results have been obtained. The algorithms have proven to be effective for the flocking control of 10 non-holonomic dynamic EL mobile robots. As this work is the first to initialize research in perspective of flocking algorithms for a more realistic and practical modelled multi-agent system (MAS), in specific non-holonomic EL modelled dynamic agents, there is still plenty of room for improvement of the proposed algorithms in order to obtain an algorithm that is realistic and robust enough for practical applicability.

8-2 Future Work

This study has shown the effectiveness of the designed flocking algorithms for non-holonomic EL systems through both a formal theoretical and advanced computational framework where interesting innovative results have been obtained. However, there is still big room for improvement as the ultimate goal is to obtain a flocking control algorithm that is realistic and robust for its application in practice.

Regarding the improvement of the theoretical results, one might focus on obtaining a more rigorous theoretical proof, especially for the virtual leader-following and obstacle avoidance case in Chapter 6 and Chapter 7, as no effective conclusions can be taken from the theoretical proof. A possible choice for improvement could be by redefining the Lyapunov candidate function such that the positive terms in the derivative of the Lyapunov candidate function get cancelled in order to have a negative semi-definite derivative, however, this is a very challenging task. As a result, if the aforementioned is possible, Barbalat's Lemma[53] can be applied to make more effective conclusions on the flocking system. As an additional remark to highlight the importance of a convenient Lyapunov candidate function, most results on nonlinear dynamics simply assume that the dynamics are smooth enough to guarantee existence and unicity of the solutions. Note that this is always the case of good physical system models (at least in classical physics). From a practical point of view, these results essentially mean that the existence of a Lyapunov function candidate to describe a system will guarantee the system's "good behavior" under some mild smoothness assumptions on the dynamics[53].

Also, one might attempt to proof uniform ultimate boundedness (UUB) of the system by using Lyapunov-like theorems, see[25]. However, in order to apply the UUB theorem, the Lyapunov candidate function requires to be a positive definite function[53;25]. This is very challenging as the APF has to be conveniently shifted to obtain $V(0,t) = 0$, since the argument of the APF is the Euclidian distance between the mobile robots in the flock. Also, the results on ultimate boundedness naturally lead to the notion of Lyapunov-like theorems for input-to-state stability (ISS)[25].

To optimize the performance of the system, as we are dealing with many control parameters, and to avoid excessive heuristic optimal tuning, an optimization method is desired such that the control parameters are chosen in agreement with the best performance. In addition, an optimization procedure, such as the distributed Model Predictive Control (DMPC) framework, is desired with state- and input constraints to take the physical limitations of the mobile robots into account for safety purposes. This is of big importance in applications where safety features are fundamental and must be taken into consideration. However, note that this method will increase in computational time.

In this work, we only considered the case where the virtual leader has a constant velocity vector. However, in practice, the mobile robots move in unpredictable and dynamic environments. As a consequence, a flocking algorithm with obstacle avoidance with varying virtual leader velocity[7;15;14] is required for robustness and adaptability in this case. Also, some applications might require posture regulation, i.e. point-to-point motion. It is interesting to study the posture regulation integrated with the flocking algorithm as a future work. The design of posture stabilization for non-holonomic systems has to face serious structural obstruction[44]. Note that the regulation problem is more difficult for non-holonomic systems compared to the tracking problem. This can intuitively be explained by analyzing the number of inputs and outputs. For the posture regulation problem, we have less input commands than the independent variables that determine its configuration, to be controlled, whereas the tracking problem demands a square control problem.

Moreover, depending on the application in practice, one might require an algorithm that is implementable for even bigger groups of networked mobile robots. Note that by adding more robots into the proximity network, the complexity and dimensionality of the interactions increase. This could lead to instability where one might have to revise the sampling time in the numerical simulation study.

An experimental implementation study is desired as future work that will help identifying practical issues, such as e.g. sampling, time-delays, communication dropout, non-identical systems, physical limitations of the robots and bounded disturbances, that constitute points of possible improvements for future studies. It might maybe be shown to work in simulations, which are of great importance as a complement to the theoretical framework for showing its potential operating ability in practice. However, evidently, in practice it is more complicated where we have uncertainties and disturbances and the issues of design and analyses are intertwined, since the design of a nonlinear control system typically involves an iterative process of analysis and design.

The stability of a system does not imply the ability to withstand persistent disturbances of even a small magnitude. The reason for this is that the stability of nonlinear systems is defined with respect to its initial conditions and only the temporary disturbances may be translated as initial conditions. The effects of persistent disturbance on the behavior of nonlinear systems are addressed by the concept of robustness. Therefore, for this reason, an adaptive and/or robust flocking control algorithm has to be designed to obtain a more practical reliable algorithm. In conclusion, a good control system can be obtained only based on effective trade-offs in terms of stability/robustness, stability/performance, cost/performance, and so on.

Appendix A

Flocking Algorithms- MATLAB R2019B

The appendix is to describe the MATLAB code files that are used to perform the numerical simulation studies in this thesis. Each folder is in specific devoted to one of the three proposed algorithms. The codes are written as compact as possible and organized well into one main file.

A-1 Flocking Algorithm (1)

This MATLAB code is to numerically simulate the flocking behavior of non-holonomic Euler-Lagrange (EL) networked dynamical agents under a proximity graph, as discussed in Chapter 5. The following MATLAB .m files are used in the folder Flocking_Algorithm_1

- DDWMR_constants.m, used to describe the properties of the differential-drive wheeled mobile robots (WMRs).
- Flocking_NH_EL_agents.m, denotes the main file.

A-2 Virtual Leader-Follower Flocking Algorithm (2)

This MATLAB code is to numerically simulate the virtual leader-follower flocking behavior of non-holonomic EL networked dynamical agents under a proximity graph, as discussed in Chapter 6. The following MATLAB .m files are used in the folder Flocking_Algorithm_2_Virtual Leader

- DDWMR_constants.m, used to describe the properties of the differential-drive wheeled mobile robots (WMRs).
- Flocking_NH_EL_agents_virtualleader.m, denotes the main file.

A-3 Flocking Algorithm With Obstacle Avoidance (3)

This MATLAB code is to numerically simulate the flocking behavior of non-holonomic EL dynamical agents under a proximity graph with obstacle avoidance properties, as discussed in Chapter 7. The following MATLAB .m files are used:

- DDWMR_constants.m, used to describe the properties of the differential-drive wheeled mobile robots (WMRs).
- Flocking_NH_EL_nodisting.m, denotes the main file.

Appendix B

Local Minima & Deadlocks

The real-time applications of artificial potential functions involve the issue of possible existence of local minima in its potential field. By choosing a proper choice of the step size, the gradient descent algorithm can be proven and guaranteed to converge to a minimum. However, there is no guarantee that this minimum corresponds to the global minimum. In other words, it is not guaranteed that the robot will be able to find a path to its target configuration q_{goal} through this approach.

B-1 Local Minima Problem & Deadlocks

A deadlock is a situation in path planning in which a solution cannot be found, even though one exists. This is typically caused by robots blocking each other's paths where the planner is unable to find a solution in which robots move out of the respected paths of the other robots.

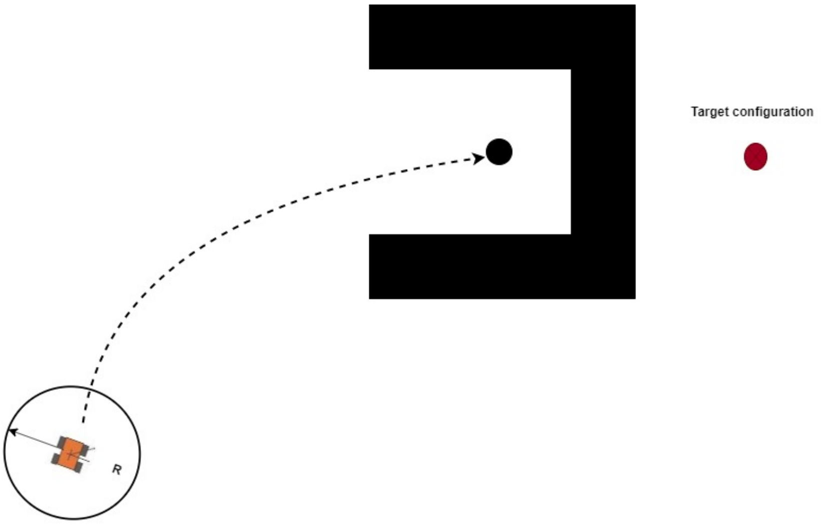


Figure B-1: Local minima problem when the robot approaches a concave shaped obstacle where the robot eventually converges to a local minima where the attractive potential function cancels the repulsive potential function.

Figure B-1 illustrates the local minimum problem as the robot approaches the concave shaped obstacle. The target attracts the robot whereas the lower part of the obstacle steers the robot upward and when the robot detects the upper part of the obstacle, it influences the robot to deflect downwards. These oscillatory effect eventually causes the robot to remain halfway between the obstacle arms. This leads to a scenario where a *conflict of tasks* situation occurs between obstacle avoidance and achieving its group objective. The robot arrives at a point where the repulsive and attractive forces exerted by the obstacle and target cancel each other, i.e. $\Delta U(q^{\star}) = 0$ where $q^{\star}$ does not represent the goal position.

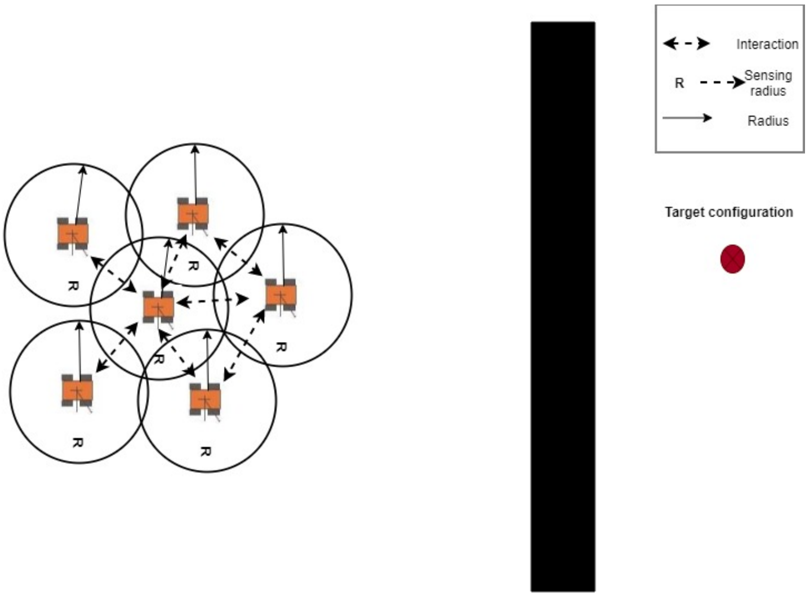


Figure B-2: Local minima problem that leads to a feasibility problem in multi-objective scenarios, i.e. conflict in objective of target reaching or obstacle avoidance.

Another example of a situation that leads to conflicting objectives is illustrated in Figure B-2. In this scenario, when the multi-robot system attempts to pass through the obstacle to reach its group objective, they clearly violated the obstacle avoidance objective. On the other hand, when the multi-robot system avoids the obstacle, the target configuration can not be reached. This raises a *feasibility question* to ensure no conflicted task scenario exists during flocking in constrained environments.
It is important to note that local minima present a drawback to the APF approach and the algorithm is not complete. The local minima problem demands a more sophisticated obstacle avoidance algorithm that attempts to eliminate the conflicting objective approach through e.g. *gyroscopic forces* for obstacle avoidance purposes (e.g.[49]). For a more detailed discussion on local minima and overview of potential functions, the reader might consult[27;1].

Appendix C

Supplementary Theory

Mathematical Background

Throughout this thesis numerous mathematical definitions and theoretical results will be used. This is briefly reviewed in this appendix. The analysis and theoretical framework of advanced non-holonomic systems rely on some essential concepts exploited from mathematical modelling of non-holonomic mechanical robotic systems[34] and controllability for nonlinear non-holonomic systems[5;34;52].

C-1 Mathematical Modelling & Control of Non-Holonomic Mechanical Systems

The aim of this section is to provide a theoretical framework using tools from nonlinear control theory for analysis of non-holonomic mechanical systems. This is an interesting topic since non-holonomic constraints are present in many advanced robotic architectures. In this study, we discuss a unified framework for mathematical modelling of non-holonomic mechanical systems exploiting the Euler-Lagrange (EL) methodology.

The behavior of non-holonomic systems in advanced robotic structures is exceptionally interesting since it involves under-actuated system. Under-actuated systems have very important applications such as free-flying space robots, underwater robots, surface vessels, manipulators with structural flexibility, etc[68]. They are used for reducing weight, cost or energy consumption, while still maintaining an adequate degree of dexterity without reducing the reachable configuration space. Some other advantages of under-actuated systems include less damage while hitting an object, and tolerance for failure of actuators[68].

The kinematic, dynamic modelling and control of non-holonomic mechanical systems are discussed. Some examples of the kinematic and dynamic modelling of non-holonomic systems can be consulted from[34;1].

C-1-1 An Introduction to Non-Holonomic Mechanical Systems

Consider a mechanical system where $q \in \mathcal{Q}$ denotes the vector of generalized coordinates of the system. The configuration space $\mathcal{Q} \in \mathbb{R}^{n_{\mathcal{Q}}}$ is an n-dimensional smooth manifold, locally diffeomorphic (i.e. locally "similar") to an open subset of $\mathbb{R}^n$[34]. Consider a trajectory $q(t) \in \mathcal{Q}$ that is given. The generalized velocity at a configuration q is the vector $\dot{q}$ belonging to the tangent space $\mathcal{T}_q(\mathcal{Q})$.

In interesting applications, the motion of the system is subject to kinematic constraints that originate from the mechanical structure or from the way the system is actuated and controlled. Such constraints are commonly classified into *bilateral* or *unilateral* constraints, i.e. equalities or inequalities, respectively. In addition, the constraints may explicitly or not explicitly depend on time (i.e. *rheonomic* or *scleronomic*) constraints[34]. In this study, we are only interested in constraints that can be expressed as equalities which do not explicitly depend on time. More formally, we are interested in bilateral scleronomic constraints.

- *Holonomic constraints*
  The motion restrictions that can be put in the form of
  $$h_i(q) = 0, \qquad i = 1, \dots, k < n \tag{C-1}$$
  are called *holonomic* constraints. The effect of (C-1) is to limit the feasible configurations of the system to an $(n-k)$-dimensional smooth submanifold of the configuration space $\mathcal{Q}$. Typically, holonomic constraints are introduced by mechanical interconnections between the bodies of the system[34]. A proper way to deal with holonomic constraints is to describe the system configuration by conveniently defining new $n-k$ coordinates on the restricted submanifold that characterizes the actual degrees of freedom of the system[34]. Then, the motion study of the reduced system is identical to the initial system.

- *Non-holonomic constraints*
  The *kinematic constraints* whose expression involves both the generalized coordinates and velocities formulated in the form
  $$a_i(q, \dot{q}) = 0, \qquad i = 1, \dots, k < n \tag{C-2}$$
  confine the permissible motions of the system by restriction of the set of generalized velocities that can be attained at a given configuration. In general, mechanical systems use the *Pfaffian form*[34]
  $$a_i^T(q)\dot{q} = 0, \qquad i = 1, \dots, k < n \quad \text{or} \qquad A^T(q)\dot{q} = 0 \tag{C-3}$$
  to express these constraints linear in the generalized velocities. In addition, it is assumed that the vectors $a_i : \mathcal{Q} \to \mathbb{R}^n$ are both smooth and linearly independent. Note that the holonomic constraints in (C-1) do imply existence of kinematic constraints, however, the opposite might not be necessarily true. In this case, the kinematic constraints in (C-3), are not integrable, i.e. the constraints can not be reformulated into the form (C-1). Those constraints are called *non-holonomic constraints.*

It is important to note that the non-holonomic constraints confine the mobility of the system in a completely different way compared to holonomic constraints, as illustrated in[34]. In case of non-holonomic constraints, the system is still possible to attain any configuration in the configuration space $\mathcal{Q}$. This conclusion can generally be stated for mechanical systems: a mechanical system with n generalized coordinates and k non-holonomic constraints, although the generalized velocities at each point are limited to an $n-k$ dimensional subspace, the accesibility of the entire configuration space $\mathcal{Q}$ is preserved[34]. The rolling disk is a classical example of a non-holonomic system used for illustration, see[34].

Integrability of Constraints

Consider a single Pfaffian constraint written as

$$a^T(q)\dot{q} = \sum_{j=1}^{n} a_j(q)\dot{q}_j = 0 \tag{C-4}$$

This constraint is integrable if there exist a non-vanishing integrable factor $\gamma(q)$ such that

$$\gamma(q)a_j(q) = \frac{\partial h}{\partial q_j}(q), \qquad j = 1, \dots, n \tag{C-5}$$

for some unknown function $h(q)$. The opposite is also valid, i.e. if there exists a $\gamma(q)$ term such that $\gamma(q)a(q)$ is the gradient vector of some function $h(q)$, the constraint (C-4) is integrable[34]. By using Schwarz's theorem[34], the integrability condition (C-5) can be replaced by

$$\frac{\partial(\gamma a_k)}{\partial q_j} = \frac{\partial \gamma a_j}{\partial q_k}, \qquad j, k = 1, \ldots, n \tag{C-6}$$

Note that the integrability condition in (C-6) does not involve the unknown function $h(q)$. In case of multiple kinematic constrains in the form of (C-3), the separate consideration of each constraints is not sufficient to conclude nonholonomy of the system. Actually, it may be possible that $p \leq k$ independent linear combinations of the constraints are integrable[34]. In this case, there exist p independent functions $h_j(q)$ such that

$$span\Big\{\frac{\partial h_1}{\partial q}, \ldots, \frac{\partial h_p}{\partial q}(q)\Big\} \subset span\Big\{a_1^T(q), \ldots, a_k^T(q)\Big\}, \qquad \forall q \in \mathcal{Q}$$

where the system configuration is restricted to an $(n-p)$-dimensional manifold that consists of the level surfaces of the functions h_j, i.e.

$$\{q \in \mathcal{Q} : h_1(q) = c_1, \ldots, h_p(q) = c_p\}$$

on which motion is started[34]. Note that in case of $p = k$, the system is equivalent to a holonomic system. To indicate partial nonholonomy, a subset of the constraint set is integrable, i.e. $1 \leq p < k$. Some relevant examples are illustrated in[34].

Until now, the integrability of multiple kinematic constraints is not evident. The remainder of this section will introduce a different framework for analyzing the integrability conditions of the system. The set of k Pfaffian constraints in (C-3) characterizes the admissible generalized velocities, as contained in the $(n-k)$-dimensional nullspace of matrix $A^T(q)$, at each configuration q[34]. Equivalently, if $\{g_1(q), \ldots, g_{n-k}(q)\}$ denote the basis of this space, the feasible trajectories of the mechanical system can be obtained as solutions of

$$\dot{q} = \sum_{j=1}^{m} g_j(q)u_j = G(q)u, \qquad m = n - k \tag{C-7}$$

for some arbitrary control input $u(t)$. Note that the system in (C-7) can be seen as a driftless nonlinear control system where $q \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ denote the state vector and control input, respectively. This system is *under-actuated* in mechanical point of view since the input dimension is less than the generalized coordinates, i.e. $m < n$. The choice of $G(q)$ is not unique and, accordingly, the components of u will assume different meanings[34].

To conclude about the holonomy or nonholonomy of kinematic constraints, a more convenient way is to study the controllability properties of the corresponding kinematic model. In fact, the following two points will be important, since in the next section tools from nonlinear control theory are reviewed applicable to general nonlinear control systems of the form

$$\dot{x} = f(x) + \sum_{j=1}^{m} g_j(x)u_j \tag{C-8}$$

- The controllability of (C-7), given arbitrary points q_1 and q_2, implies the existence of a steering control input $u(t)$ that drives the system from q_1 to q_2. In other words, a trajectory $q(t)$ consists from q_1 to q_2 in correspondence with the kinematic constraints. Consequently, the system is completely non-holonomic as the configuration space $\mathcal{Q}$ is not confined.

- Conversely, if the kinematic system is not controllable, the reasoning in the previous point is not valid. As a consequence, the kinematic constraints characterize loss of accessibility of the system configuration space $\mathcal{Q}$. Therefore, dependent on the dimension v $(< n)$ of the accessible region, the corresponding constraints are partially or completely integrable[34].

By effectively making use of the controllability approach, it is important to have a framework involving practical conditions to evaluate and verify controllability for the nonlinear control system in (C-8). This motivates using tools from nonlinear control theory which will be discussed in the next section.

C-1-2 Tools from Nonlinear Control Theory

The analysis of nonlinear control systems requires many important concepts from differential geometry[34]. This study concisely reviews relevant introductory definitions and a fundamental result (Frobenius' theorem). Moreover, nonlinear controllability conditions are recalled to characterize non-holonomic constraints. Finally, basic elements for stabilization of nonlinear systems are introduced. It is important to note that the concepts reviewed from nonlinear control theory are briefly introduced, though, it attempts to provide an helpful overview of the tools that can be used for analysis and control purposes of non-holonomic nonlinear control systems used in flocking algorithms. The reader might consult[5;21] for a more detailed and complete analysis.

Differential Geometry

For convenience and simplicity, we denote the state vector with $x \in \mathbb{R}^n$ and the tangent space of $\mathbb{R}^n$ at x by $T_x(\mathbb{R}^n)$. A smooth vector field $g : \mathbb{R}^n \to T_x(\mathbb{R}^n)$ is a smooth mapping assigning to each point $x \in \mathbb{R}^n$ a tangent vector $g(x) \in T_x(\mathbb{R}^n)$[34]. Consider two smooth vector fields g_1 and g_2. Then, the new vector field $[g_1, g_2]$ that is coordinate-dependent can be expressed as

$$[g_1, g_2](x) = \frac{\partial g_2}{\partial x} g_1(x) - \frac{\partial g_1}{\partial x} g_2(x)$$

and is denoted as the *Lie bracket* of g_1 and g_2. To illustrate the relevance of the Lie bracket operation, we consider the following differential equation

$$\dot{x} = g_1(x)u_1 + g_2(x)u_2 \tag{C-9}$$

where g_1 and g_2 denote vector fields. An example of the Lie bracket motion is illustrated in Figure C-1 where the two inputs u_1 and u_2 are not active at the same instant. Therefore, the solution is obtained by decomposition of the two vector fields g_1 and g_2.

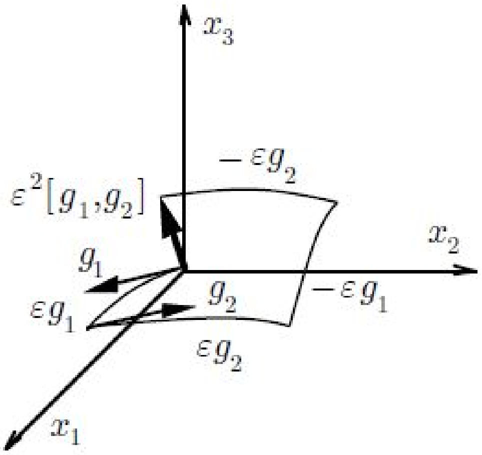


Figure C-1: Lie bracket motion[34].

At every point in the configuration space $\mathcal{Q}$, infinitesimal motion is possible not only in the directions contained in the span of the input vector fields, but also in the directions of their Lie brackets. This is characteristic to the nonlinearity of the input vector fields in the driftless control system (C-9). In addition, by using more complicated input sequences, it is possible to obtain motion in the direction

of higher-order Lie brackets, such as $[g_1,[g_1,g_2]]$[34].
The Lie derivative of $\alpha : \mathbb{R}^n \rightarrow \mathbb{R}$ along g is defined as

$$L_g\alpha(x) = \frac{\partial\alpha}{\partial x}g(x)$$

The most important important properties of the Lie brackets useful in computations are[34]

- *Skew-symmetry*: [f,g] = -[g,f]
- *Jacobi identity*: [f,[g,h]] + [h,[f,g]] + [g,[h,f]] = 0
- *Chain rule*: [αf,βg] = $\alpha\beta$[f,g] + $\alpha(L_f\beta)$g - $\beta(L_g\alpha)$f

with $\alpha, \beta : \mathbb{R}^n \rightarrow \mathbb{R}$. The vector space $\mathcal{V}(\mathbb{R}^n)$ of smooth vector fields on $\mathbb{R}^n$, equipped with the Lie brackets as a product, is called a *Lie algebra*[34].
The smooth distribution Δ that is associated with the m smooth vector fields $\{g_1, g_2\}$ is the map that assigns to each point $x \in \mathbb{R}^n$ a linear subspace of its tangent space[34], i.e.

$$\Delta(x) = span\{g_1(x), \ldots, g_m(x)\} \subset T_x(\mathbb{R}^n)$$

In the sequel, we will neglect the coordinate term (x) for simplification. Then, Δ is said to be non-singular when dim(Δ) $= r$ and constant for all x. Here, r is denoted as the dimension of the distribution Δ. Furthermore, Δ is *involutive* if it is closed under the Lie bracket operation[34]

$$[g_i, g_j] \in \Delta, \qquad \forall g_i, g_j \in \Delta$$

Theorem 5 (Frobenius[34])**.** *A non-singular distribution is completely integrable if and only if it is involutive.*

The proof of this theorem can be consulted from[21].

Controllability of Nonlinear Systems

A nonlinear control system with a smooth drift vector field f in the form of

$$\dot{x} = f(x) + \sum_{j=1}^{m} g_j(x)u_j \tag{C-10}$$

is considered. The system is affine in the class of piecewise-constant functions $\mathcal{U}$ of control inputs u_j and the state vector is denoted as $x \in \mathbb{R}^n$. Unfortunately, a general framework for the natural form of controllability does not exist, i.e. the control system (C-10) is *controllable* if for any choice of x_1, $x_2 \in \mathbb{R}^n$, there exists a finite time T and an input $u : [0, T]$ such that $x(T, 0, x_1, u) = x_2$[34]. Next, we will define other types of structural characterization of the system related to differential geometry concepts.
Define $\mathcal{R}^V(x_0, \tau)$ to be the set of states $\mathcal{E}$, with V a given neighborhood of x_0, for which an input $u : [0, \tau] \rightarrow \mathcal{U}$ exists such that $x(\tau, 0, x_0, u) = \mathcal{E}$ and $x(t, 0, x_0, u) \in V$ for $t \leq \tau$[34]. In fact, $\mathcal{R}^V(x_0, \tau)$ denotes the reachable states at time τ from x_0. Then, the concepts of local accessibility and small-time local controllability can be defined as[34]

- *locally accessible from* x_0, if $\mathcal{R}^V_T(x_0)$ has a non-empty open set Ω, for all neighborhoods V of x_0 and all T.
- *small-time locally controllable from* x_0, if $\mathcal{R}^V_T(x_0)$ has a non-empty neighborhood x_0, for all neighborhoods V of x_0 and all T.

An illustrative example is given in[34] to recognize the differences between these concepts. The interest in concepts as local accessibility and small-time local controllability comes from the existence of mathematical tools to characterize the properties in a system. It is important to remark the following

- The concepts introduced involves only locality. However, globalization of the concepts can be achieved if the statement holds for any $x_0 \in \mathbb{R}^n$.
- Small-time local controllability $\implies$ local accessibility and controllability. In general, the converse does not hold, i.e. local accessibility $\not\Longrightarrow$ controllability. However, in case of no drift vector f in (C-10), the converse does hold.

The *accessibility algebra* $\mathcal{C}$ of the control system (C-10) is defined as the smallest sub-algebra of $\mathcal{V}(\mathbb{R}^n)$ that contains (f,$g_1, \ldots, g_m$[34]). In accordance with the definition, all the Lie brackets and its repeated Lie brackets belong to $\mathcal{C}$. The *accessibility distribution* $\Delta_{\mathcal{C}}$ is defined as

$$\Delta_{\mathcal{C}} = span\{v|v \in \mathcal{C}\}$$

The accessibility algebra can be computed as an iterative procedure

$$\begin{aligned}
&\Delta_{\mathcal{C}} = span\{v|v \in \Delta_i, \quad \forall i \geq 1\} \\
&\text{with} \\
&\Delta_1 = span\{f, g_1, \ldots, g_m\} \\
&\Delta_i = \Delta_{i-1} + span\{[g,v]\}|g \in \Delta_1, v \in \Delta_{i-1}, \quad i \geq 2
\end{aligned}$$

The procedure stops after at most $n - m$ steps, due to the necessity of $dim\Delta_{\mathcal{C}} \leq n$. The accessibility distribution is an important algebraic test involved in verification of local accessibility and small-time local controllability. The following theorems are important in determining algebraic tests that are able to characterize such properties.

Theorem 6 (Chow[34])**.** *If the accessibility rank condition*

$$dim\Delta_{\mathcal{C}}(x_0) = n$$

holds, then the control system (C-10) *is locally accessible from* x_0*. If the accessibility rank condition holds* $\forall x \in \mathbb{R}^n$*, the system is locally accessible. Conversely, if system* (C-10) *is locally accessible, then* $dim\Delta_{\mathcal{C}}(x) = n$ *holds in an open and dense subset of* $\mathbb{R}^n$*.*

The theorem of Chow can be applied to driftless control systems, i.e. control systems in the form of (C-9), as a sufficient condition for controllability. It is important to note that, when the kinematic system is controllable, its dynamic extension is also controllable[34]. Trivially, the converse is also valid.

Theorem 7 (Sussmann[34])**.** *Assume that* $dim\Delta_{\mathcal{C}}(x_0) = n$ *and that, for any bracket* $v \in \Delta_{\mathcal{C}}$ *such that* $\delta^0(v)$ *is odd, and* $\delta^1(v), \ldots, \delta^m(v)$ *are even,* v *may be written as a linear combination of brackets of lower degree. Then, system* (C-10) *is small-time locally controllable from* x_0*.*

For small-time local controllability, the only sufficient condition that exist is the theorem above based on the following concept[34]: Consider a vector field $v \in \Delta_{\mathcal{C}}$ obtained as a (repeated) Lie bracket of the system vector fields, and denote by $\delta^0(v), \delta^1(v), \ldots, \delta^m(v)$ the number of occurrences of $g_0 = f, g_1, \ldots, g_m$, respectively, in v. The degree of the bracket v is defined as $\sum_{i=0}^{m} \delta^i(v)$. For concluding remarks and more detailed information on the discussed concepts, the reader is referred to[34].

Stabilizability of Nonlinear Systems

The problem of *stabilization* of the control system (C-10) involves the design of a feedback control law formulated as

$$u = \alpha(x) + \beta(x)v, \qquad u, v \in \mathbb{R}^m \tag{C-11}$$

with the objective to obtain closed-loop asymptotic stability of an equilibrium point x_e or admissible trajectory $x_e(t)$. The design of feedback control laws are particularly relevant in motion control to counteract disturbances and uncertainties in the system. The tracking case requires the stabilized trajectories to be in accordance with the admissible motions of the system. In particular, this is important in non-holonomic systems due to the involved kinematic constraints that preclude following a generic trajectory.

For linear systems, recall from control theory, that controllability of the system implies asymptotic stability (in fact, even exponential stability) by a smooth state feedback control law[34]. Note that for nonlinear systems only local results may be obtained by analysis of the approximate linearization of the system in a neighborhood of x_e

$$\delta\dot{x} = \frac{\partial f}{\partial x}(x_e)\delta x + \sum_{j=1}^{m} g_j(x_e)u_j = A_e\delta x + B_e\delta u \tag{C-12}$$

with $\delta x = x - x_e$ and $\delta u = u - u_e$. Then, if the approximate linearization is controllable, the nonlinear system in (C-10) can locally be stabilized. This is a sufficient condition, though, not necessary. In the critical scenario, i.e. when the approximate linearization has zero real part uncontrollable eigenvalues, it is not possible to conclude anything on the local controllability of the nonlinear system. In some cases, however, one may use *necessary* conditions that imply the existence of a C^1 stabilizing feedback law to gain insight into the critical case[34]. A fundamental result that is in particular useful in this framework is discussed below.

Theorem 8 (Brockett[34]). *If the system*

$$\dot{x} = \phi(x, u)$$

admits a C^1 feedback u=u(x) that makes x_e asymptotically stable, then the image of the map

$$\phi : \mathbb{R}^n \times \mathcal{U} \to \mathbb{R}^n$$

contains some neighborhood of x_e.

Some remarks on the obtained results are given in[34]. Concluding, under-actuated systems ($m < n$) cannot be stabilized by continuously differentiable static feedback laws as a consequence of the Brockett's necessary conditions[5]. This imposes major difficulties in the design of feedback control laws for non-holonomic systems. More information is discussed in Section 3-1-2.

C-2 Signal Norms

One way to describe the performance of a control system might be in terms of the size of certain signals of interest[12]. For example, for the performance of a tracking system, one might measure the performance by the size of the error signal. Here, we look at different ways of defining the size of a signal, i.e. at different norms for signals. For a continuous-time vector signal $x \in \mathbb{R}^n$ its $\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_\infty$ signal norm are defined as

- $||x(.)||_1 = \int_0^\infty ||x(t)||_1 \, d_t = \int_0^\infty (|x_1(t)| + \ldots + |x_n(t)|) \, d_t$
- $||x(.)||_2 = \sqrt{\int_0^\infty ||x(t)||_2^2 \, d_t} = \sqrt{\int_0^\infty (x_1^2(t) + \ldots + x_n^2(t)) \, d_t}$

- $||x(.)||_\infty = \sup_{t \geq 0} ||x(t)||_\infty = \sup_{t \geq 0} \sup_{1 \leq i \leq n} |x_i(t)|_\infty$

Corresponding to a norm, it is possible to define the space of all the signal which are bounded in that norm. This will be exploited later on for the theoretical stability analysis of the flocking algorithms. For the signal norms $\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_\infty$, we have[12].

(i) $x \in \mathcal{L}_1 \cap \mathcal{L}_\infty \implies x \in \mathcal{L}_p$ for $p \in [1, \infty)$.

(ii) $x \in \mathcal{L}_1$ or $x \in \mathcal{L}_2$ may NOT imply ($\not\Rightarrow$) $x \in \mathcal{L}_\infty$ or $\lim_{t \to \infty} x(t) = 0$.

It is important to note that a signal can be bounded with respect to a certain norm, but unbounded with respect to a different norm[12].

Appendix D

Theoretical Proof of Theorem 4

Chapter 6

In this appendix, we will proof the theoretical results of Theorem 4 in Chapter 7.

Proof. Using the computed torque nonlinear feedback control algorithm (7-5) into (7-1), a state-space model of the closed-loop system can compactly be written in companion form as

$$\dot{x}_i = f_i(x_i) + g_i(x_i)u_i = \begin{bmatrix} \mathcal{S}_i(q_i)v_{p,i} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I_m \end{bmatrix} u_i \tag{D-1}$$

Now, consider the following lower bounded "energy-like" function as a Lyapunov function candidate $V(t) = V_1(t) + V_2(t) + V_3(t) + V_4(t)$, where

$$\begin{aligned} V(p, \widetilde{\theta}_p, \widetilde{v}_p, t) = & \underbrace{\frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)} \phi(||p_{ij}||_\sigma) + \sum_{i=1}^{N}\phi(|p_{i0}||_\sigma)}_{V_1(t)} + \underbrace{\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i^o(t)} \phi(||p_{ij}||_\sigma)}_{V_2(t)} + \underbrace{\frac{1}{2}\sum_{i=1}^{N}\widetilde{v}_p^T{}_{,i}\,\widetilde{v}_{p,i}}_{V_3(t)} \\ & + \underbrace{\frac{1}{2}\widetilde{\theta}_p^T[\mathcal{H}_t(t)\otimes I_m]\widetilde{\theta}_p}_{V_4(t)} \end{aligned} \tag{D-2}$$

For the importance of maintaining the connectivity of the directed proximity graph $\widetilde{\mathcal{G}}_t(t)$ (i.e. $\widetilde{\mathcal{G}}_t(t)$ is a spanning tree for all $t \geq 0$) and to avoid collision between neighboring agents, the non-negative potential function $V_1(t)$ is defined. The derivative of $V_1(t)$ is given by

$$\begin{aligned} \dot{V}_1 &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)} \Big(\dot{p}_i^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \dot{p}_j^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}\Big) + \sum_{i=1}^{N}\Big(\dot{p}_i^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i} - \dot{p}_0^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i}\Big) \\ &= \sum_{i=1}^{N}\dot{p}_i^T\sum_{j\in\mathcal{N}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \sum_{i=1}^{N}\Big(\dot{p}_i^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i} - \dot{p}_0^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i}\Big) \\ &= \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)}(\dot{p}_i - \dot{p}_0)^T\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \sum_{i=1}^{N}\dot{\widetilde{p}}_i^T\frac{\partial\phi(||p_{i0}||_\sigma)}{\partial p_i} \\ &= \sum_{i=1}^{N}\dot{\widetilde{p}}_i^T\sum_{j\in\widetilde{\mathcal{N}}_i(t)}\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} \end{aligned} \tag{D-3}$$

by using Lemma 3.1 in[7] and the fact that $\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = -\frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_j}$ to obtain the second equality term, and using the fact that $\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)} \dot{p}_0^T \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = \dot{p}_0^T \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = 0$ to obtain the third equality term.

The derivative of the second part of the Lyapunov candidate function $V_2(t)$ is given by

$$\begin{aligned}\dot{V}_2 &= \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i^o(t)} \Big(\dot{p}_i^T \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} - \underbrace{\dot{p}_j^T \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}\Big)}_{=0} \\ &= \sum_{i=1}^{N}\dot{\tilde{p}}_i^T \sum_{j\in\mathcal{N}_i^o(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} + \dot{p}_0^T \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i^o(t)} \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}\end{aligned} \tag{D-4}$$

where we have used the fact that $\dot{p}_j^T \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i} = 0$ since $\dot{p}_j^T \perp \frac{\partial\phi(||p_{ij}||_\sigma)}{\partial p_i}$ $j \in \mathcal{N}_i^o(t)$. In other words, the repulsive potential force that steers the mobile robot away from the obstacle is always orthogonal to the velocity of the "virtual robot" on the surface of the obstacle. Therefore, the corresponding dot product is equal to zero. Then, the time derivative of $V_3(t)$ can be formulated as

$$\begin{aligned}\dot{V}_3 &= \sum_{i=1}^{N} \widetilde{v}_{p,i}^{\;T}\, \dot{\widetilde{v}}_{p,i} \\ &= \sum_{i=1}^{N} \widetilde{v}_{p,i}^{\;T}\, u_i\end{aligned} \tag{D-5}$$

by using the fact that $\dot{\widetilde{v}}_{p,i} = \dot{v}_{p,i} - \dot{v}_{p,0} = u_i \in \mathbb{R}^m$, since the virtual leader has a constant velocity. Note that the flocking protocol in (7-4) can mathematically equivalent be written as

$$\begin{aligned}u_i = -\underbrace{-\gamma_\phi H_i \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} \frac{\partial\phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i}}_{\text{APF gradient term}} \underbrace{- \gamma_{v_p} \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j}) - \gamma_\theta \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j})}_{\text{Consensus term}} \\ \underbrace{- \gamma_r H_i \sum_{j\in\widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(p_i - p_0)}_{\text{Consensus term}}\end{aligned} \tag{D-6}$$

where $\widetilde{p}_{ij} = \widetilde{p}_i - \widetilde{p}_j$ with $\widetilde{p}_i = p_i - p_0$, $\widetilde{v}_{p,i} = v_{p,i} - v_{p,0}$ and $\widetilde{\theta}_{p,i} = \theta_{p,i} - \theta_{p,0}$, respectively. The time derivative of $V_4(t)$ can be written as

$$\dot{V}_4 = \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{\theta}_p + \frac{1}{2}\gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}_t(t) \otimes I_m] \widetilde{\theta}_p \tag{D-7}$$

Now, the derivative of the Lyapunov function candidate $V(t)$ is given as

$$\dot{V} = \underbrace{\sum_{i=1}^{N} \dot{\widetilde{p}}_i^T \sum_{j \in \widetilde{\mathcal{N}}_i(t)} \frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i}}_{\dot{V}_1} + \underbrace{\sum_{i=1}^{N} \dot{\widetilde{p}}_i^T \sum_{j \in \mathcal{N}_i^o(t)} \frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i} + \dot{p}_0^T \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^o(t)} \frac{\partial \phi(||p_{ij}||_\sigma)}{\partial p_i}}_{\dot{V}_2} + \underbrace{\sum_{i=1}^{N} \widetilde{v}_{p,}{}_i^T\, u_i}_{\dot{V}_3} \tag{D-8}$$

$$+ \underbrace{\gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{\theta}_p + \frac{1}{2} \gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}_t(t) \otimes I_m] \widetilde{\theta}_p}_{\dot{V}_4}$$

$$= \sum_{i=1}^{N} \dot{\widetilde{p}}_i^T \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} + \dot{p}_0^T \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} + \sum_{i=1}^{N} \widetilde{v}_{p,}{}_i^T \Big(-\gamma_\phi H_i \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} -$$

$$\gamma_{v_p} \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j}) - \gamma_\theta \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j}) - \gamma_r H_i \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(p_i - p_0) \Big)$$

$$+ \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{\theta}_p + \frac{1}{2} \gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}_t(t) \otimes I_m] \widetilde{\theta}_p$$

$$= \sum_{i=1}^{N} \Big(\dot{\widetilde{p}}_i^T - \gamma_\phi v_p^T{}_{,i} H_i \Big) \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} + \dot{p}_0^T \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp} \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j})$$

$$- \gamma_\theta \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j}) - \gamma_r \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} H_i \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(p_i - p_0)$$

$$+ \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{\theta}_p + \frac{1}{2} \gamma_\theta \widetilde{\theta}_p^T [\dot{\mathcal{H}}_t(t) \otimes I_m] \widetilde{\theta}_p \tag{D-9}$$

Using the fact that $\dot{\widetilde{p}}_i^T = \dot{p}_i^T - \dot{p}_0^T = v_p^T{}_{,i} H_i - v_p^T{}_{,0} H_0$ and $\widetilde{v}_p^T{}_{,i} H_i = v_p^T{}_{,i} H_i - v_p^T{}_{,0} H_i$, where $p_i = [x_i, y_i]^T$ denotes the position of the center of mass (COM), and by following the realistic assumption made in Remark 16, we can formulate the derivative of the Lyapunov function candidate as

$$\dot{V} = \sum_{i=1}^{N} \Big(\dot{\widetilde{p}}_i^T - \gamma_\phi \widetilde{v}_p^T{}_{,i} H_i \Big) \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} + \dot{p}_0^T \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} - \gamma_{vp} \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{v}_{p,i} - \widetilde{v}_{p,j})$$

$$- \gamma_r \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} H_i \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(p_i - p_0) - \gamma_\theta \sum_{i=1}^{N} \widetilde{v}_p^T{}_{,i} \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} a_{ij}(t)(\widetilde{\theta}_{p,i} - \widetilde{\theta}_{p,j}) + \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{\theta}_p$$

$$= \sum_{i=1}^{N} \Big(v_p^T{}_{,i} H_i - v_p^T{}_{,0} H_0 - \gamma_\phi (v_p^T{}_{,i} H_i - v_p^T{}_{,0} H_i) \Big) \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i} + \dot{p}_0^T \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^o} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i}$$

$$- \gamma_{vp} \widetilde{v}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{v}_p - \gamma_r \widetilde{v}_p^T H [\mathcal{H}_t(t) \otimes I_m] \widetilde{p} - \gamma_\theta \widetilde{v}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{\theta}_p + \gamma_\theta \dot{\widetilde{\theta}}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{\theta}_p \tag{D-10}$$

where $\widetilde{v}_p$ and $\widetilde{p}$ are column stack vectors of $\widetilde{v}_{p,i}$ and $\widetilde{p}_i$, $i = 1, \ldots, N$, respectively. $\mathcal{H}_t = \mathcal{L} + \Lambda + \Lambda_o$ denotes the interaction network of system where $\mathcal{L}$ is the Laplacian matrix, $\Lambda = diag\{a_{10}, \ldots, a_{N0}\}$ and $\Lambda_o = diag\{a_{1j}, \ldots, a_{Nj}\}$, $j \in \mathcal{N}_i^o$. Furthermore, H is a column stack vector of H_i, $i = 1, \ldots, N$ and I_m is the $m \times m$ identity matrix. Now, by choosing $\gamma_\phi = 1$ and noticing that the last two terms are equivalent to each other, the derivative of the Lyapunov function candidate is given as

$$\dot{V} = \underbrace{\sum_{i=1}^{N} \Big(v_p^T{}_{,0} (H_i - H_0) \Big) \sum_{j \in \widetilde{\mathcal{N}}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i}}_{(1)} + \underbrace{\dot{p}_0^T \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^o(t)} \frac{\partial \phi(||\widetilde{p}_{ij}||_\sigma)}{\partial p_i}}_{(2)}$$

$$\underbrace{- \gamma_{vp} \widetilde{v}_p^T [\mathcal{H}_t(t) \otimes I_m] \widetilde{v}_p}_{(3)} - \gamma_r \widetilde{v}_p^T H [\mathcal{H}_t(t) \otimes I_m] \widetilde{p} \tag{D-11}$$

Note that (1) and (2) are upper bounded by Assumption 3, the definition of the artificial potential function (APF) and from the fact that the orthogonal rotation matrix $||H_i|| = 1$ is always bounded. Moreover, by the positive (semi)-definiteness of the interaction graph $\mathcal{H}_t(t)$ (see Figure 4-4), it can be concluded that (3) is already negative (semi)-definite, i.e. $-\gamma_{vp}\widetilde{v}_p^T[\mathcal{H}_t(t) \otimes I_m]\widetilde{v}_p \leq 0$. Now, by algebraic manipulation, we can write the expression (D-11) equivalently as

$$\dot{V} = \underbrace{\begin{bmatrix} v_p^T{,_0}\,(H_1 - H_0) & v_p^T{,_0}\,(H_2 - H_0) & \dots & v_p^T{,_0}\,(H_N - H_0) \end{bmatrix}}_{\widetilde{v}_e^T} \underbrace{\begin{bmatrix} \sum_{j \in \widetilde{\mathcal{N}}_1^o(t)} \frac{\partial\phi(||\widetilde{p}_{1j}||_\sigma)}{\partial p_1} \\ \sum_{j \in \widetilde{\mathcal{N}}_2^o(t)} \frac{\partial\phi(||\widetilde{p}_{2j}||_\sigma)}{\partial p_2} \\ \vdots \\ \sum_{j \in \widetilde{\mathcal{N}}_N^o(t)} \frac{\partial\phi(||\widetilde{\widehat{p}}_{Nj}||_\sigma)}{\partial p_N} \end{bmatrix}}_{\Delta_{p_i}\phi_s}$$

$$\underbrace{\begin{bmatrix} \dot{p}_0^T & \dot{p}_0^T & \dots & \dot{p}_0^T \end{bmatrix}}_{\dot{P}_0^T} \underbrace{\begin{bmatrix} \sum_{j \in \mathcal{N}_1^o(t)} \frac{\partial\phi(||\widetilde{p}_{1j}||_\sigma)}{\partial p_1} \\ \sum_{j \in \mathcal{N}_2^o(t)} \frac{\partial\phi(||\widetilde{p}_{2j}||_\sigma)}{\partial p_2} \\ \vdots \\ \sum_{j \in \mathcal{N}_N^o(t)} \frac{\partial\phi(||\widetilde{p}_{Nj}||_\sigma)}{\partial p_N} \end{bmatrix}}_{\Delta p_i \phi_{s,o}} -\gamma_{vp}\widetilde{v}_p^T[\mathcal{H}_t(t) \otimes I_m]\widetilde{v}_p - \gamma_r \widetilde{v}_p^T H[\mathcal{H}_t(t) \otimes I_m]\widetilde{p} \tag{D-12}$$

Then, by conveniently introducing a new variable $\widetilde{x}_s = [\widetilde{v}_p, \Delta_{p_i}\phi_s, \Delta_{p_i}\phi_{s,o}, \widetilde{\theta}_p, \widetilde{p}]^T$, we can formulate the expression (D-12) as

$$\begin{aligned}
\dot{V} = &-\gamma_{vp} \underbrace{\begin{bmatrix} \widetilde{v}_p & \Delta_{p_i}\phi_s & \Delta_{p_i}\phi_{s,o} & \widetilde{\theta}_p & \widetilde{p} \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} [\mathcal{H}_t(t) \otimes I_m] & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\widetilde{\mathcal{H}}_t(t)} \underbrace{\begin{bmatrix} \widetilde{v}_p \\ \Delta_{p_i}\phi_s \\ \Delta_{p_i}\phi_{s,o} \\ \widetilde{\theta}_p \\ \widetilde{p} \end{bmatrix}}_{\widetilde{x}_s} \\
&- \gamma_r \underbrace{\begin{bmatrix} \widetilde{v}_p & \Delta_{p_i}\phi_s & \Delta_{p_i}\phi_{s,o} & \widetilde{\theta}_p & \widetilde{p} \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & H[\mathcal{H}_t(t) \otimes I_m] \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\overline{\mathcal{H}}_t(t)} \underbrace{\begin{bmatrix} \widetilde{v}_p \\ \Delta_{p_i}\phi_s \\ \Delta_{p_i}\phi_{s,o} \\ \widetilde{\theta}_p \\ \widetilde{p} \end{bmatrix}}_{\widetilde{x}_s} \\
&+ \underbrace{\begin{bmatrix} \widetilde{v}_p & \Delta_{p_i}\phi_s & \Delta_{p_i}\phi_{s,o} & \widetilde{\theta}_p & \widetilde{p} \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} 0 \\ I \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{E_1} \widetilde{v}_e + \underbrace{\begin{bmatrix} \widetilde{v}_p & \Delta_{p_i}\phi_s & \Delta_{p_i}\phi_{s,o} & \widetilde{\theta}_p & \widetilde{p} \end{bmatrix}}_{\widetilde{x}_s^T} \underbrace{\begin{bmatrix} 0 \\ 0 \\ I \\ 0 \\ 0 \end{bmatrix}}_{E_2} \dot{P}_0 \\
= &\underbrace{-\gamma_{vp}\widetilde{x}_s^T \widetilde{\mathcal{H}}_t(t)\widetilde{x}_s}_{\leq 0} - \gamma_r \widetilde{x}_s^T \overline{\mathcal{H}}_t(t)\widetilde{x}_s + \widetilde{x}_s^T E_1 \widetilde{v}_e + \widetilde{x}_s^T E_2 \dot{P}_0
\end{aligned} \tag{D-13}$$

Clearly, the first part of this expression is negative semi-definite, due to the quadratic expression and the positive semi-definiteness of $\widetilde{\mathcal{H}}_t(t)$. In fact, $\widetilde{\mathcal{H}}_t(t)$ is a symmetric and positive semi-definite matrix, therefore, by using some properties of norms for vectors and matrices (see Subsection 3-5-2), we can write the expression (D-13) as

$$\begin{aligned}\dot{V} &\leq -\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}_t(t)]\ ||\widetilde{x}_s||^2 - \gamma_r \widetilde{x}_s^T \overline{\mathcal{H}}_t(t)\widetilde{x}_s + \widetilde{x}_s^T E_1 \widetilde{v}_e + \widetilde{x}_s^T E_2 \dot{P}_0 \\ &\leq -\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}_t(t)]\ ||\widetilde{x}_s||^2 - \gamma_r \widetilde{x}_s^T \overline{\mathcal{H}}_t(t)\widetilde{x}_s + \underbrace{||E_1 \widetilde{v}_e||}_{\leq ||E_1||\ ||\widetilde{v}_e||}\ ||\widetilde{x}_s|| + \underbrace{||E_2 \dot{P}_0||}_{\leq ||E_2||\ ||\dot{P}_0||}\ ||\widetilde{x}_s|| \\ &\leq -\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}_t(t)]\ ||\widetilde{x}_s||^2 - \gamma_r \widetilde{x}_s^T \overline{\mathcal{H}}_t(t)\widetilde{x}_s + l_e||\widetilde{x}_s|| + l_0||\widetilde{x}_s|| \qquad \text{(D-14)}\end{aligned}$$

where we have assumed that $||\widetilde{v}_e|| < l_e$, $||\dot{P}_0|| < l_0$ and the fact that $||E_{1,2}|| = 1$. Therefore, with the introduction of a new auxiliary variable $\widetilde{x}_s$ defined as $\widetilde{x}_s = [\widetilde{v}_p, \Delta_{p_i}\phi_s, \Delta_{p_i}\phi_{s,o}\,, \widetilde{\theta}_p, \widetilde{p}]$, we have that $\dot{V} \leq 0$, if

$$-\gamma_{vp}\lambda_{min}[\widetilde{\mathcal{H}}_t(t)]\ ||\widetilde{x}_s||^2 - \gamma_r \widetilde{x}_s^T \overline{\mathcal{H}}_t(t)\widetilde{x}_s + l_e||\widetilde{x}_s|| + l_0||\widetilde{x}_s|| \leq 0 \qquad \text{(D-15)}$$

More formally, when this holds, the energy of the system is non-increasing (i.e. $\dot{V}(.\ ,\ t) \leq 0$) along the trajectory of the collective dynamics of the networked mobile multi-robot system. □

Bibliography

[1] PRINCIPLES OF ROBOT MOTION, theory, algorithms and implementations, by howie choset et al., MIT press, 2005. xix 603 pp., index, ISBN 0-262-03327-5, 433 references (hb. £38.95). *Robotica*, 24(2):271–271, March 2006.

[2] Jan Carlo Barca and Y. Ahmet Sekercioglu. Swarm robotics reviewed. *Robotica*, 31(3):345–359, jul 2012.

[3] Aabha Barve and Manisha J. Nene. Survey of flocking algorithms in multi-agent systems. 2013.

[4] Levent Bayındır. A review of swarm robotics tasks. *Neurocomputing*, 172:292–321, jan 2016.

[5] A.M. Bloch. *Nonholonomic Mechanics and Control*. Springer New York, 2015.

[6] Hu Cao, Jie Chen, Yutian Mao, Hao Fang, and Huagang Liu. Formation control based on flocking algorithm in multi-agent system. In *2010 8th World Congress on Intelligent Control and Automation*. IEEE, July 2010.

[7] Yongcan Cao and Wei Ren. Distributed coordinated tracking with reduced interaction via a variable structure approach. *IEEE Transactions on Automatic Control*, 57(1):33–48, January 2012.

[8] Chien Chern Cheah, Saing Paul Hou, and Jean Jacques E. Slotine. Region-based shape control for a swarm of robots. *Automatica*, 45(10):2406–2411, October 2009.

[9] Yang Quan Chen and Zhongmin Wang. Formation control: a review and a new consideration. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005.

[10] Xing Chu, Zhaoxia Peng, Guoguang Wen, and Ahmed Rahmani. Distributed formation tracking of multi-robot systems with nonholonomic constraint via event-triggered approach. *Neurocomputing*, 275:121–131, January 2018.

[11] Yi Dong and Jie Huang. Consensus and flocking with connectivity preservation of uncertain euler–lagrange multi-agent systems. *Journal of Dynamic Systems, Measurement, and Control*, 140(9), apr 2018.

[12] dr. Simone Baldi. Adaptive and predictive contol (sc4060). Lecture notes, Delft University of Technology, 2014.

[13] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, September 2004.

[14] Zhi Feng and Guoqiang Hu. Connectivity-preserving flocking for networked lagrange systems with time-varying actuator faults. *Automatica*, 109:108509, nov 2019.

[15] Sheida Ghapani, Jie Mei, Wei Ren, and Yongduan Song. Fully distributed flocking with a moving leader for lagrange networks with parametric uncertainties. *Automatica*, 67:67–76, may 2016.

[16] Chris Godsil and Gordon Royle. *Algebraic Graph Theory*. Springer New York, 2001.

[17] Wang Guanghua, Li Deyi, Gan Wenyan, and Jia Peng. Study on formation control of multi-robot systems. In *2013 Third International Conference on Intelligent System Design and Engineering Applications*. IEEE, January 2013.

[18] Jing Han, Ming Li, and Lei Guo. Soft control on collective behavior of a group of autonomous agents by a shill agent, 2010.

[19] Rached Dhaouadi Ahmad Abu Hatab. Dynamic modelling of differential-drive mobile robots using lagrange and newton-euler methodologies: A unified framework. *Advances in Robotics & Automation*, 02(02), 2013.

[20] Dirk Helbing, Illés Farkas, and Tamás Vicsek. Simulating dynamical features of escape panic. *Nature*, 407(6803):487–490, September 2000.

[21] Alberto Isidori. *Nonlinear Control Systems*. Springer Berlin Heidelberg, 1989.

[22] A. Jadbabaie, Jie Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, June 2003.

[23] Yongnan Jia and Long Wang. Experimental implementation of distributed flocking algorithm for multiple robotic fish. *Control Engineering Practice*, 30:1–11, September 2014.

[24] Kiattisin Kanjanawanishkul. Formation control of mobile robots: Survey. 2016.

[25] Hassan K Khalil. *Nonlinear Control: Global Edition*. Pearson Education Limited, 2015.

[26] Eric Klavins. Communication complexity of multi-robot systems. In *Springer Tracts in Advanced Robotics*, pages 275–291. Springer Berlin Heidelberg, 2004.

[27] Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Norwell, MA, USA, 1991.

[28] Bin Lei, Wenfeng Li, and Fan Zhang. Stable flocking algorithm for multi-robot systems formation control. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*. IEEE, June 2008.

[29] N.E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. In *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228)*. IEEE.

[30] Herbert Levine, Wouter-Jan Rappel, and Inon Cohen. Self-organization in systems of self-propelled particles. *Physical Review E*, 63(1), December 2000.

[31] Qin Li and Zhong-Ping Jiang. Flocking of decentralized multi-agent systems with application to nonholonomic multi-robots. *IFAC Proceedings Volumes*, 41(2):9344–9349, 2008.

[32] Aurelio Lima, Josiel A. Gouvea, Fernando Lizarralde, and Liu Hsu. Trajectory tracking, pose regulation and adaptive formation control of a group of nonholonomic mobile robots. *IFAC Proceedings Volumes*, 47(3):5709–5714, 2014.

[33] Yuanchang Liu and Richard Bucknall. A survey of formation control and motion planning of multiple unmanned vehicles. *Robotica*, 36(7):1019–1047, March 2018.

[34] A. De Luca and G. Oriolo. Modelling and control of nonholonomic mechanical systems. In *Kinematics and Dynamics of Multi-Body Systems*, pages 277–342. Springer Vienna, 1995.

[35] Xiaoyuan Luo, Shaobao Li, and Xinping Guan. Flocking algorithm with multi-target tracking for multi-agent systems. *Pattern Recognition Letters*, 31(9):800–805, July 2010.

[36] Sabato Manfredi. Design of a multi-hop dynamic consensus algorithm over wireless sensor networks. *Control Engineering Practice*, 21(4):381–394, April 2013.

[37] Silvia Mastellone, Dušan M. Stipanović, Christopher R. Graunke, Koji A. Intlekofer, and Mark W. Spong. Formation control and collision avoidance for multi-agent non-holonomic systems: Theory and experiments. *The International Journal of Robotics Research*, 27(1):107–126, January 2008.

[38] Andrew McKenzie, Qingquan Sun, and Fei Hu. Multi-flock flocking for multi-agent dynamic systems. In *Mobile Robots - Control Architectures, Bio-Interfacing, Navigation, Multi Robot Motion Planning and Operator Training*. InTech, December 2011.

[39] Hossein Mirzaeinejad. Optimization-based nonlinear control laws with increased robustness for trajectory tracking of non-holonomic wheeled mobile robots. *Transportation Research Part C: Emerging Technologies*, 101:1–17, April 2019.

[40] T. Nguyen, T. Han, and H. M. La. Distributed flocking control of mobile robots by bounded feedback. In *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 563–568, 2016.

[41] R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, March 2006.

[42] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, September 2004.

[43] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, January 2007.

[44] G. Oriolo, A. De Luca, and M. Vendittelli. WMR control via dynamic feedback linearization: design, implementation, and experimental validation. *IEEE Transactions on Control Systems Technology*, 10(6):835–852, November 2002.

[45] Lynne E. Parker. Path planning and motion coordination in multiple mobile robot teams. 2009.

[46] Petersen & Pedersen. The matrix cookbook, nov 2012.

[47] Wei Ren. Distributed attitude alignment in spacecraft formation flying. *International Journal of Adaptive Control and Signal Processing*, 21(2-3):95–113, 2007.

[48] Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 21(4):25–34, August 1987.

[49] Lorenzo Sabattini, Cristian Secchi, and Cesare Fantuzzi. Collision avoidance for multiple lagrangian dynamical systems with gyroscopic forces. *International Journal of Advanced Robotic Systems*, 14(1):172988141668710, January 2017.

[50] R.O. Saber and R.M. Murray. Consensus protocols for networks of dynamic agents. In *Proceedings of the 2003 American Control Conference, 2003.* IEEE.

[51] Daito Sakai, Hiroaki Fukushima, and Fumitoshi Matsuno. Flocking for multirobots without distinguishing robots and obstacles. *IEEE Transactions on Control Systems Technology*, 25(3):1019–1027, May 2017.

[52] Bruno Siciliano and Oussama Khatib, editors. *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008.

[53] Jean-Jacques E. Slotine and Weiping Li. Applied nonlinear control. 1991.

[54] Housheng Su, Xiaofan Wang, and Zongli Lin. Flocking of multi-agents with a virtual leader part i: with a minority of informed agents. In *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007.

[55] Housheng Su, Xiaofan Wang, and Zongli Lin. Flocking of multi-agents with a virtual leader. *IEEE Transactions on Automatic Control*, 54(2):293–307, February 2009.

[56] Housheng Su, Xiaofan Wang, and Wen Yang. Flocking in multi-agent systems with multiple virtual leaders. *Asian Journal of Control*, 10(2):238–245, 2008.

[57] Herbert G. Tanner, Ali Jadbabaie, and George J. Pappas. Coordination of multiple autonomous vehicles. 2003.

[58] Herbert G. Tanner, Ali Jadbabaie, and George J. Pappas. Flocks of autonomous mobile agents. 2003.

[59] Herbert G. Tanner, Ali Jadbabaie, and George J. Pappas. Flocking in fixed and switching networks. *IEEE Transactions on Automatic Control*, 52(5):863–868, May 2007.

[60] H.G. Tanner. Flocking with obstacle avoidance in switching networks of interconnected vehicles. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*. IEEE, 2004.

[61] S. Tatar. Flocking algorithm for formation control of networked multi-robot systems. Literature study, Delft University of Technology (TU Delft), dec 2019.

[62] John Toner and Yuhai Tu. Flocks, herds, and schools: A quantitative theory of flocking. *Phys. Rev. E*, 58:4828–4858, Oct 1998.

[63] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75(6):1226–1229, August 1995.

[64] Hanlei Wang. Flocking of networked uncertain euler–lagrange systems on directed graphs. *Automatica*, 49(9):2774–2779, sep 2013.

[65] Miaomiao Wang, Housheng Su, Miaomiao Zhao, Michael Z.Q. Chen, and Hongwei Wang. Flocking of multiple autonomous agents with preserved network connectivity and heterogeneous nonlinear dynamics. *Neurocomputing*, 115:169–177, sep 2013.

[66] G. Wen, Z. Duan, H. Su, G. Chen, and W. Yu. A connectivity-preserving flocking algorithm for multi-agent dynamical systems with bounded potential function. *IET Control Theory & Applications*, 6(6):813, 2012.

[67] Wikipedia. Spanning tree — from Wikipedia, the free encyclopedia, 2020. [Online; accessed 7-July-2020].

[68] Rong Xu and Ümit Özgüner. Sliding mode control of a class of underactuated systems. *Automatica*, 44(1):233–241, January 2008.

[69] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, 10(12):399, January 2013.

[70] Hai-Tao Zhang, Chao Zhai, and Zhiyong Chen. A general alignment repulsion algorithm for flocking of multi-agent systems. *IEEE Transactions on Automatic Control*, 56(2):430–435, February 2011.

[71] Qing Zhang, Yaru Hao, Zhengquan Yang, and Zengqiang Chen. Adaptive flocking of heterogeneous multi-agents systems with nonlinear dynamics. *Neurocomputing*, 216:72–77, dec 2016.

[72] Xiao-Wen Zhao, Bin Hu, Zhi-Hong Guan, Chao-Yang Chen, Ming Chi, and Xian-He Zhang. Multi-flocking of networked non-holonomic mobile robots with proximity graphs. *IET Control Theory & Applications*, 10(16):2093–2099, October 2016.

[73] L. Zheng and X. Zhang. Numerical methods. In *Modeling and Analysis of Modern Fluid Problems*, pages 361–455. Elsevier, 2017.

[74] Jiandong Zhu, Jinhu Lu, and Xinghuo Yu. Flocking of multi-agent non-holonomic systems with proximity graphs. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 60(1):199–210, January 2013.

Glossary

List of Acronyms

MASs	multi-agent systems
MAS	multi-agent system
UGVs	Unmanned Ground Vehicles
UAVs	Unmanned Aerial Vehicles
USVs	Unmanned Surface Vehicles
AUVs	Autonomous Underwater Vehicles
MRSs	multi-robot systems
SR	swarm robotics
UVs	unmanned vehicles
DoD	Department of Defense
EL	Euler-Lagrange
WMRs	wheeled mobile robots
WMR	wheeled mobile robot
UAVs	Unmanned Aerial Vehicles
APF	artificial potential function
COM	center of mass
DMPC	distributed Model Predictive Control
2D	2-dimensional
SOS	sum-of-squares
RK4	fourth order Runge-Kutta
UUB	uniform ultimate boundedness
ISS	input-to-state stability