**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**
**Ecole Nationale Polytechnique**

Department of Automatic
Laboratory of Process Control

# Doctoral Thesis in Automatic

Presented by **ALLAM Ahmed**
Magister in Automatic of Military Polytechnic School

Theme

# Control and navigation of UAVs formation (application to Quadrotors)

Examined on 24 November 2021 by a committee composed of:

| | | |
|---|---|---|
| **President** | Mr BOUCHERIT Mohamed Seghir | Professor, ENP |
| **Supervisor** | Mr TADJINE Mohamed | Professor, ENP |
| **Co- Supervisor** | Mr NEMRA Abdelkrim | MCA, EMP |
| **Examiner** | Mr HADDAD Moussa | Professor, EMP |
| **Examiner** | Mr BELHOCINE Mahmoud | Researches Director, CDTA |
| **Examiner** | Mr BENMANSOUR Khelifa | Professor, ESDAT |
| **Examiner** | Mr CHAKIR Messaoud | MCA, ENP |

ENP 2021

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**
**Ecole Nationale Polytechnique**

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

مخبر التحكم في العمليات التصنيعية
Laboratoire de Commande des Processus

Department of Automatic
Laboratory of Process Control

# Doctoral Thesis in Automatic

Presented by **ALLAM Ahmed**
Magister in Automatic of Military Polytechnic School

Theme

# Control and navigation of UAVs formation (application to Quadrotors)

Examined on 24 November 2021 by a committee composed of:

| | | |
|---|---|---|
| **President** | Mr BOUCHERIT Mohamed Seghir | Professor, ENP |
| **Supervisor** | Mr TADJINE Mohamed | Professor, ENP |
| **Co- Supervisor** | Mr NEMRA Abdelkrim | MCA, EMP |
| **Examiner** | Mr HADDAD Moussa | Professor, EMP |
| **Examiner** | Mr BELHOCINE Mahmoud | Researches Director, CDTA |
| **Examiner** | Mr BENMANSOUR Khelifa | Professor, ESDAT |
| **Examiner** | Mr CHAKIR Messaoud | MCA, ENP |

ENP 2021

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**
**Ecole Nationale Polytechnique**

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

مخبر التحكم في العمليات التصنيعية
Laboratoire de Commande des Processus

Département d'Automatique
Laboratoire de commande des processus

# Thèse de Doctorat En Automatique

Présenté par **ALLAM Ahmed**
Magister en Automatique de l'Ecole Militaire Polytechnique

Intitulée

# Commande et navigation d'une formation d'UAVs (application sur les Quadrotors)

Soutenue publiquement le 24 novembre 2021 devant le jury composé de:

| | | |
|---|---|---|
| **Président** | BOUCHERIT Mohamed Seghir | Professeur, ENP |
| **Directeur de Thèse** | TADJINE Mohamed | Professeur, ENP |
| **Co-directeur de Thèse** | NEMRA Abdelkrim | MCA, EMP |
| **Examinateur** | HADDAD Moussa | Professeur, EMP |
| **Examinateur** | BELHOCINE Mahmoud | DR, CDTA |
| **Examinateur** | BENMANSOUR Khelifa | Professeur, ESDAT |
| **Examinateur** | CHAKIR Messaoud | MCA, ENP |

ENP 2021

### ملخص

يمكن وصف نظام مركبات متعددة/ نظام متعدد العملاء(ن م م/ن م ع) تعاوني على أنها مجموعة من العملاء/ المركبات المستقلين يملكون القدرة على صنع القرار و التي تعمل معًا وتسعى إلى هدف مشترك وجماعي (أي سلوكيات المجموعة) بناءً على استشعارهم وأيضا على المعلومات المشتركة بين الوكلاء. تبحث هذه الأطروحة في الأساليب والتقنيات من وجهة نظر التحكم لتعظيم الأداء الكلي المكتسب لنظام الروبوتات عند إنجاز مهمة معينة بشكل تعاوني من حيث الاستقرار والقوة والسرعة ودقة التتبع. الهدف الرئيسي هو تصميم وحدات تحكم تعاونية لتتبع تشكيل متغير الوقت لمجموعة من المركبات المتصلة بالشبكة ، أثناء تتبع مسار مرجعي للتكوين. في الابحاث السابقة ، تم دراسة و بعمق في مناهج التحكم (ن م م) الكلاسيكية مثل الطريقة القائمة على السلوك ، نهج الهيكل الافتراضي وتقنية القائد - التابع. ومع ذلك، فقد ثبت أن هذه الأساليب موحدة في إطار تقنيات التوافق. يشير أسلوب (أو بروتوكول) التوافق إلى قانون تحكم مصمم لمجموعة من العملاء (يتميزون بديناميات خطية) للتوصل إلى اتفاق في بعض المتغيرات ذات الأهمية (مثل الإحداثيات). تم تمديد مشاكل الإجماع إلى التحكم في تشكيل متغير . في هذه الرسالة ، نركز على تصميم وحدات تحكم تتبع (ت م ز) الموزعة لنظام (ن م م) المتصل بالشبكة. كانت المساهمة الرئيسية في ضمان تبادل معلومات الشبكة الموزعة والمقلصة بين أفراد (ن م م). بالإضافة إلى ذلك ، تم تخفيف الافتراضات العملية مثل اتصالية الشبكة. تمت دراسة استقرار (ن م م) بالاعتماد على نظرية ليابونوف. تم توفير عمليات المحاكاة العددية لتأكيد تخمينات التطورات المقترحة مع تطبيقها على أنظمة طائرات رباعية المحركات.

**الكلمات المفتاحية**: التحكم التعاوني ، نظرية الرسم البياني ، نظرية ليابونوف ، أنظمة العملاء المتعددة ، مراقبة التشكيل ، تقنيات التوافق ، طائرات رباعية المحركات ، الروبوتات المتنقل.

### Résumé

Un Système Multi-Véhicules/Multi-Agents SMV/SMAs coopératif peut être caractérisé comme un groupe d'agents/véhicules autonomes ayant la capacité de prise de décision, opérant ensemble et cherchant un objectif commun et de manière collectif (c'est-à-dire à travers des comportements en groupe) en fonction de leurs perceptions et les informations partagées entre les agents. Cette thèse étudie les méthodes et les techniques du point de vue commande pour maximiser les performances globales acquises d'un système robotique lors de l'accomplissement en coopération d'une tâche donnée en termes de stabilité, de robustesse, de rapidité et de précision. L'objectif principal est de concevoir des contrôleurs coopératifs de suivi de formation variant dans le temps pour un groupe de véhicules connectés en réseau, tout en suivant une trajectoire de référence de formation. Dans la littérature, les approches de commande des SMV classiques ont été profondément étudiées, telle que la méthode basée sur le comportement, l'approche de la structure virtuelle et la technique leader-suiveur. Cependant, ces méthodes se sont avérées unifiées dans le cadre des techniques de consensus. La technique de consensus (ou protocole) fait référence à une loi de contrôle conçue pour qu'un groupe d'agents (caractérisé par une dynamique linéaire) parvienne à un accord sur un variable d'intérêt (comme par exemple les états ou les vitesses). Les problèmes de consensus ont été étendus à un contrôle de formation de SMV. Dans cette thèse, nous nous concentrons sur la conception de contrôleurs de suivi TVF distribués pour un MVS interconnectés en réseau. La principale contribution a été d'assurer un échange d'informations de réseau distribué et réduit entre les individus de SMV. De plus, des hypothèses pratiques ont été assouplies telles que la connectivité du réseau. La stabilité du SMV a été étudiée en s'appuyant sur la théorie de Lyapunov. Des simulations numériques ont été fournies pour confirmer les conjectures des développements proposés avec une application sur systèmes multi-Quadrotors.

**Mots-clés:** Commande coopérative, Théorie des graphes, Théorie de Lyapunov, Systèmes multi-agents, Suivi de formation, Techniques de consensus, Quadrotors, Robots mobiles.

### Abstract

A cooperative Multi-Vehicles Systems/Multi-Agent Systems MVS/MASs can be characterized as a group of decision-making autonomous agents/vehicles operating together and seeking a common and collective objective (i.e., group behaviors) based on their sensed information and the shared inter-agents information. This thesis investigates the methods and techniques from a control point of view to maximize the overall gained performance of a robots-system when accomplishing cooperatively a given task in terms of stability, robustness, speed and tracking accuracy. The main objective is to design cooperative Time Varying Formation Tracking controllers for a group of networked vehicles, while tracking a formation reference trajectory. In the literature, classical MVS control approaches have been deeply investigated such as, behavioral based method, virtual structure approach and leader-Follower technique. However, these methods have been shown to be unified within the framework of Consensus techniques. Consensus technique (or protocol) refers to a control law designed for a group of agents (featured by a linear dynamics) to reach an agreement in some variable of interest (i.e., States). The consensus problems have been extended to a MVS formation control. In this thesis, we focus into designing distributed TVF tracking controllers for a networked MVS. The main contribution was into ensuring a distributed and a reduced network information exchange among the MVS individuals. In addition, practical assumptions have been relaxed such as the connectivity of the network. The MVS stability has been studied relying on Lyapunov theory. Numerical simulations have been provided to confirm the conjectures of the proposed developments with an application to multi-Quadrotors systems.

**Key-words:** Cooperative control, Graph theory, Lyapunov theory, Multi-Agents Systems, Formation tracking, Consensus techniques, Quadrotors, Mobile robots.

# DEDICATION

**To my parents: Dalila and Brahim**

**To my spouse: Asma El-ibda3**

**To my daughters: Rassil, Hibet-Errahmane, and Ikhlass**

**To my brother and sisters**

**Algiers, December 2021**

# ACKNOWLEDGEMENT

This thesis has been conducted at both, *(i)* the Laboratory of Process Control Laboratory in the Department of Automatic of the National Polytechnic School (ENP), Algiers, *(ii)* the Military Polytechnic School (EMP) of Algiers, and *(iii)* the Ecole Supérieure des Techniques de l'Aeronautique (ESTA) of Dar el-beida.

First of all, I would like to take this opportunity to express my deepest gratitude and thanks to my project supervisors, Professor **Mohamed Tadjine** and Dr **abdelkrim Nemra** for their constant guidance, assistance and support as well as all the knowledge they shared during the course of this research.
I also wish to thank President of jury Mr **Boucherit Mohamed Seghir**, Professor at ENP, and the members of the jury: Mr **Moussa Haddad**, Professor at EMP, Mr **Belhocine Mahmoud**, Director of Research at the CDTA and Mr **Chakir Messaoud**, MCA at ENP for accepting to be members of the reading committee and for their constructive analysis of the present work.

I would like to thank The High Commandment of **CFA** and **ANP** for giving me the opportunity to follow my doctoral studies.

Further, I sincerely thank Professor **Rochdi Merzouki** for kindly inviting me to a scientific stay as a PhD student researcher at the CRISTAL center, Ecole Polytechnique of Lille, in France. The friendly and focused atmosphere at the Cristal has been a very memorable experience. In addition, my thanks go to my friend **Wei Jiang** from the School of Electrical Engineering, Aalto University, Espoo, Finland for his valuable and technical support during my stay at CRISTAL Center.

Last but not least, I would like to express my appreciation and gratitude to my family members in particular my parents and my spouse, who have encouraged, motivated and supported me during my studies.

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# List of Abbreviations:

| | | | |
|---|---|---|---|
| AI | Artificial Intelligence | MASs | Multi-Agents Systems |
| EDFs | Elliptic Fourier Descriptors | MRSs | Multi-Robots Systems |
| GLS | General Linear System | MVSs | Multi-Vehicles Systems |
| LMI | Linear Matrix Inequality | UAV | Unmanned Aerial Vehicle |
| LTI | Linear Time-Invariant | UGV | Unmanned Ground Vehicle |
| IPFs | Implicit Polynomial Function | TVF | Time-Varying Formation |

# List of Notations:

| | |
|---|---|
| $\mathbb{R}^n$ | Set of the $n-$dimensional Euclidean vector space |
| $\mathbb{R}^{n \times m}$ | Set of the $mxn$ real matrix space |
| $diag\,(a_1, \ldots, a_n)$ | The $n-$dimensional identity matrix |
| $I_n$ | The $n$-dimensional identity matrix |
| $\|x\|$ | The 2-norm of a vector $x$ |
| $|x|$ | The absolute value of a scaler $x$ |
| $\lambda_{min}(\mathrm{P})$ | Smallest eigenvalue of the real symmetric matrix $P$ |
| $\lambda_{max}(\mathrm{P})$ | Largest eigenvalue of the real symmetric matrix $P$ |
| $P > (\geqslant) 0$ | The matrix $P$ is positive (nonnegative) definite |
| $\otimes$ | The Kronecker product |
| (.) | Element by element matrix multiplication |
| $\underline{0}_n$, $\underline{1}_n$ | The column vector with $n$ entries being 0 or 1 respectively |

# CHAPTER 1

# INTRODUCTION AND STATE OF THE ART ON SYSTEMS COOPERATIVE CONTROL

## Contents

## 1.1 Introduction

On the past two decades, rapid advances in miniaturizing of computing, communication, sensing, and actuation have made it feasible to deploy a large number of autonomous vehicles or agents to work cooperatively to accomplish civilian and military missions. Thus, compared to a single complex vehicle/agent, multi-vehicles system has the capability to significantly improve the operational effectiveness, reduce the costs, and provide additional degrees of redundancy. Having multiple autonomous agents to work together efficiently to achieve collective group behaviors is usually referred to as cooperative control of multi-agent **(MASs)** or multi-vehicle systems **(MVSs)**. Due to its potential applications in various areas such as satellite formation flying, distributed computing, robotics, surveillance and reconnaissance systems, electric power systems, cooperative attack of multiple missiles, and intelligent transportation systems, cooperative control of multi-agent (multi-vehicle) systems has received compelling attention from various scientific communities, especially the systems and control community.

For a cooperative control problem, the main task is to design appropriate controllers to achieve the desired group objective. Due to the large number of agents, the spatial distribution of actuators, limited sensing capability of sensors, and short wireless communication ranges, it is considered too expensive or even infeasible in practice to implement centralized controllers. Thus, distributed control, depending only on local information of the agents and their neighbors, appears to be a promising tool for handling multi-agent systems.

Designing appropriate distributed controllers is generally a challenging task, especially for multi-agent systems with complex dynamics, due to the interconnected effect of the agent dynamics, the interaction graph among agents, and the cooperative control laws.

In this thesis we address typical study of the cooperative control problems of MASs/MVS including,

(i) UAV-UGVs coordination and robustness with nonlinear vehicle dynamics subjected to unknown external disturbance under centralised control architecture.

(ii) Output based feedback Time-Varying Formation control and Distributed Tracking of MASs/MVSs with general linear agent dynamics based on the consensus tracking framework. The focus on this part is into relaxing some requirements on the communication topology among the agents to achieve the overall system stability and render the proposed formation control more suitable and more applicable in real conditions. Furthermore, the proposed formation control is designed in sort to reduce the network information exchange-rate among the MASs, which result in a light interaction burden and decrease the used computing resources. In addition, the effect of the interaction or the communication topology among the networked agents on the convergence rate is further studied.

The study is conducted for large-scale MASs/MVSs, from centralized control architecture to a fully distributed one, from formation stabilisation to formation tracking under Leader-Followers control structure, from undirected communication topology to directed one, and from a leader whose control input is zero to a leader whose control input is nonzero and unknown.

## 1.2 Motivations

**A] What is a MAS/MRS:**

An agent is a computable entity with the ability of communication, collaboration and adaptability, which can take flexible and independent actions to achieve the tasks by interaction with the environment. Roughly talking, it denotes a dynamical system which can be a ground or underwater vehicle, an aircraft, a satellite, a smart sensor with microprocessors, and so on.

In some literature, multi-agent systems are also called multi-vehicle systems (MVS) or multi-robot systems (MRS), with efforts to avoid causing confusion with the multi-agent systems in *computer science*. The multi-agent systems concerned in this thesis and in **the robotics, systems and control** community are quite different from those in the *computer science* community, regarding the meanings, the objectives, and the commonly used tools, even though they share the same name. Therefore, hereafter we denote by agent a physical vehicle/robot that is featured by certain dynamics. Thus, each agent is equipped with a micro-controller or processor, sensors, actuators, and with the capability to communicate with other agents (e.g., transceivers). Therefore, the theories proposed for MASs can be applied to the MVSs/MRSs of which the types are ground, aerial or underwater vehicles. Those vehicles are usually featured by nonlinear dynamics and may need to be linearized to adopt the proposed theories in relation with the MAS theory such as the consensus techniques.

Currently, multi-agent system (MAS) has become one of core concepts in Artificial Intelligent (AI) area, and has been applied to research of Distributed AI. A MAS can be seen as a group of complex distributed intelligent vehicles. The characters of MAS contain: (1) Each agent has limited information resources and problem solving ability; (2) Global control is inexistent in MAS system; (3) Knowledge and data may be continuous of discrete; (4) Computing is executed in asynchronous mode. Actually, a multi-vehicles system approximately can be treated as MAS, in which each vehicle can be treated as a intelligent and autonomous agent with the ability to deal with local tasks and the ability to coordinate with neighbouring agents.

**B] Why cooperative control of MAS/MVS:**

Over the past two decades, the researches on Multi-Agent (Multi-Robot) Systems (MASs, MRSs) control and coordination have attracted considerable attention from different fields. In particular with the rapid advances of communication, sensing and embedded techniques. This is due to the broad applications a MASs offer such as, surveillance, transportation, cooperative construction, search and rescue and forest fire-monitoring [3,4]. Among the aspects related to the MASs, the cooperative control problems have attracted the attention of scientists. This is because of its essential role for the coordination of systems consisting of multiple robots/agents.

The distributed cooperative control of a MVS/MAS can be characterized as a group of decision-making autonomous agents/vehicles operating together and seeking a common and collective objective (i.e., group behaviours) based on their sensed information and the shared inter-agents information. Here **cooperative** refers to a close relationship among all vehicles in the group where **information sharing** plays

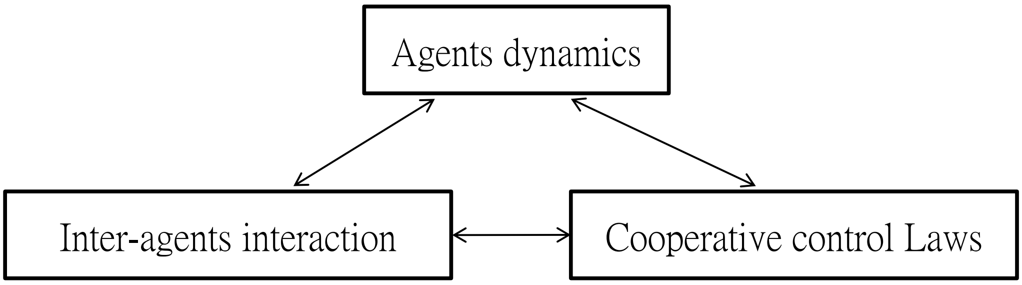


Figure 1.1: Configuration of a MASs.

a central role. If there is no inter-agents communication or information sharing (exchanging) through form example a wireless network, we denote the cooperative control in this case as decentralized. Given a group objective, the cooperative control problem of MASs is mainly composed of three components, namely, the agent dynamics, the interactions among the agents, and the cooperative control laws required to achieve the group objective. The configuration of these three components is depicted in Fig. 1.1. The selection of the cooperative control laws depends on the agent dynamics and the interaction topology. The interplay of these three components generally renders the design of the control cooperative laws troublesome, especially for the case with complex agent dynamics. For different scenarios, the dynamics of the employed agents may also be different (i.e., first integrator, second integrator, general linear dynamics). For MASs concerned by the systems and control community are generally dynamically decoupled from each other, which implies the necessity of cooperation in terms of information exchange between the agents to achieve collective behavior. Specifically, each agent needs to receive information from other agents via a direct sensing or communication network. The interaction (information exchange) topology among the agents is usually represented as a graph.

Broadly speaking, a distributed MVSs/MASs coordination/control algorithm has to satisfy four different constraints for it to be useful, namely it must be (i) **local** in the sense that individual vehicles can only act on information it has available to it, i.e., through sensing or active communications, this is sometimes referred to as "distributed"; (ii) **scalable** the algorithms executed by the individual robots cannot depend on the size of the entire team for instance, sometimes referred to as "decentralized";

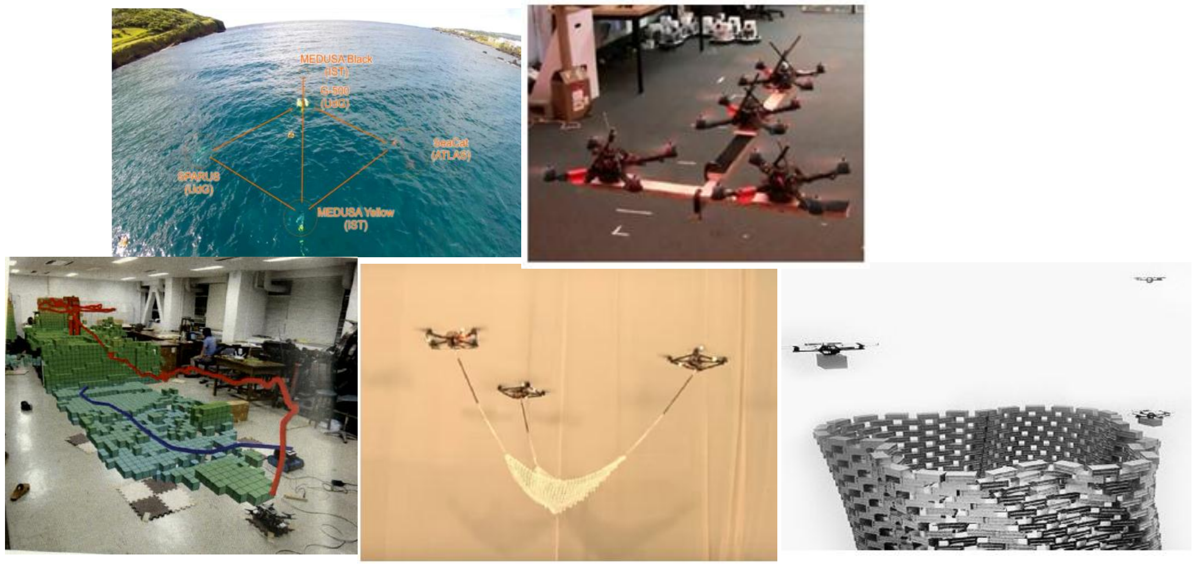


Figure 1.2: Laboratory and real-world cooperative control examples.

(iii) **safe**, as vehicles are physical agents deployed in the real world, they must be safe both relative to collisions with each other and relative to the environment; and (iv) **emergent** in the sense that global properties (e.g., **give example**) should emerge from the local interaction rules, preferably in a provable manner and not predefined[4].

The motivation for cooperative control of MASs can be summarized as follows:
**1.** The deployment of cooperative group of robots to accomplish a mission yields greater benefits comparing to a single robot performing solo mission in terms of, power, reliability, efficiency, and accuracy.
**2.** It is much cheaper to build some robots/vehicles that have limited function than a single powerful robot/vehicle.
**3.** Multiple robots/vehicles can solve problems faster than only one and increase robustness through redundancy.
**4.** Some missions can not be accomplished with a single robot.

**C] Why fully distributed control:**

The overall control architecture of MASs/MRSs has a significant influence on the robustness and scalability of the whole system. The most common control architectures or structure of MASs are[5,6],

**1. Centralized control architecture**: this architecture control is based on the assumption that a unique central station (i.e., control unit) is available and sufficiently powerful to manage the whole information about environment and to calculate the control inputs for the whole group of vehicles (i.e., decomposes and assigns tasks through plan algorithm and optimize algorithm, organizes vehicles to complete tasks by sending commands). This architecture has the advantage that all information is collected by a single unit. However, obviously this model has disadvantages in flexibility, integrality, expandability and fault tolerance.

In the other hand, in decentralized control architecture, the action of each robot is based only on local sensing information. Meaning, each agent has its own control unit, and there are no central units. The method exhibits robustness, scalability and parallel processing yielding high computation gain. however, the method suffers of the inability to achieve a group optimal performance at all times, since each robot/agent cannot predict the group behaviour, because it has only limited and incomplete information of the other individuals.

**2. Hierarchical**: This architecture is directly inspired by the military command protocol, where it is recommended for certain applications. It is based on the idea that some robots can command, like leaders (i.e., supervisors), that is, a small group of robots. Once more, as in centralized architectures, the problem with this approach lies in the case of the failure of supervisors.

**3. Hybrid structure**: This architecture is a compromise between the centralized architecture and the decentralized architecture. In particular, it is based on the idea that one or more high-level supervisors affect the tasks, lower-level resources and robots in the hierarchy using the local information to accomplish predefined tasks.

**4. Distributed architecture**: In the contrary to centralized architecture, the distributed architecture does not require a central station for control. In addition, no global information is required for the implementation of the local agents-controller.

The distributed control architecture has many advantages in achieving cooperative group performances, especially with low operational costs, less system requirements, high robustness, strong adaptability, and flexible scalability, therefore has been widely

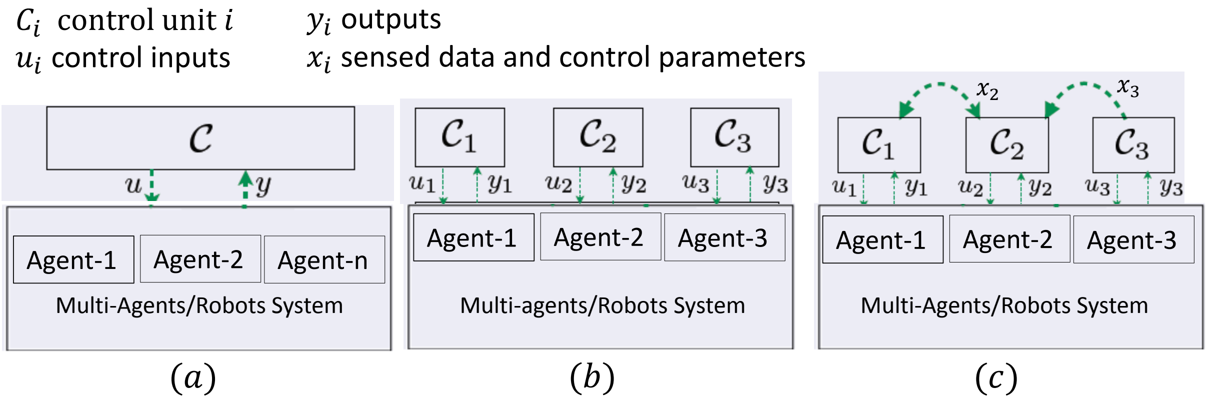


Figure 1.3: Comparison between the different MAS/MRS control architectures, (a) centralized, (b) decentralized and (c) distributed.

appreciated. However, theses advantages come at the cost of becoming far *more complex in structuring, organization the MAS/MRS and in designing local controllers.*

It is further worth to note that the decentralized architecture is a subset of the distributed architecture. The main difference is that in the latter, the individuals can exchange locally states and control parameters with their neighbours, whereas, in the decentralized method the vehicles are not interacting locally through communication.

**Remark 1.1** *It is worth mentioning that both the centralized and distributed control architectures are the most adopted and are considered to be practical depending on the situations and conditions of the real applications. For example, the centralized control structure may come more adequate for controlling a few number of slow dynamics UGVs operating in free-obstacles environment. However, the distributed method is believed more promising due to many inevitable physical constraints such as limited resources and energy, short wireless communication ranges, narrow bandwidths, and large sizes/number of vehicles to manage and to control. Thus, distributed control structure, depending only on local information of the agents and their neighbours, appears to be a promising tool for handling MRSs/MRSs. Therefore, in this thesis, the centralized method is adopted in* ***Chapter-2*** *and* **3**, *while the focus is relatively more on the distributed tracking feature in rest of the thesis.*

In the literature, many distributed controllers designed in the existing works such as [7–9], cannot apply to large-scale systems. In other words, the designed controller in the aforementioned works are not fully distributed. This is because the controllers design requires the knowledge of some global information of the system like, (i) the knowledge of the Laplacian matrix $\mathcal{L}$ of the communication topology, e.g. the minimum eigenvalue of $\mathcal{L}$, or (ii) the total number of the robots/agents. Therefore, designing the control protocol with the fully distributed property is important, vital for practical applications and challenging.

### D] Why Time Varying Formation TVF tracking

Among the branches/problems related to the cooperative control of MASs/MRSs such as consensus control, flocking, formation control, containment control, cooperative synchronization; the **formation control** is considered as one of the most important issues and an interesting and very active research topic. This is due to its main role for the success of any task executed by a MRSs that requires synchronized motion control [3]. In addition, formation control is applied to numerous areas and wide range of applications such as, target enclosing, sensor networks, cooperative surveillance, load transportation and localization [10–13].

A TVF tracking refers to the ability of MRS to change its formation shapes (i.e. geometric relations between robots) in certain circumstances, while tracking a reference trajectory and keep being stable simultaneously. The formation shape changing can be required for many reasons such as,

**1.** Covering large parts of an area, wherein specific applications relating to environment-mapping, the ability of a MRS to spread out and to gather is essential.

**2.** Avoiding obstacles during the formation motion is critical for the MRS, where it is a practical way for the MRS to change the shape of the formation when facing obstacles to avoid collision.

**3.** Tracking and enclosing a target, where a group of robots needs to adjust the formation shape in such a way to track and surround a target for protection purposes, for instance.

### E] Why heterogeneous MASs

In the literature, the first works related to MASs/MRSs dealt with the large scale of homogeneous agents/vehicles, called swarms which obtain inspiration from biological societies (particularly ants, bees, fishes and birds) to develop similar behaviors to accomplish impressive group tasks (see Fig. 1.4). In such swarm systems, individual agents/robots are usually unaware of the actions of other robots, other than information on proximity. In contrast, heterogeneous agents/robots in which team members may vary significantly in their, (i) types or the operating environment (ground, aerial, underwater robots), (ii) behaviour and dynamics, (iii) size and cognition, (iv) capabilities, gain an increasing researchers attention more and more since the last decade. The motivation to investigate heterogeneity can be three folds[5],

**1.** From designing aspect, with heterogeneity, different robots can have different capabilities (boardability, mobility, energy autonomy, field of view) and a complementarity to finish a cooperative task with less cost (e.g., localization, reconnaissance and surveillance task by aerial-ground robots).

**2.** From engineering aspect, sometimes it is too difficult do equip the same robots with all the necessary calculating, sensing and executing equipments to finish a specific cooperative task. Therefore, robots in team can have different functionality resulting in an increase of the achievable performance(See Fig. 1.5).

**3.** It is nearly impossible to build a truly homogeneous systems in reality.

## 1.3 Overview on MAS/MRS Formation Control

### 1.3.1 Formation Control approaches

The motion control of MRS has received considerable attention from researchers in the robotic community. In literature, the MRS/MAS motion control is mainly based on three approaches. The most common one is, (i) **leader-follower**, in which some robots are assigned as leaders while the rest of the formation robots are considered followers. Each follower is controlled in order to maintain with its leader a given configuration, while the leader tracks a predefined reference trajectory. This approach presents the advantages of simplicity and efficiency. However, the main drawbacks are in error propagation, non self-organization formation, and when one of the leaders fails to track its trajectory, its follower robots fail too [14]. To overcome these limita-

(a) Ants move a leaf together.

(b) Swarm bees.

(c) Swarm fishes.

(d) Swarm birds.

Figure 1.4: Examples of swarm from nature.

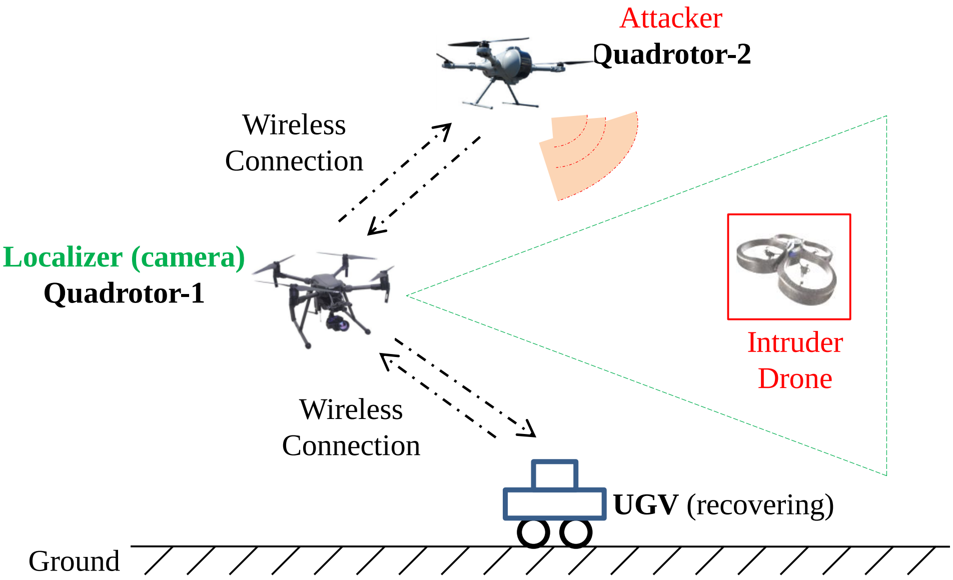


Figure 1.5: An example of an heterogeneous system consisting of two quadrotors with different capabilities (one equipped with optic-detection system and the second with launch-fillet system), where the UGV is charged for recovering the intruder target. The whole system is charged for surveillance, detection and neutralizing intruded targets.

tions, other alternative solutions are proposed by implementing leader reassignment technique or by adopting a strategy based on a virtual leader [15]. Another control approach is, (ii) **the virtual structure model**, where the leader is virtual and the system is considered as a virtual rigid body with a fixed geometric model describing the spacial relationship among robots [16]. Therefore, the leader never fails and the stability of the whole system is not depending on the leader. The third motion control approach is, (iii) **behavioural-based method**, in which we assign to each robot/agent some desired local interactional sub-behaviours (i.e., robot-robot spacing, obstacle avoidance, goal-achieving), and the combination of these sub-behaviours results in a robot final behaviour [17]. The method exhibits the advantages of decentralization aspect, scalability, robustness, and easy implementation. However, the method requires high computations gain. The aforementioned MRS/MAS control approaches can be unified within the general framework of the consensus-based methods or concept[18]. The consensus techniques/protocols are based on the idea where all the vehicles update individually their information state relying on their local sensing and the exchanged neighbour's information states. As a result, the entire vehicles final information state converges to a common value. Therefore, an agreement is reached by all the agents/robots on certain variables of interest (e.g., states or a function of states). The states could represent vehicle headings positions or outputs, estimates of sensor readings in a sensor network, oscillation frequencies in a oscillators-group, and so on.

### 1.3.2 Sate of art on MASs/MRSs Formation Control

Centralized and Distributed cooperative control has been researched for decades, particularly in control community. The motivation has been stated clearly in Section-1.2. Many research branches relating to cooperative control field have appeared such as, formation control [19], cooperative synchronization [20], consensus control [21], containment control [22] and UAV-UGVs coordination [23]. In this thesis, the main focus is attributed to the consensus control, formation control, and in particular to the TVF tracking of a MVSs.

Many frameworks have been reported in TVF tracking of UGVs group assisted by a UAV. In [24], the authors have considered a cooperative maneuver among a UAV-UGVs system, where the UGVs are guided by the UAV for obstacles avoidance purposes. In [25], a coherent TVF control of heterogeneous multi-agent system was considered. Lyapunov theory and synchronization method were used to design a decentralized controller to stabilize the swarming of UAV-UVGs system. In [26], a vision-based control method was presented for the guidance of a set of UGVs to reach a desired formation. The UVGs control relies on multiple cameras-equipped UAV as a control unit. In [27], a self-assembling of UGVs formation assisted by UAV was proposed, where the UAV uses the environment views to control and supervise the morphology formation of UGVs. In [28], a leader-follower based approach was presented for heterogeneous UAV-UGV system control. The controller is based on kinematic models and relies on a centralized structure. Similar to [28], the authors in [29] proposed a virtual structure based approach to control a line formation of UGVs guided by a quadrotor. The method is based on kinematic models too. In [23], an improved expert PID target tracking control algorithm was proposed for the UAVs-UGVs system yielding an improvement of the system stability.

Almost all the frameworks mentioned above did not consider the dynamical models of the heterogeneous MRS. Furthermore, the handled formation shapes are specific forms such as (circle, line, rectangle,...,etc.). However, in practical applications, tracking free formation shapes by MRS is very practical, without losing the system stability. In [30] an operator named Elliptic Fourier Descriptors (EFDs) has been used for shape modeling and robots path planning. Image processing and computer vision are basically the fields that use the EFDs, where free-form shapes (i.e., closed contours) could be represented in digital images. In [30], the parametric and implicit models of the desired formation shape have been used to design the robots convergence controller. However, the controller is valid only for robots featured by first integrator model. Dynamic EFDs have been introduced in [31], and a formation controller has been derived to maintain a group of holonomic robots on a dynamic 2D curve. In our previous work [2], we extended the method proposed in [31] to design a controller for

3D planar formations shapes tracking using an extended dynamic version of EFDs.

It has been shown that many existing formation control approaches such as leader-follower, behavioural and virtual structure approach can be unified within the general framework of consensus control [18]. The consensus problem has been extensively studied for different continuous or discrete dynamics node under fixed or switching topologies. Then it is extended to formation control based consensus with first order dynamics node[32], second order dynamics [33, 34] and general linear dynamics [20, 35, 36]. The difference between formation control and consensus control is that in consensus control, an agreement among the agents in term of some variable of interest (i.e., states) has to be reached. While in formation control, a desired configuration is required to be achieved by the agents, thus constraints on the desired formation are to consider.

In general context, depending on whether the formation has a reference trajectory or not, formation control can be categorized into two sub-branches. Firstly, the formation stabilization (leaderless), that refers to design protocols for a MASs to only achieve a desired geometric shape. Secondly, the distributed formation tracking or leader-follower formation tracking. In this latter, the followers agents seek achieving TVF configuration while tracking the trajectory of the real/virtual leader [6]. The algorithm design in formation tracking is more challenging and difficult. However, it provides high level applications such as target enclosing [10, 37]. The formation control consists on designing algorithms for a group of networked agents with on-board sensing. Moreover, interaction capacities are considered to reach and maintain a desired (fixed or time-varying) configuration autonomously, while keep being stable [3,6].

Many interesting frameworks within the context of Time-Varying Formation (TVF) control of MASs featured by general linear dynamics and based consensus control framework have been achieved. In [35,38,39], TVF control of networked MASs relying on relative state exchanging is proposed. In this case, the authors assume the interaction topology to be undirected and some global information such as the smallest eigenvalue of the Laplacian matrix are required. In [8], successful implementation of

TVF stabilization for high order general linear MASs under directed topology is presented. However, in this latter, the communication network is assumed to be strongly connected. Furthermore, the proposed protocol is not distributed and is based on relative-state exchange. The protocol is said to be fully distributed when it is not dependant to any kind of global information such as the global number of agents or the structure of the interaction topology. This property is necessary and vital for practical applications.

In [36,40], an interesting distributed TVF stabilisation is presented, however, it is valid only for undirected graphs. Note that undirected graphs implies bidirectional information exchange, therefore, more communicating resources are needed, which is less suitable assumption in real applications. In [41,42], the authors have extended the proposed protocol in [36,40] to directed graphs. However, the control protocol is still based on relative full-state exchange and deal only with TVF stabilization. In some applications, only partial system states are measurable, thus, implementing distributed relative-output measurement based TVF control is more interesting and practical. In [43], distributed adaptive TVF stabilization based on relative output-feedback is proposed. However, it is valid only for graphs being undirected. The latter protocol has been extended to TVF stabilization (leaderless) under directed interaction topology in [44,45]. However, some constraints are assumed such as full rank of observation matrix B, and some extra conditions on the desired TVF are to be satisfied. Moreover, the interaction topology is assumed to be strongly directed rather than to have a spanning tree, which is a mild condition that implies less network communication links.

All the above works have dealt only with TVF stabilization (leaderless). In [46–48], distributed TVF tracking implementations for high order MASs have been made. However, some assumptions have been imposed. Firstly, the leader's input is assumed to be zero and only undirected interaction topology is considered. This is considered as restrictive, in the sense limited classes of formation trajectories can be generated with leader of zero control input. Secondly, the leader's input is assumed known to all the followers with bidirectional information exchange. Thus, implies an increase of communicating resources and may make the control less robust.

## 1.4 The structure of the thesis

As stated previously, the focus of this thesis is the development and analysis of cooperative and adaptive TVF tracking algorithms for a formation of agents/vehicles (ex., UGVs and UAVs). Besides the present introduction chapter, the thesis is developed in **three** additional chapters.

**Chapter 2** is devoted to construct an analytical frame of the branch UAV-UGVs system control and coordination. As such, the aim is to develop and design a tracking control for the UGVs (i.e., two-wheeled mobile robot) group assisted by a UAV (i.e., quadrotor) which acts as an eye in the sky or a leader. Consequently, it is natural to opt for the leader-followers control approach. The cooperative control of UAV-UGVs system in this chapter is investigated under centralized control structure (the UAV plays the role of central component). The main aims are to design cooperative tracking controller to improve the coordination and the flexibility on the choice of the desired UGVs-formation shapes, to provide a closed loop dynamics stability analysis and enhancing the robustness property of the cooperative control face external disturbances and unmodelled dynamics. In this chapter the UAV-UGVs system control is studied from a control point of view, where the complete dynamical models of the vehicles of (i.e., UAV, UGVs) are considered under the assumption of permanent communication-links among the UGVs and the UAV. As a result, the distributed property isn't under scope in this chapter. The desired formation shape is modelled by using the EFDs tool and the implicit polynomial functions.

**Chapter 3** is devoted to present a formation cooperative control for a group of homogeneous UAVs (i.e., quadrotor) to achieve a deployment behaviour. We expand the 2D-EFDs formation based control method presented in [31,49] to design deployment control of a UAVs (i.e., quadorors) group based 3D-EFDs model. The proposed deployment control is based on virtual structure control approach consists of converging the quadrotors to a planar 3D contour defined by the EFDs and the correspondent implicit function. The particularity of the proposed formation lies in the ability of tracking 3D planar free-form formation shapes relying only on the knowledge of their EFDs parameters.

**Chapter 4** presents a unified framework of distributed output TVF tracking control design for homogeneous Linear Time Invariant (LTI) MASs based on an *observer viewpoint* and relying on *consensus region approach* to design distributed cooperative control laws. The formation vehicles are viewed as agents equipped with on-board-sensors, local communicating capabilities and featured by linear dynamics. The interaction (i.e., communication) among the agents (i.e., vehicles) is modelled using graph theory tools (eg., Laplacian and Adjacency matrices,...,etc.). The opted control structure of the MASs/MVSs is Leader-Follower. The analysis is presented from undirected interaction topology to a directed one, and from a leader with zero control input to a one with unknown and bounded control input.

The analysis in chapter-4 is conducted in order to derive a description as global as possible of the closed loop dynamics stability induced by the proposed protocols by using the Lyapunov theory. As a consequence, a quantitative and qualitative description of the advantages (contributions brought) given by these algorithms are given comparing with the previous frameworks in the literature. The proposed algorithms are designed in a fully distributed fashion (see Section-4.4). Thus, each agent (vehicle) is able to implement local controller without using any global information relating the interaction topology such as the smallest eigenvalue of the Laplacian matrix or the global number of agents. Further, the proposed algorithms are designed in order to reduce the overall network information exchange among the agents. Thus, the agents exchange locally their outputs vectors rather than the agent state vector. In addition, an analysis of the effect of the communication graph on the MASs convergence rate is provided. Finally, the designed cooperative tracking controllers (protocols) were applied to a group of UAVs (i.e., quadrotors).

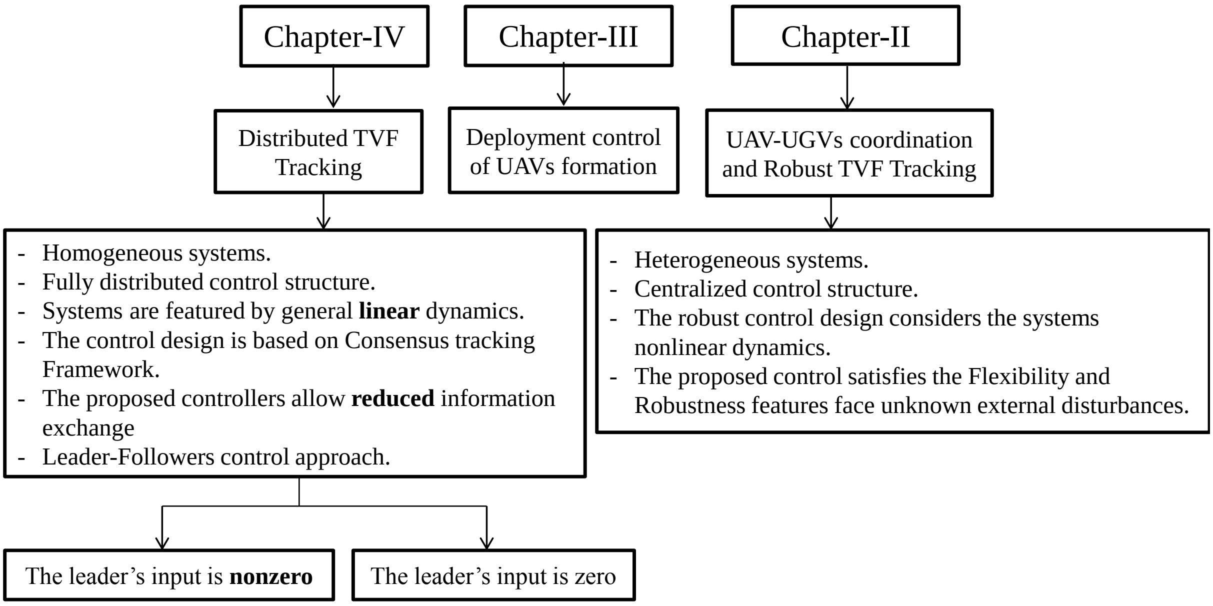

Chapter-IV
Chapter-III
Chapter-II
Distributed TVF Tracking
Deployment control of UAVs formation
UAV-UGVs coordination and Robust TVF Tracking
- Homogeneous systems.
- Fully distributed control structure.
- Systems are featured by general **linear** dynamics.
- The control design is based on Consensus tracking Framework.
- The proposed controllers allow **reduced** information exchange
- Leader-Followers control approach.
- Heterogeneous systems.
- Centralized control structure.
- The robust control design considers the systems nonlinear dynamics.
- The proposed control satisfies the Flexibility and Robustness features face unknown external disturbances.
The leader's input is **nonzero**
The leader's input is zero

Figure 1.6: Organization of the thesis.

## 1.5 Main contributions

The main contribution of this thesis can be summarized in two parts as follows,

1] In the UAV-UGVs cooperation control branch (Chapter-2), the focus in this part of the thesis was into improving the flexibility and the robustness of the UAV-UGVs cooperative control, as such,

- A novel deployment controller for the UGVs based on the kinematic model is proposed. In which, the nonholonomic constraints are considered. The key idea is in using the estimated implicit representation of the desired formation as a potential function to generate the UGVs reference trajectories.

- A novel robust cascaded velocity-torque controller based on the UGV kinematic and dynamic models is proposed to ensure the UGVs TVF-tracking. The key feature of the controller design is first, in introducing a virtual auxiliary control input to control indirectly the actual UGVs velocity vector. Next, we added a switching term to the torque input, to compensate for the unknown external disturbance and the unmodelled dynamics.

- A dynamic version of the EFDs tool to model the motion of the desired UGVs formation shape has been introduced. Thus, the dynamics of the formation is considered in the TVF tracking design.

2] In the second part of the thesis, the main challenge in formation control of a networked MASs, is that the control algorithm design must be fully distributed. Thus, the agent uses only the exchanged information from neighbours. Furthermore, it has to be independent of any global information such as the global number of agents, or the communication structure (eigenvalues of the Laplacian matrix). Moreover, some assumptions and constraints on the interaction topology among agents need to be relaxed, such as being undirected or (directed and strongly connected). This latter fact may increase the network burden and the communicating resources compared to directed topology containing a spanning tree. In other words, computation complexity is proportional with the increase of the interconnection links among agents over the network [50]. Another aspect to be considered when dealing with the formation control is the size of exchanged information among agents that is required to be minimized as possible. Hence, relative-outputs feedback based formation control is more suitable in real application than relative-states feedback based formation control. It is worth noting that dealing with distributed formation tracking under leader-follower schema is more challenging than formation stabilization (leaderless). In particular, when the leader's input is nonzero, unknown and its output measurement is available to at most a small subset of the followers.

The distributed formation control considering the above facts become very suitable, adaptive, flexible and promising for real applications in different situations. In particular, when some physical constraints are inevitable such as, short and/or directional wireless communication ranges, limited resources and energy, communicating interference, high number of systems and narrow bandwidths [6,50]. Hence, studying the fully distributed output feedback formation control under directed topology having a spanning tree (minimal communication links) ensuring: (i) reduced exchanged information among agents and (ii) considering a leader with unknown nonzero input is necessary, practical and vital. The contributions in this part are summarized as follows,

- Firstly, an adaptive and fully distributed TVF Tracking controller is proposed. In this controller, the leader's input is zero. Moreover, only one local observer is designed for each agent to observe the synthesized network formation signal. Thus latter contains only the neighbouring output measurements and the output of the distributed observer's states. Thus, agents aren't required to exchange the local observer states, which result on a less information exchange among agents. In other words, the proposed protocol enables reduced network information exchange. This is due to the fact that the size of the output system signal is generally smaller than the size of state system. Furthermore, the TVFT control design relies on the fact the leader's output is known to only a subset of followers (at least the rooted follower agent). These result in a mild requirements and make the controller more suitable for real applications and applicable for large scale systems.

- Secondly, the first proposed controller is extended to the case where the leader's input is nonzero, bounded and unknown to all the agents. Partly inspired by [51,52], a discontinuous protocol is proposed. In which, the idea is to deal with the leader's nonzero input as an external disturbance that has to be compensated. It is worth noting that no constraints are assumed on the agents dynamics such as assuming the observation matrix $B$ to have full rank and some extra-conditions on the desired TVF as in [53]. Thirdly, the tracking-error convergence rate analysis of the MASs towards the desired formation is provided, and then the proposed protocol has been applied to group of Quadrotors, by using a feedback linearization technique to achieve distributed TVF tracking scenario as an example of target enclosing and trajectory tracking application.

## 1.6 Communications and publications

**List of publications and communications (national and international)**

**International Publications:**

[1] Ahmed Allam, Abdelkrim Nemra and Mohamed Tadjine, "Parametric and Implicit Features based UAV-UGVs Time-Varying Formation Tracking: Dynamic Approach”, Unmanned Systems Journal, 10(1)(2021), https://doi.org/10.1142/S2301385022500066.

[2] Ahmed Allam, Abdelkrim Nemra and Mohamed Tadjine, "Distributed Time-Varying Formation Tracking of MASs with reduced network information exchange and nonzero leader’s input", International Journal of Modeling and Simulation (under review, 2021).

**National and International Communications:**

[1] Ahmed Allam, Mohamed Tadjine, Abelkrim Nemra and Elhaouari Kobzili, "Stereo vision as a sensor for slam based smooth variable structure filter with an adaptive boundary layer width", in: 2017 6th International Conference on Systems and Control (ICSC), IEEE, 2017, pp. 14–20 (2017).

[2] Ahmed Allam, Mohmed Tadjine, Elhaouari Kobzili, Abdelghani Boucheloukh and Abelkrim Nemra, "3D Robots Formation cooperative Control Using EFDs", in: 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS), IEEE, 2018, pp. 1–6 (2018).

[3] Ahmed Allam, Mohamed Tadjine, Abdelkrim Nemra and KOBZILI Elhaouari, Cooperative control of wireless interconnected multi-quadrotor system with time-varying formation shape, in: National Conference on Electrical Engineering CEE’19, Bordj El-bahri, EMP (2019).

[4] Elhaouari Kobzili, Cherif Larbes and Ahmed Allam, "Multi-rate robust scale estimation of monocular slam", in: 2017 6th International Conference on Systems and Control (ICSC), IEEE, 2017, pp. 1–5 (2017).

[5] Fethi Demim, Abdelkrim Nemra, Kahina Louadj, Abdelghani Boucheloukh, Elhaouari Kobzili, Ahmed Allam, Mustapha Hamerlain and Abdelouahab Bazoula, "Visual SVSF-SLAM Algorithm Based on Adaptive Boundary Layer Width", in: International Conference on Electrical Engineering and Control Applications, Springer, 2017, pp. 97–112 (2017).

[6] Elhaouari Kobzili, Cherif Larbes, Ahmed Allam, Fethi Demim and Abdelghani Boucheloukh, "Geometric binary descriptor based monocular slam", in: 2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS), IEEE, 2018, pp. 1–6 (2018).

[7] Elhaouari Kobzili, Cherif Larbes, Billel Kellalib, Fethi Demim, Ahmed Allam and Abdelghani Boucheloukh, “Multi sensor data fusion with risk assessment”, in:International Conference on Advanced Electrical Engineering (ICAEE), IEEE, Algiers, (2019).

# CHAPTER 2

# PARAMETRIC AND IMPLICIT FEATURES BASED UAV-UGVS TIME-VARYING FORMATION TRACKING: DYNAMIC APPROACH

## Contents

## 2.1 Introduction

THE collaboration in MVSs offers valuable advantages comparing to the use of a single robot when accomplishing complex tasks, and increases the capabilities and the efficiency of execution. One of the promising research branches of MVSs/MRSs is UAV-UGV control and coordination. The researches on UAVs- UGVs control and coordination have attracted increasing attention. In particular with the rapid advances of communication, sensing, and embedded techniques. This is due to the broad and various applications of UAV-UGV coordination systems in both civilian and military fields such as exploration, surveillance and inspection, rescue, and environmental monitoring. The UAV-UGV coordination has demonstrated the capabilities in providing effectiveness, robustness, reliability, and practical solutions to the real-world that cannot be brought by other types of coordination.

Among the aspects relating to MRSs control, we consider in this chapter the time-varying formation tracking of a UGVs group assisted by a UAV that acts as an eye in the sky and as leader that decides the UGVs formation shape. The opted MRS motion control is Leader-Follower. A TVF control refers to the ability of MRS to change its formation shape (i.e. geometric relations between robots) in specific conditions, while tracking a reference trajectory and preserving the whole system stability.

In this chapter, we consider two tasks to be accomplished by the UGVs-UAV system. Firstly, UGVs will carry out the deployment task, which consists of forming an initial free geometric configuration around the planar position of the quadrotor. In the second task, the UVGs will track TVF shapes while tracking a formation reference trajectory. The UAV-UGVs system is studied from a control point of view, where the complete dynamical models of the MRS are considered (i.e., heterogeneousness is considered) and the assumption of permanent communicating-links among the UGVs and the UAV is assumed in this chapter, thus the distributed property isn't under scope. The principal control aims are the flexibility in the choice of the formation shapes through the use of the EFDs tool, and the MRS stability and robustness face the external disturbance and the unmodelled-dynamics. We first present the EFDs formation modeling tool, the complete dynamical models of the mobile robot and quadrotor are further developed. Then, we illustrate the UAV-UVGs system control structure, followed by the design of the UGVs deployment control and the TVF tracking velocity-control designing. The UGVs formation robust torque-control design and the stability analysis with some meaningful remarks are further detailed step by step. Finally, the proposed formation controllers are validated through numerical simulation and experimental results.

## 2.2 UAV-UGVs system model

The control designing of a UGVs formation assisted by UAV is more challenging than the control of a homogeneous system. Because in the UAV-UGVs, each agent may have different dynamics and constraints. In this section, we highlight this challenge. Firstly it is described how to model parametrically a 2D time-varying formation shape using the EFDs and its corresponding IPF. Secondly, the dynamical model of the UAV (i.e., quadrotor) is presented, which will be considered as a flying leader of the ground mobile robots formation. The UAV decides the desired formation shape as well as the formation reference trajectory. Finally, the dynamical model of the UGV (i.e., two-wheeled mobile robot) is briefly presented.

### 2.2.1 Parametric and Implicit Representation of planar curve

**Elliptic Fourier Descriptors EFDs**

The EFDs were originally introduced [54], where the authors proposed to use the elliptical descriptors of a planar curve (i.e., contour) with Fourier descriptors representation. Therefore, any free-form 2D curve is defined by EFDs as a sum of ellipses. EFDs operate as a transformation featured by translation, rotation, and scale. Biology and anatomy are the applications fields most relating to the use of EFDs Fig. 2.1. [30,55] show how to model a closed curve using the EFDs as,

$$\begin{cases} x\left(\delta\right) = a_0 + \sum\limits_{k=1}^{n_h} \left(A_k \cos\left(k\delta\right) + B_k \sin\left(k\delta\right)\right) \\ y\left(\delta\right) = c_0 + \sum\limits_{k=1}^{n_h} \left(C_k \cos\left(k\delta\right) + D_k \sin\left(k\delta\right)\right) \end{cases} \tag{2.1}$$

In (2.1) , $k$ is an index and $n_h$ denotes the number of harmonics used to represent the curve. $a_0$ and $c_0$ are the coordinates of the curve center, and $A = [A_1, \ldots, A_{n_h}]^T \in \mathbb{R}^{n_h \times 1}$, $B \in \mathbb{R}^{n_h \times 1}$, $C \in \mathbb{R}^{n_h \times 1}$ and $D \in \mathbb{R}^{n_h \times 1}$ are vectors that represent the curve descriptors. The number of harmonics defines the representation accuracy, where the bigger is $n_h$, the more accurate representation the curve is. $x(\delta)$ and $y(\delta)$ denote the coordinates of the points forming the curve contour. These coordinates are expressed as functions of a normalized parameter $\delta$ with $\delta \in [0, 2\pi]$. For example, the EFDs parameters of a circle are given as,

$$\begin{bmatrix} a_0 & c_0 \end{bmatrix}^T = \begin{bmatrix} x_0 & y_0 \end{bmatrix}^T ; A = R_c; B = 0; C = 0; D = R_c; n_h = 1.$$

where $(x_0, y_0)$ and $R_c$ are respectively the coordinates of the center of the circle and its radius.

It is further possible to estimate the EFDs vectors of any 2D closed curve defined by a set of points, with coordinates denoted by $(x_i, y_i)\,, i \in [1, M_s]$. The EFDs vectors are estimated as in [56], [31],

$$\begin{cases} A_k = \frac{1}{M_s} \sum\limits_{i=1}^{M_s} x_i \cos\left(k\delta_i\right); B_k = \frac{1}{M_s} \sum\limits_{i=1}^{M_s} x_i \sin\left(k\delta_i\right) \\ C_k = \frac{1}{M_s} \sum\limits_{i=1}^{M_s} y_i \cos\left(k\delta_i\right); D_k = \frac{1}{M_s} \sum\limits_{i=1}^{M_s} y_i \sin\left(k\delta_i\right) \\ a_0 = \frac{1}{M_s} \sum\limits_{i=1}^{M_s} (x_i); c_0 = \frac{1}{M_s} \sum\limits_{i=1}^{M_s} (y_i)\,; k \in \{1, n_h\} \end{cases} \tag{2.2}$$

where $M_s$ is the number of points that define the curve and $\delta_i = i2\pi/\mathrm{M_s}$. It is worth noting that the precision of estimation is dependent to the choice of the number of harmonics $n_h$.

To show how EFDs can be very useful for defining and modeling any 2D curve, Fig. 2.2 is depicted as an illustrative example, where the desired curve is defined in a dotted-line on an acquired image for surveillance purposes. This curve can be modeled using (2.2), and then reconstructed using (2.1). In virtual fashion, we can represent and model any desired closed 2D curve based on this method increasing the flexibility of the description. The authors in [31] proposed an approach to describe

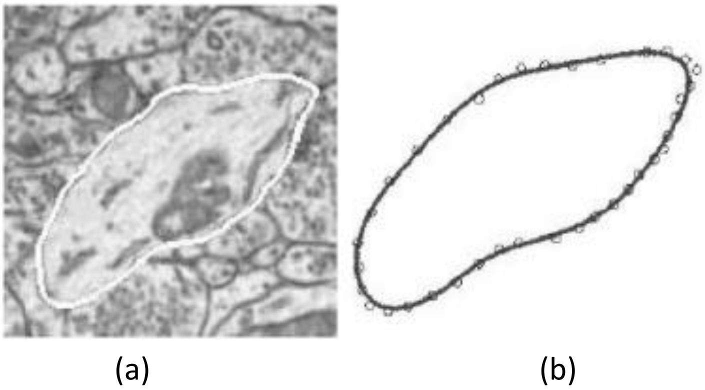


Figure 2.1: (a) Cell image, (b) reconstructed controur using EFDs (with 5 harmonics) [1].

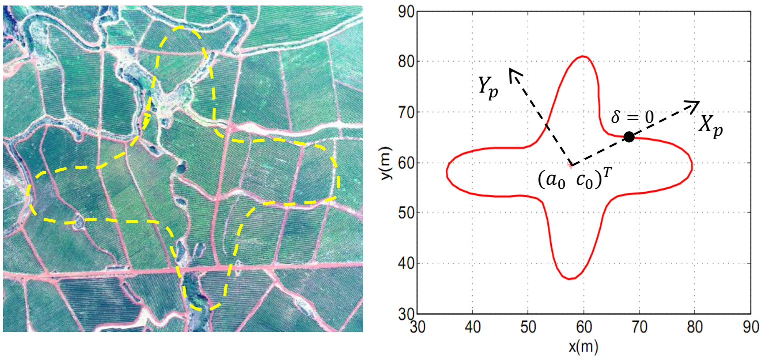


Figure 2.2: Example of a 2D closed curve reconstructed after modelling using EFDs from an acquired image.

and to model the motion of a 2D free-form curve using dynamical EFDs as,

$$\begin{aligned} x(\delta, t) &= a_0(t) + \sum_{k=1}^{n_h} \left(A_k(t)\cos(k\delta) + B_k(t)\sin(k\delta)\right) \\ y(\delta, t) &= c_0(t) + \sum_{k=1}^{n_h} \left(C_k(t)\cos(k\delta) + D_k(t)\sin(k\delta)\right) \end{aligned} \tag{2.3}$$

where $A \in \mathbb{R}^{n_h \times 1}$, $B(t)$, $C(t)$, and $D(t)$ are time-varying vectors and they are referred to as dynamical EFDs vectors and $a_0(t)$, $c_0(t)$, $x$, $y$, $\delta$, $n_h$ are already defined in (2.1). The parametric description (2.3) can be expressed as,

$$\begin{bmatrix} x(\delta, t) \\ y(\delta, t) \end{bmatrix} = \begin{bmatrix} a_0(t) \\ c_0(t) \end{bmatrix} + \underline{EFDs}(t) \tag{2.4}$$

where $\underline{EFDs} \in \mathbb{R}^2$ is a relative offset vector relating the points of the curve contour with the curve center coordinates. Dynamical EFDs will be used to model the time-varying formation shape of a group of robots.

### Implicit Polynomial Function (IPF) of a closed 2D curve

Further to the parametric representation (2.3), 2D curves could also be represented using an IPF denoted by $\mathcal{H}(x,y) = 0$. This implicit function of a closed curve is derived by the implicitization of the EFDs vectors. In this study, the IPF is obtained using the method detailed in [1]. The IPF that represents a closed curve modeled by EFDs takes the form,

$$\mathcal{H}\left(x,y\right) = \sum_{0<i+j<d_p} a_{ij}x^i y^j = 0 \tag{2.5}$$

where $a_{ij}$ are coefficients and $d_p$ is the polynomial degree with $d_p = 2n_h$ [1]. For example, the IPF of a circle is defined as $\mathcal{H}\left(x,y\right) = (x-x_0)^2 + (y-y_0)^2 - R_c^2$, where $(x_0, y_0)$ and $R_c$ are respectively its center coordinates and radius. The IPF of the formation shape can be considered as a potential fields function that could have control purposes.

### 2.2.2 Dynamical Model of a quadrotor UAV

In this section, the dynamical model of a quadrotor used as the UAV is presented. This latter is considered as a rigid body. Let $\mathcal{F}_w = (O_W, X_W, Y_W, Z_W)$ be the global inertial frame, and let $\mathcal{F}_q = (O_q, X_q, Y_q, Z_q)$ be the body-fixed frame. Let $\eta = [\varphi_q, \theta_q, \psi_q]^T$ describes the orientation of the quadrotor (Euler angles) and $\chi_q = [x_q, y_q, z_q]^T$ denotes the position of its mass center with respect to the inertial frame.

A brief explanation of the classic process is given in order to derive the simplified dynamical model [57,58]. The structure and the propellers are rigid and symmetric. The translational and rotational dynamics are expressed as,

$$\begin{aligned} M_q\ddot{\chi}_q &= -diag(K_f)\dot{\chi}_q - M_q g\underline{e}_z + u_z R_q\left(\varphi_q,\theta_q,\psi_q\right)\underline{e}_z \\ I_q\dot{\overline{w}}_B &= -\overline{w}_B \times I_q\overline{w}_B + G_a + \tau_q \end{aligned} \tag{2.6}$$

where $\underline{e}_z = (0,0,1)^T$ denotes the unit vector of $Z_W$-axis, $\mathcal{R}_q$ is the rotation matrix from the body frame to the inertial frame, $M_q$ is the quadrotor mass, $g$ is the gravitational acceleration, $u_z$ is the total thrust and $K_f = [K_{fx}, K_{fy}, K_{fz}]^T$ are the aerodynamic translation coefficients along the $x$, $y$ and $z$ axes respectively.

$\overline{w}_B = [w_{Bx}, w_{By}, w_{Bz}]^T$ denotes the angular velocity, $I_q = diag(I_{q,x}, I_{q,y}, I_{q,z})$ is the diagonal inertia matrix, $\tau_q = \left[u_\phi, u_\theta, u_\psi\right]^T$ are moments due to propellers forces acting on the quadrotor along the $x$, $y$ and $z$ body-fixed frame axes. $G_a = [J_r\dot{\theta}_q\Omega_r, J_r\dot{\phi}_q\Omega_r, 0]^T$ denotes the propellers gyroscopic effect with $\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4$ is a mixture of rotors speeds and $J_r$ is the rotor inertia [57]. The quadrotor rates $\overline{w}_B$ are transformed into Euler angular rates $\dot{\eta}$ by the following relation,

$$\dot{\eta} = \begin{bmatrix} 1 & S_{\phi_q}\tan\theta_q & C_{\phi_q}\tan\theta_q \\ 0 & C_{\phi_q} & -S_{\phi_q} \\ 0 & S_{\phi_q}/C_{\theta_q} & C_{\phi_q}/C_{\theta_q} \end{bmatrix} \overline{w}_B \tag{2.7}$$

where $S_{(.)}$ and $C_{(.)}$ are abbreviations for $sin(.)$ and $cos(.)$ respectively. The dynamical model of the quadrotor could be written using (2.6), (2.7) as follows,

$$\ddot{\chi}_q = \begin{bmatrix} \ddot{x}_q \\ \ddot{y}_q \\ \ddot{z}_q \end{bmatrix} = \begin{bmatrix} -\frac{K_{fx}}{M_q}\dot{x}_q + u_z\frac{u_x}{M_q} \\ -\frac{K_{fy}}{M_q}\dot{y}_q + u_z\frac{u_y}{M_q} \\ -\frac{K_{fz}}{M_q}\dot{z}_q - g + u_z\frac{C_{\theta_q}C_{\phi_q}}{M_q} \end{bmatrix} \tag{2.8}$$

$$\ddot{\eta} = \begin{bmatrix} \ddot{\varphi}_q \\ \ddot{\theta}_q \\ \ddot{\psi}_q \end{bmatrix} = \begin{bmatrix} \dot{\theta}_q\dot{\psi}_q\left(\frac{I_y-I_z}{I_x}\right) - \frac{J_r\dot{\theta}_q\Omega_r}{I_x} + \frac{u_\phi}{I_x} \\ \dot{\phi}_q\dot{\psi}_q\left(\frac{I_z-I_x}{I_y}\right) + \frac{J_r\dot{\phi}_q\Omega_r}{I_y} + \frac{u_\theta}{I_y} \\ \dot{\phi}_q\dot{\theta}_q\left(\frac{I_x-I_y}{I_z}\right) + \frac{u_\psi}{I_z} \end{bmatrix} \tag{2.9}$$

In (2.9), $u_x$ and $u_y$ are considered as virtual control inputs for $x_q$ and $y_q$ states, defined as [58],

$$u_x = C_{\psi_q}S_{\theta_q}C_{\phi_q} + S_{\psi_q}S_{\phi_q}; u_y = S_{\psi_q}S_{\theta_q}C_{\phi_q} - C_{\psi_q}S_{\phi_q}$$

The quadrotor control inputs $u_\phi, u_\theta, u_\psi$ and $u_z$ are defined as,

$$u_\phi = b\ell[\Omega_3^2 - \Omega_4^2]; u_\theta = b\ell[\Omega_1^2 - \Omega_2^2]$$
$$u_\psi = \sigma[\Omega_1^2 + \Omega_2^2 - \Omega_3^2 - \Omega_4^2]; u_z = \sum_{i=1}^4 b\Omega_i^2$$

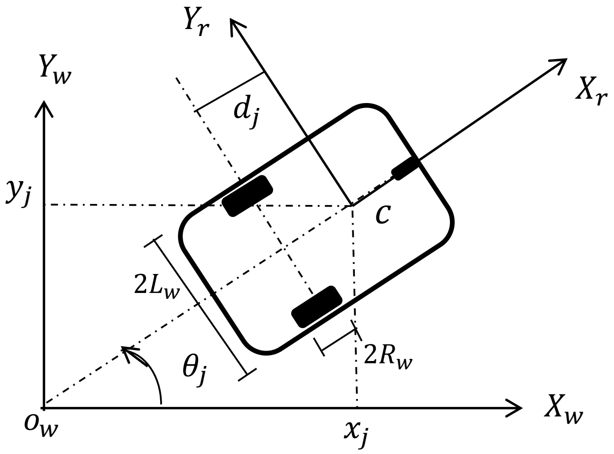


Figure 2.3: Frames and parameters of a nonholonomic mobile robot.

where $\Omega_i, i = [1,4]$ being the $i^{th}$ propeller angular velocity, $\ell$ is the distance between the motor and the quadrotor center, $b$ and $\sigma$ are the propeller aerodynamic lift and drag coefficients respectively.

### 2.2.3 Dynamical Model of a Mobile robot UGV

A group of $N_R$ homogeneous UGVs (i.e., non-holonomic two-wheeled mobile robots) is considered, where the generalized coordinates of each mobile robot are given by,

$$q_j = [x_j, y_j, \theta_j]^T \tag{2.10}$$

where $x_j$, $y_j$, $\theta_j$ are respectively, the $x$, $y$ coordinates, and the orientation of the $j^{th}$ mobile robot. In Fig. 2.3, point $c$ is the robot center of mass. The kinematic model of the $j^{th}$ nonholnomic mobile robot is written as [59,60],

$$\dot{q}_j = \begin{bmatrix} \dot{x}_j \\ \dot{y}_j \\ \dot{\theta}_j \end{bmatrix} = S^T \underline{v}_j = \begin{bmatrix} \cos(\theta_j) & -d_j\sin(\theta_j) \\ \sin(\theta_j) & d_j\cos(\theta_j) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_j \\ w_j \end{bmatrix} \tag{2.11}$$

where $d_j$ is the distance from the rear axle to the mass center of robot $j$, $L_w$ is the half distance between the left and right wheels, $R_w$ is the robot wheel radius, $\underline{v}_j = [\begin{array}{cc} v_j & w_j \end{array}]^T$ with $v_j$ and $w_j$ are respectively the linear and the angular velocities of the $j^{th}$ mobile robot.

A Nonholonomic mobile robot characterized by $n$ generalized coordinates $(q_{j1}, ..., q_{jn})$, having $r$ control inputs and subject to $m$ constraints described in detail in [60,61], and mathematically after applying the transformation described in [60, 61] to eliminate Lagrange multipliers yields the alternative dynamical model,

$$\overline{M}_j\left(q_j\right)\dot{\underline{v}}_j + \overline{V}_{mj}\left(q_j, \dot{q}_j\right)\underline{v}_j + \overline{F}_j(\dot{q}_j) + \overline{\tau}_{dj} = \overline{B}_j\underline{\tau}_j \tag{2.12}$$

In (2.12), $\overline{M} \in \mathbb{R}^{r\times r}$ is a symmetric positive definite inertia matrix, $\overline{V}_{mj} \in \mathbb{R}^{r\times r}$ is the centripetal and coriolis matrix, $\overline{F}_j \in \mathbb{R}^{r\times 1}$ is the friction vector, $\overline{\tau}_{dj} \in \mathbb{R}^{r\times 1}$ represents unknown bounded disturbances including unstructured unmodeled dynamics, and $\overline{B}_j \in \mathbb{R}^{r\times r}$ is the input matrix and $\underline{v}_j \in \mathbb{R}^{r\times 1}$, $\underline{\tau}_j \in \mathbb{R}^{r\times 1}$ are respectively the velocity and the torque vectors. For the non-holonomic mobile robot described in Fig. 2.3. The complete dynamical model is expressed as follows [60,61],

$$\begin{array}{c} \left(m + \frac{2I_w}{R_w^2}\right)\dot{v}_j - m_c d_j\left(w_j\right)^2 + \overline{F}_{j,1} + \overline{\tau}_{dj,1} = \frac{\left(\tau_{R,j}+\tau_{L,j}\right)}{R_w} \\ \left(I + \frac{2L_w^2}{R_w^2}I_w\right)\dot{w}_j + m_c d_j v_j w_j + \overline{F}_{j,2} + \overline{\tau}_{dj,2} = \frac{L_w\left(\tau_{R,j}-\tau_{L,j}\right)}{R_w} \end{array} \tag{2.13}$$

where $m = m_c + 2m_w$ is the total mass of the mobile robot, $I = (I_c + m_c d_j^2 + 2m_w L_w^2 + 2I_m)$ is the total equivalent inertia, $m_c$ is the robot mass without the driving wheels and actuators (DC motors), $m_w$ is the mass of each driving wheel (with actuator), $I_c$ is the moment of inertia of the robot about the vertical axis through the center of mass, $I_w$ and $I_m$ are the moments of inertia of each driving wheel (with actuator) around the wheel axis, and the moment of inertia of each driving wheel with a motor about the wheel diameter, respectively. The matrices $\overline{M}_j$, $\overline{V}_{mj}$ and $\overline{B}_j$ in (2.12) are obtained from (2.13) as follows,

$$\overline{M}_j = \begin{bmatrix} m + \frac{2I_w}{R_w^2} & 0 \\ 0 & I + \frac{2L_w^2}{R_w^2}I_w \end{bmatrix}; \overline{V}_{mj} = \begin{bmatrix} 0 & -m_c d_j\dot{\theta}_j \\ m_c d_j\dot{\theta}_j & 0 \end{bmatrix}; \overline{B}_j = \frac{1}{R_w}\begin{bmatrix} 1 & 1 \\ L_w & -L_w \end{bmatrix}$$

**Remark 2.1** *Common to robotic systems, the skew-symmetric property that is defined as,* $X^T(\dot{\overline{M}}_j - 2\overline{V}_{mj})X = 0$ *for all vector* $X$ *[62], is an important feature that will be used later in the stability analysis section.*

The complete mobile robot behavior can be described by (2.11) and (2.12). Let define the nonlinear feedback control input,

$$\underline{\tau}_j = \overline{B}_j^{-1} \left[ \overline{M}_j \underline{u}_j + \overline{V}_{mj} \underline{v}_j + \overline{F}_j \right] \tag{2.14}$$

where $\underline{u}_j \in \mathbb{R}^{2\times 1}$ is an auxiliary input. Applying the control law (2.14) to (2.12) allows the conversion of the dynamical control problem into the kinematic control problem such that,

$$\begin{cases} \dot{q}_j = S^T \underline{v}_j \\ \dot{\underline{v}}_j = \underline{u}_j \end{cases} \tag{2.15}$$

where the matrix $S^T$ is defined in (2.11). The feedback control input (2.14) yields a form of cascaded kinematic and dynamic linearization structure. Thus, the alternative model (2.15) is convenient for the purpose of control.

## 2.3 The UAV-UGVs Control Structure

As depicted in Fig. 2.4, the MRS control scheme consists of two consecutive tasks. Accordingly, two controllers have been proposed namely, deployment controller and TVF tracking controller.

The first task is robots deployment control, which consists of converging a group of mobile robots towards an initial formation shape (i.e., 2D curve) modeled by EFDs and its corresponding IPF. The quadrotor will be located at the center of the formation, being the leader that produces the formation shape. This scenario has a number of applications such as target enclosing and fire-monitoring.

In the second task (see Fig. 2.4), once the initial formation shape is completed, the quadrotor will track a predefined reference trajectory, while each mobile robot will generate and tack its own trajectory and simultaneously maintains a desired time-varying formation shape. The desired shape is produced by the quadrotor depending on the mission requirements, circumstances, and the changes occurring in the operating environment. For example, the quadrotor can use an onboard camera sensor

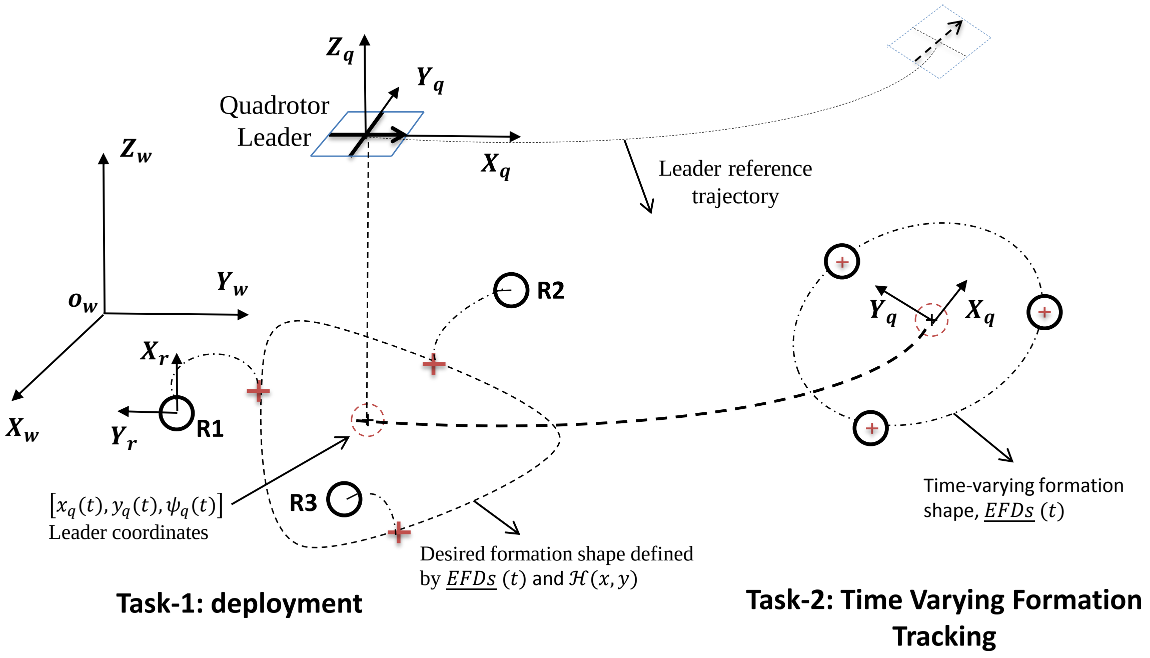

$Z_q$
$Y_q$
Quadrotor
Leader
$Z_w$
$X_q$
Leader reference
trajectory
$Y_w$
$o_w$
R2
$Y_q$
$X_q$
$X_r$
R1
$X_w$
$Y_r$
$[x_q(t), y_q(t), \psi_q(t)]$
Leader coordinates
R3
Time-varying formation
shape, $\underline{EFDs}$ $(t)$
Desired formation shape defined
by $\underline{EFDs}$ $(t)$ and $\mathcal{H}(x, y)$
Task-1: deployment
Task-2: Time Varying Formation
Tracking

Figure 2.4: Mobile Robots formation Control structure consisting of two tasks.

to acquire images from the operating environment, and then process these images to define the adequate formation shape. Both the quadrotor path planning and the process of the formation shape selection are not considered in this thesis. Thus, the EFDs vectors relating to the desired formation shape and the quadrotor reference trajectory are predefined.

**Remark 2.2** *The control of single mobile robot can be developed by considering only the kinematic model* (2.11)*,* [59]*, resulting in a pseudo velocity control input that we denote* $\underline{v}_c(t) = [\; v_c(t) \quad w_c \;\; (t)]^T$*. However, the control performance of the mobile robot can be improved by incorporating its dynamics on the controller design. Consequently, a control torque* $\underline{\tau}_j(t)$ *can be designed to make the actual velocity* $\underline{v}(t)$ *follow the designed pseudo velocity control input* $\underline{v}_c(t)$*. Thus, in this section, we seek to expand the framework developed for the control of a single mobile robot to the control of mobile robots formation based Leader-Followers approach, where perfect velocity tracking assumption is removed. Therefore, we seek to develop a robust formation torque-controller relying on velocity control input* $\underline{v}_c(t)$ *in such a way the formation desired behaviour is improved in terms of flexibility, tracking accuracy, and robustness, in contrast with the case when perfect velocity assumption is assumed.*

## 2.4 Deployment Control (Kinematic Based Approach)

The control aim in this task is to find a smooth velocity-control input $\underline{v}_{jc}(t)$, $j \in [1, N_R]$ in such a way all the mobile robots converge to the initial desired formation shape, based only on the knowledge of its IPF denoted by $\mathcal{H}(x, y)$. The idea is to use the IPF to generate the robots reference trajectory leading to the desired shape perimeter. As detailed in section-2.2, the position error function between the $j^{th}$ mobile robot and the formation shape (i.e., 2D closed curve) is expressed using the IPF as,

$$e^{j}_{Form} = \mathcal{H}(x_j, y_j) \tag{2.16}$$

where, $(x_j, y_j)$ are the $j^{th}$ robot $x$ and $y$ coordinates, $e^{j}_{Form}$ is an algebraic distance between the $j^{th}$ robot and the desired formation shape and $\mathcal{H}(x, y)$ is the IPF of the desired shape. By using (2.16), when the robot is exactly on the curve, the error is zero. While, the error is positive or negative if the robot is outside or inside the closed curve respectively. We consider the reference trajectory for the $j^{th}$ mobile robot to be defined as,

$$\dot{x}_{jr} = v_{jr}\cos(\theta_{jr}); \dot{y}_{jr} = v_{jr}\sin(\theta_{jr}); \dot{\theta}_{jr} = w_{jr} \tag{2.17}$$

where $x_{jr}$, $y_{jr}$ and $\theta_{jr}$ are the positions and orientation of a virtual reference robot, mobile robot $j$ seeks to follow. $v_{jr}(t)$ and $w_{jr}(t)$ are respectively its reference linear and angular velocities. If $x_{jr}$, $y_{jr}$ and $\theta_{jr}$ are continuously differentiable and bounded when $t \to \infty$, and $(\dot{x}_{jr} + \dot{y}^2_{jr}) \neq 0$ it can be shown that,

$$\begin{pmatrix} v_{jr} \\ w_{jr} \\ \theta_{jr} \end{pmatrix} = \begin{pmatrix} \dot{x}_{jr}\cos(\theta_{jr}) + \dot{y}_{jr}\sin(\theta_{jr}) \\ (\ddot{y}_{jr}\dot{x}_{jr} - \ddot{x}_{jr}\dot{y}_{jr}) \,/\, \left(\dot{x}^2_{jr} + \dot{y}^2_{jr}\right) \\ \tan^{-1}(\dot{y}_{jr}/\dot{x}_{jr}) \end{pmatrix} \tag{2.18}$$

The reference trajectory of the mobile robot $j$ is derived from (2.17) and (2.18), during the deployment stage is depicted in Fig. 2.5.

Refer to Fig. 2.5, in the first step, the mobile robot is approached to point-mass particle and exhibit the kinematic model given as, $[\ \dot{x}_j \quad \dot{y}_j\ ]^T = F_{a,j}$; $F_{a,j} = \begin{bmatrix} F_{xj} & F_{yj} \end{bmatrix}^T$, where $F_{xj}$ and $F_{yj}$ are the components of a virtual force. The 2D desired curve described by the IPF produces an external potential field (see Fig. 2.7), yielding an attractive force

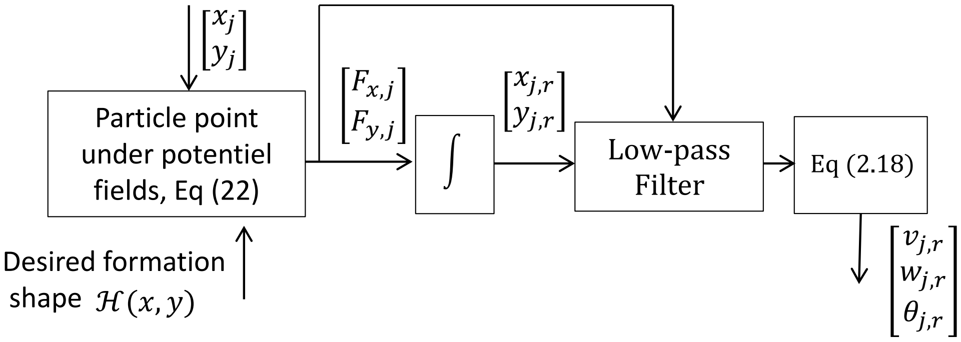

$\begin{bmatrix} x_j \\ y_j \end{bmatrix}$
Particle point under potentiel fields, Eq (22)
Desired formation shape $\mathcal{H}(x,y)$
$\begin{bmatrix} F_{x,j} \\ F_{y,j} \end{bmatrix}$
$\int$
$\begin{bmatrix} x_{j,r} \\ y_{j,r} \end{bmatrix}$
Low-pass Filter
Eq (2.18)
$\begin{bmatrix} v_{j,r} \\ w_{j,r} \\ \theta_{j,r} \end{bmatrix}$

Figure 2.5: Block diagram of reference trajectory derivation during the deployment task.

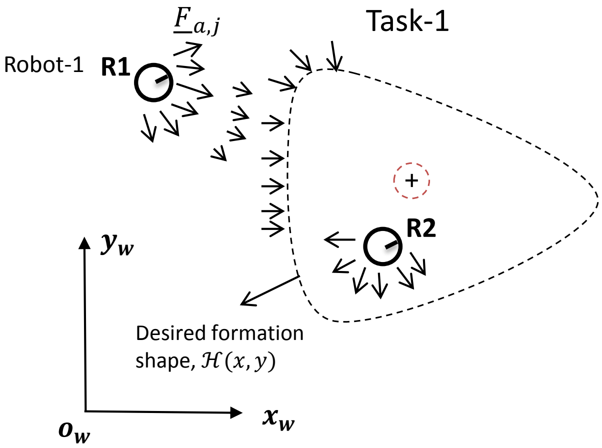


Figure 2.6: Artificial attractive force exerted by the 2D curve on robots.

$F_{a,j}$ being exerted on the robot. By integrating the virtual force, the robot reference position $(x_{jr}, y_{jr})$ is obtained and smoothed out using a low-pass filter. Then, equation (2.18) is used to calculate the corresponding reference velocities and orientation of the virtual reference robot. Now, in the following, we explain how to derive the virtual force $F_{a,j}$ exerted by the 2D curve on the $j^{th}$ robot relying on Lyapunov theory. The attractive force will be designed to force the error (2.16) to exponentially decrease,

$$\dot{e}^j_{Form} = -\lambda_1 e^j_{Form} \tag{2.19}$$

where $\lambda_1$ is a positive number. Substituting (2.16) into (2.19) yields,

$$\dot{\mathcal{H}}(x_j, y_j) = -\lambda_1 \mathcal{H}(x_j, y_j) \tag{2.20}$$

After deriving (2.16) with respect to time and substituting the particle model (i.e.,

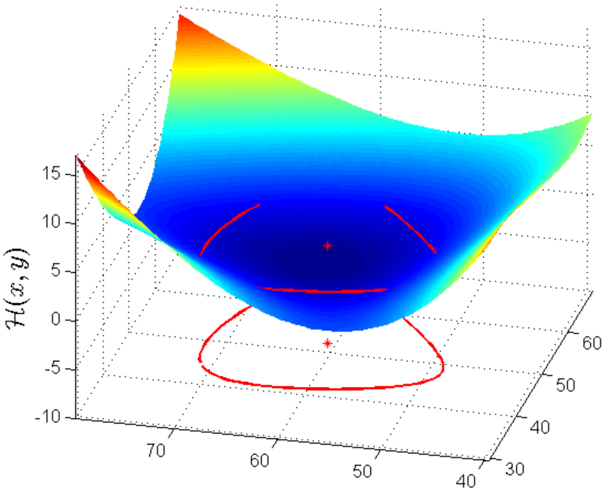


Figure 2.7: 3D plot of the IPF, $\mathcal{H}(x,y)$.

simple integrator model) $(\dot{x}_j, \dot{y}_j)^T = (F_{x,j}, F_{y,j})^T$, yields,

$$\begin{pmatrix} \mathcal{H}_x & \mathcal{H}_y \end{pmatrix} \begin{pmatrix} F_{x,j} \\ F_{y,j} \end{pmatrix} = -\lambda_1 \mathcal{H}(x_j, y_j) \tag{2.21}$$

The virtual force $F_{a,j} = [F_{x,j}, F_{y,j}]^T$ is determined from (2.21) for the mobile robot $j$ relying on pseudo inverse operator as follows,

$$\begin{pmatrix} F_{x,j} \\ F_{y,j} \end{pmatrix} = -\lambda_1 \frac{1}{\|\nabla \mathcal{H}(x_j, y_j)\|^2} \mathcal{H}(x_j, y_j) \begin{pmatrix} \mathcal{H}_x(x_j, y_j) \\ \mathcal{H}_y(x_j, y_j) \end{pmatrix} \tag{2.22}$$

where $\mathcal{H}_x$, and $\mathcal{H}_y$ are the partial derivatives of $\mathcal{H}(x,y)$ with respect to $x$ and $y$ respectively and $\nabla \mathcal{H} = \begin{bmatrix} \mathcal{H}_x & \mathcal{H}_y \end{bmatrix}^T$.

Once the reference trajectories of the mobile robots are derived, namely, $[x_{jr}, y_{jr}, \theta_{jr}, v_{jr}, w_{jr}]$, $j \in [1, N_R]$, the tracking error system $\underline{e}_j = [e_j^x, e_j^y, e_j^\theta]^T$ of the $j^{th}$ mobile robot can be expressed in the robot coordinate frame as,

$$\begin{bmatrix} e_j^x \\ e_j^y \\ e_j^\theta \end{bmatrix} = \begin{bmatrix} cos(\theta_j) & -sin(\theta_j) & 0 \\ sin(\theta_j) & cos(\theta_j) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{jr} - x_j \\ y_{jr} - y_j \\ \theta_{jr} - \theta_j \end{bmatrix} \tag{2.23}$$

Thus, the problem of controlling a mobile robots formation may be expressed as the requirement to make the error dynamics of (2.23) globally asymptotically stable.

By developing the first-order time derivative along (2.23), and using Lyapunov theory, a smooth pseudo control-input $\underline{v}_{jc}(t)$ that guarantee the asymptotic stability can be selected as [59],

$$\begin{bmatrix} v_{jc} \\ w_{jc} \end{bmatrix} = \begin{bmatrix} v_{jr}\cos\left(e_j^{\theta}\right) + K_x e_j^x \\ w_{jr} + \left(v_{jr} + K_v\right) K_y e_j^y + \left(v_{jr} + K_v\right) K_{\theta} sin\left(e_j^{\theta}\right) \end{bmatrix} \tag{2.24}$$

where $K_x$, $K_y$, $K_v$, and $K_\theta$ are properly selected positive gains. Comparing the velocity controller proposed in [59] (equation 21) with (2.24), we find that the term $K_v > 0$ is added to $v_{jr}$, thus, we ensure that the asymptotic stability holds even when $v_{jr} = 0$.

**Remark 2.3** *The followed method to generate the reference mobile robots trajectory leading to guarantee $\mathcal{H}\left(x_j, y_j\right) \to 0, j \in (1, N_R)$, as $t \to \infty$ is different from the classic Artificial Potential Fields (APFs) method, in the sense the controller is designed to converge the mobile robots to a 2D closed curve rather than to converge them to a goal point. Thus, the potential applications could be 2D obstacle avoidance, where the avoided obstacle curve can have any shape rather than being a circle as presented in many previous works on obstacle avoidance based APFs [17]. Thus, the mobile robots navigation space could be increased.*

## 2.5 UGVs Time-Varying Formation Tracking Control

During the deployment task, the initial desired formation shape is reached by the mobile robots using (2.24), where the leader (i.e., quadrotor) is located in the center of the formation $\begin{bmatrix} x_q & y_q \end{bmatrix}^T = \begin{bmatrix} a_0 & c_0 \end{bmatrix}^T$. In the second task, the aim is to design a time-varying formation tracking controller enabling the mobile robots to maintain the desired 2D curve, while tracking the formation reference trajectory (see Fig. 2.4).

First, we explain the process of generating the mobile robots reference trajectory relying on the knowledge of the desired TVF shape (i.e., EFDs vectors) and the leader's coordinates. Depending on the location of each mobile robot on the desired curve, each robot $j$ forms an angle $\delta_j$ relative to the $x$-axis of the quadrotor coordinate frame Fig. 2.8.

The *EFDs* dynamical representation (2.4) is used to define the desired position of the

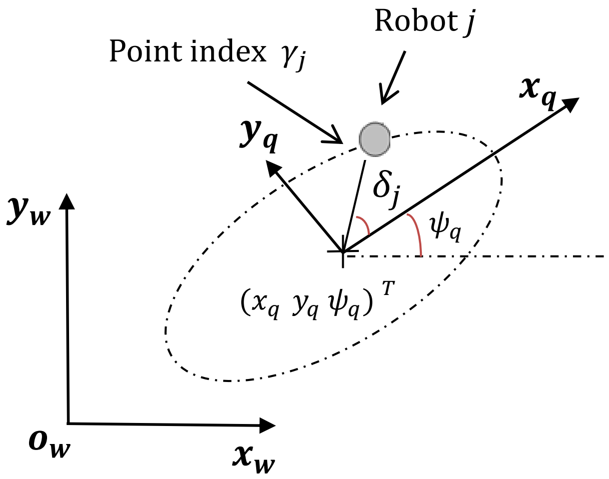


Figure 2.8: Location of the mobile robot $j$ prior to start Task-2.

mobile robots within the time-varying desired curve. The reference trajectory of the $j^{th}$ mobile robot, $q_{jr} = [x_{jr}, y_{jr}, \theta_{jr}]^T$ comes from the leader's output position $[x_q, y_q]^T$, yaw angle $\psi_q$ and the desired time-varying EFDs,

$$q_{jr}(t) = \begin{bmatrix} x_q(t) \\ y_q(t) \\ \theta_{jr}(t) \end{bmatrix} + R_q^w \begin{bmatrix} \underline{EFDs}(\delta_j, t) \\ 0 \end{bmatrix} \tag{2.25}$$

where $\theta_{jr} = tan^{-1}(\dot{y}_{jr}, \dot{x}_{jr})$, $R_q^w(t)$ is a rotational matrix relating the quadrotor $x$-$y$ coordinate frame to the world $x$-$y$ coordinate frame (see Fig. 2.8),

$$R_q^w(t) = \begin{bmatrix} \cos(\psi_q) & -\sin(\psi_q) & 0 \\ \sin(\psi_q) & \cos(\psi_q) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.26}$$

From (2.4), we have $\underline{EFDs}(\delta_j, t) = \begin{bmatrix} \overline{AB}_j & \overline{CD}_j \end{bmatrix}^T \in \mathbb{R}^{2\times1}$ denotes the desired relative offset vector of the $j^{th}$ mobile robot with respect to the leader coordinates given as,

$$\underline{EFDs}(\delta_j, t) = \begin{cases} \sum\limits_{k=1}^{n} (A_k(t)\cos(k\delta_j) + B_k(t)sin(k\delta_j)) \\ \sum\limits_{k=1}^{n} (C_k(t)\cos(k\delta_j) + D_k(t)sin(k\delta_j)) \end{cases} \tag{2.27}$$

where $A(t) \in \mathbb{R}^{n_h\times1}$, $B(t)$, $C(t)$ and $D(t)$ are the desired time-varying EFDs vectors owing the same size. The desired orientation of each robot is defined in such a way that the mobile robot is always facing its desired position. Now, in the following, we explain the design of the TVF tracking controller. In a single mobile robot control, a steering control input $\underline{v}_{jc}(t)$ is designed to solve three basic problems: path following, point stabilization, and trajectory following such that $\lim\limits_{t\to\infty}(q_{jr} - q_j) = 0$ and $\lim\limits_{t\to\infty}(\underline{v}_{jc} - \underline{v}_j) = 0$ [59]. If the mobile robot controller can successfully track a class of smooth velocity control inputs, then all three problems can be solved with the same controller. We will extend the stated three basic tracking control problems for a single mobile robot to the proposed formation control based Leader-Followers model.

**Definition 2.4** *The mobile robots formation system is said to achieve the output TVF tracking control if and only if for any given initial states $q_j(t=0)$, $\underline{v}_j(t=0)$, $j \in [1, N_R]$ there exists a control torque $\underline{\tau}_j$ , such as,*

$$\begin{cases} \lim\limits_{t\to\infty} \left\| q_j\left(t\right) - q_{jr}\left(t\right) \right\| = 0 \\ \lim\limits_{t\to\infty} \left\| \underline{v}_j\left(t\right) - \underline{v}_{jc}\left(t\right) \right\| = 0 \end{cases} ; j \in [1, N_R] \tag{2.28}$$

In this section, the aim is limited on finding a smooth control velocity input $\underline{v}_{jc} = f(e_j, \underline{EFDs}(\delta_j, t), \dot{\chi}_q, \dot{\eta}_q, j)$, in such a way $\lim\limits_{t\to\infty}\left(q_j\left(t\right) - q_{jr}\left(t\right)\right) = 0$, $j = [1, 2, .., N_R]$. In the next section (section-2.6), we focus on designing a robust torque-control input $\underline{\tau}_j$ for each mobile robot with dynamic behaviour described by (2.11) and (2.12), so that $\lim\limits_{t\to\infty}[\underline{v}_j\left(t\right) - \underline{v}_{jc}\left(t\right)] = 0$. Achieving this for all the mobile robots $j = 1, 2, \ldots N_R$, guarantees the mobile robots are tracking the formation reference trajectory, simultaneously, the formation is maintained and stabilized around the quadrotor planar coordinates $(x_q, y_q)$.

The formation time-varying tracking error expressed in the mobile robot frame coordinate is written as in (2.23). The UGVs error dynamics system are determined by differentiating (2.28), substituting (2.11), (2.25), and using trigonometric relations yields,

$$
\begin{aligned}
\dot{e}_j^x &= \begin{cases} -v_j + v_q \cos\left(\theta_j - \rho_q\right) + w_j e_j^y \\ -\dot{\psi}_q \sqrt{\left(\overline{AB}_j\right)^2 + \left(\overline{CD}_j\right)^2} \sin\left(e_j^\theta + \psi_q + \beta_1\right) \\ +\sqrt{\left(\dot{\overline{AB}}_j\right)^2 + \left(\dot{\overline{CD}}_j\right)^2} \cos\left(e_j^{\theta_j,\psi_q} + \beta_2\right) \end{cases} \\
\dot{e}_j^x &= \begin{cases} -w_j e_j^x + v_q \sin\left(\theta_j - \rho_q\right) + d_j w_j \\ +\dot{\psi}_q \sqrt{\left(\overline{AB}_j\right)^2 + \left(\overline{CD}_j\right)^2} \cos\left(e_j^\theta + \psi_q + \beta_1\right) \\ +\sqrt{\left(\dot{\overline{AB}}_j\right)^2 + \left(\dot{\overline{CD}}_j\right)^2} \sin\left(e_j^{\theta_j,\psi_q} + \beta_2\right) \end{cases} \\
\dot{e}_j^x &= \left\{\dot{\theta}_{jr} - w_j\right.
\end{aligned}
\tag{2.29}
$$

In (2.29), $v_q = \sqrt{\dot{x}_q + \dot{y}_q}$, $\rho_q = \tan^{-1}\left(\dot{y}_q, \dot{x}_q\right)$ and $e_j^{\theta_j,\psi_q} = \left(\theta_j - \psi_q\right)$. Furthermore $\left(\dot{\overline{AB}}_j\right)$ and $\left(\dot{\overline{CD}}_j\right)$ are the derivatives of $\left(\overline{AB}_j\right)$ and $\left(\overline{CD}_j\right)$. The parameters $\beta_1$, $\beta_2$ are defined as,

$$
\begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} tan^{-1}(\overline{CD}_j, \overline{AB}_j) \\ tan^{-1}(\dot{\overline{CD}}_j, \dot{\overline{AB}}_j) \end{bmatrix}
\tag{2.30}
$$

To stabilize dynamics error (2.29), the steering control input (2.31)-(2.33) is proposed for mobile robot $j$ to maintain the desired time-varying formation with respect to the quadrotor coordinates and yaw angle $(x_q, y_q, \psi_q)$,

$$
\begin{cases} v_{jc} = v_q \cos\left(\theta_j - \rho_q\right) + K_x e_j^x + \xi_{vj} \\ w_{jc} = \dot{\theta}_{jr} + \left(v_q + K_v\right) K_y e_j^y + \left(v_q + K_v\right) K_\theta sin\left(e_j^\theta\right) + \xi_{wj} \end{cases}
\tag{2.31}
$$

where $\xi_{vj}$ and $\xi_{wj}$ are given as,

$$
\xi_{vj} = \begin{cases} -\dot{\psi}_q \sqrt{\overline{\mathrm{AB}}_j^{\,2} + \overline{\mathrm{CD}}_j^{\,2}} \sin\left(e_j^\theta + \psi_q + \beta_1\right) \\ +\sqrt{\left(\dot{\overline{AB}}_j\right)^2 + \left(\dot{\overline{CD}}_j\right)^2} \sin\left(e_j^\theta + \psi_q + \beta_2\right) \end{cases}
\tag{2.32}
$$

$$
\begin{aligned}
\xi_{wj} = -\frac{\left|e_j^y\right|}{1/K_y + \left|e_j^y\right| d_j} & \left[\dot{\psi}_q (d_j + \sqrt{\overline{\mathrm{AB}}_j^{\,2} + \overline{\mathrm{CD}}_j^{\,2}})\right. \\
& \left. +(v_q + K_v) K_\theta d_j + \sqrt{\left(\dot{\overline{AB}}_j\right)^2 + \left(\dot{\overline{CD}}_j\right)^2} + K_v \right]
\end{aligned}
\tag{2.33}
$$

**Remark 2.5** *If we compare the proposed formation controller* (2.31) *with the robots deployment controller* (2.24)*, we find that* (2.32) *and* (2.33) *are new terms. These new terms guarantee the stability of the overall mobile robots formation. Furthermore, the parameter $K_v > 0$*

*is introduced to ensure that the asymptotic stability holds even when the quadrotor velocity* $v_q = 0$.

Before we proceed the stability analysis, the following assumptions are needed,

**Assumption 1.** *Each mobile robot in the formation is wirelessly connected to the quadrotor, whose control inputs* $u_z$ *and* $\tau_q$ *are bounded* $\forall t > 0$.

**Assumption 2.** *The desired time-varying formation defined by the EFDs vectors (i.e.,* $A(t)$, $B(t)$, $C(t)$ *and* $D(t)$*) as well as the leader's state and its control inputs are communicated to whole formation mobile robots.*

**Assumption 3.** *The vectors* $A(t)$, $B(t)$, $C(t)$ *and* $D(t)$ *used in (2.27) are continuous, differentiable, and bounded. Furthermore,* $v_q(t)$ *and* $\dot{\psi}_q(t)$ *are bounded* $\forall t > 0$.

**Assumption 4.** *The perfect velocity tracking hold such* $\underline{v}_j = \underline{v}_{jc}$ *(this assumption will be removed later).*

**Theorem 2.6** *[49] Let a formation of* $N_R$ *nonholonomic mobile robots robots with dynamics described by (2.11) and (2.12), owning* $n$ *generalized coordinates* $q_j$, $r$ *actuators, and* $m$ *independent constraints, guided by a quadrotor satisfying (2.28). Let Assumption* $1-4$ *hold. The* $j^{th}$ *mobile robot will be steered by a smooth velocity control input* $\underline{v}_{jc}$ *given by (2.31)-(2.33). Then, the origin* $\underline{e}_j = 0$ *consisting of the position and orientation error for the mobile robots formation* $j = 1, ..., N_R$ *are asymptotically stable.*

***Proof of Theorem-2.6:*** Consider the following candidate Lyapunov function,

$$\mathbb{Q}_j = \frac{1}{2}\left((e_j^x)^2 + (e_j^y)^2\right) + \frac{1-\cos(e_j^\theta)}{K_y} \tag{2.34}$$

We can notice that $\mathbb{Q}_j \geqslant 0$ and $\mathbb{Q}_j = 0$ only if $\underline{e}_j = 0$. Thus, $\mathbb{Q}_j$ is positive definite with respect to $\underline{e}_j$. The time-derivative of (2.34) with the substitution of (2.31) and (2.32), yields,

$$\begin{aligned}\dot{Q}_j = & -K_x\left(e_j^x\right)^2 - d_j K_y (v_q + K_v)\left(e_j^y\right)^2 - \left(v_q + K_v\right)\frac{K_\theta}{K_y}\sin^2\left(e_j^\theta\right) \\ & -d_j e_j^y \left[\begin{array}{c} \dot{\psi}_q + \left(v_q + K_v\right) K_\theta \sin\left(e_j^\theta\right) \\ -\frac{\dot{\psi}_q}{d_j}\sqrt{\overline{\mathrm{AB}}_j^{\,2} + \overline{\mathrm{CD}}_j^{\,2}}\cos\left(e_j^\theta + \psi_q + \beta_2\right) \\ -\frac{\sqrt{\left(\dot{\overline{AB}}_j\right)^2 + \left(\dot{\overline{CD}}_j\right)^2}}{d_j}\sin\left(e_j^\theta + \psi_q + \beta_2\right) + \frac{K_v \sin\left(e_j^\theta\right)}{d_j} \end{array}\right] \\ & -\xi_{wj}\left[\frac{\sin\left(e_j^\theta\right)}{K_y} + e_j^y d_j\right]\end{aligned} \tag{2.35}$$

The equation (2.35) could be expressed as an inequality,

$$\begin{aligned}\dot{Q}_j \leqslant & -K_x\left(e_j^x\right)^2 - d_j K_y \left(v_q + K_v\right)\left(e_j^y\right)^2 - \left(v_q + K_v\right)\frac{K_\theta}{K_y}\sin^2\left(e_j^\theta\right) \\ & +\left|e_j^y\right| \left(\begin{array}{l} \dot{\psi}_q\left(d_j + \sqrt{\overline{\mathrm{AB}}_j^{\,2} + \overline{\mathrm{CD}}_j^{\,2}}\right) + \left(v_q + K_v\right) K_\theta d_j \\ +\sqrt{\left(\dot{\overline{AB}}_j\right)^2 + \left(\dot{\overline{CD}}_j\right)^2} + K_v \end{array}\right) \\ & +\xi_{wj}\left[\frac{1}{K_y} + \left|e_j^y\right| d_j\right]\end{aligned} \tag{2.36}$$

In (2.36), it is noticeable that the first three terms are negative whatever $\underline{e}_j \neq 0$. Moreover, when substituting (2.33) into the terms remaining in (2.36), yields,

$$\begin{aligned}& \left|e_j^y\right| \left[\begin{array}{l} \dot{\psi}_q\left(d_j + \sqrt{\overline{\mathrm{AB}}_j^{\,2} + \overline{\mathrm{CD}}_j^{\,2}}\right) + \left(v_q + K_v\right) K_\theta d_j \\ +\sqrt{\left(\dot{\overline{AB}}_j\right)^2 + \left(\dot{\overline{CD}}_j\right)^2} + K_v \end{array}\right] \\ & +\xi_{wj}\left[\frac{1}{K_y} + \left|e_j^y\right| d_j\right] \leqslant 0\end{aligned} \tag{2.37}$$

Now, substituting (2.37) into (2.36) yields,

$$\dot{Q}_j \leqslant \left[\begin{array}{l} -K_x\left(e_j^x\right)^2 - d_j K_y \left(v_q + K_v\right)\left(e_j^y\right)^2 \\ -\left(v_q + K_v\right)\frac{K_\theta}{K_y}\sin^2\left(e_j^\theta\right) \end{array}\right] = \Pi \leqslant 0 \tag{2.38}$$

It is clear that $\dot{Q}_j \leqslant 0$, yields that $Q_j$ is bounded $|Q_j| < \infty$, and consequently $e_j^x$, $e_j^y$ and $e_j^\theta$ are bounded too. Further, we have $\int_0^\infty \Pi dt \leqslant Q_j(\infty) - Q_j(0)$ has a finite limit since $\dot{Q}_j \leqslant 0$ and $Q_j(t) \geq 0$. And from (2.29) and assumption-3, we get $\left\|\underline{e}_j\right\|$ and $\left\|\underline{\dot{e}}_j\right\|$ are bounded, i.e., $\Pi$ is uniformly continuous. As we have $\int_0^\infty \Pi dt$ does not increase and converge to some constant value, then by Barbalat's Lemma [63], $\Pi \to 0$ as $t \to \infty$, which implies from (2.38) that $\Pi \equiv 0 \Rightarrow \underline{e}_j = \begin{bmatrix} e_j^x & e_j^y & e_j^\theta \end{bmatrix}^T = 0$. Consequently, the pseudo velocity control (2.31)-(2.33) guaranties that the error system (2.23) and (2.29) is stable and $\underline{e}_j \to 0$ as $t \to \infty$.

## 2.6 Robust Torque-Control (Backstepping-like Feedback Linearization)

Hereafter, the *Assumption-4* is invalid. The proposed UGVs TVF tracking control task is based on robust cascaded velocity/torque control. In the previous section, a pseudo velocity control input $\underline{v}_{jc}(t)$, i.e., (2.31)-(2.33) was designed based on kinematic model. The aim is to design a robust torque-control input $\underline{\tau}_j(t)$ for each mobile robot with dynamic behavior described by (2.11) and (2.12), such that $\lim\limits_{t\to\infty} \left[\underline{v}_j(t) - \underline{v}_{jc}(t)\right] = 0$ (see Definition-2.4). Achieving this for $j = 1, 2, ..., N_R$ guarantees the mobile robots are tracking the formation reference trajectory, and simultaneously the formation is maintained and stabilized around the quadrotor coordinates $(x_q, y_q)$. Before we start presenting the torque-control design, let introduce the following assumption.

**Assumption 5.** *The disturbance $\bar{\tau}_{dj}(t) = \begin{bmatrix} \bar{\tau}_{dj,1}(t) & \bar{\tau}_{dj,2}(t) \end{bmatrix}^T$ is unknown and satisfies $\left|\bar{\tau}_{dj,i}(t)\right|_{abs} \le \bar{\tau}_{d,Max,i}$, $i = 1, 2$, $\forall t$, where $\bar{\tau}_{d,Max} \in \mathbb{R}^{2\times 1}$ is a vector with positive entries denoting the disturbance upper-bound.*

The $j^{th}$ mobile robot velocity tracking error is written as follows,

$$\underline{e}_{vjc} = \underline{v}_{jc} - \underline{v}_j \tag{2.39}$$

In order to write the $j^{th}$ mobile robot dynamics as a function of $\underline{e}_{vjc}$ and $\underline{\dot{e}}_{vjc}$, we add and subtract $\overline{M}_j(q_j)\underline{\dot{v}}_{jc}$ and $\overline{V}_{mj}(q_j)\underline{v}_{jc}$ to (2.12), yields,

$$\overline{M}_j(q_j)\,\underline{\dot{e}}_{vjc} = -\overline{V}_{mj}(q_j, \dot{q}_j)\,\underline{e}_{vjc} - \overline{B}_j\underline{\tau}_j + f_j(\varepsilon_j) + \bar{\tau}_{dj} \tag{2.40}$$

where $f_j(\varepsilon_j)$ is given as,

$$f_j(\varepsilon_j) = \overline{M}_j(q_j)\,\underline{\dot{v}}_{jc} + \overline{V}_{mj}(q_j, \dot{q}_j)\,\underline{v}_{jc} + \overline{F}_j(\underline{v}_j) \tag{2.41}$$

where $\varepsilon_j = [\ddot{\chi}_q, \dot{\chi}_q, \ddot{\eta}, \dot{\eta}, q_j, \underline{v}_j, \underline{e}_j, \dot{\underline{e}}_j]$. The quadrotor dynamics will be brought to (2.40) by the function $f_j\left(\varepsilon_j\right)$ through the term $\dot{\underline{v}}_{jc}$,

$$\dot{\underline{v}}_{jc} = f_{vjc}(\ddot{\chi}_q, \dot{\chi}_q, \ddot{\eta}, \dot{\eta}, q_j, \underline{e}_j, \dot{\underline{e}}_j) \tag{2.42}$$

From (2.31), (2.8), and (2.9), the quadrotor dynamics can be written in function of $\tau_q$, $u_z$, $\dot{\eta}$ and $\eta$. Substituting (2.8) and (2.9) into (2.42) results in the dynamics of the quadrotor to become a part of $\dot{\underline{v}}_{jc}$ as,

$$\dot{\underline{v}}_{jc} = f_{vjc}(\tau_q, u_z, \dot{\eta}, \eta, \underline{e}_j, \dot{\underline{e}}_j) \tag{2.43}$$

The term $\dot{\underline{v}}_{jc}$ is bounded and constructible by the $j^{th}$ mobile robot relying on assumptions 1 and 2. The function $f_{vjc}$ can be accurately approximated using a local neural network. We define the auxiliary control input $\underline{u}_j$ in (2.15) as,

$$\underline{u}_j = \dot{\underline{v}}_{jc} + K_4 \underline{e}_{vjc} \tag{2.44}$$

where $K_4 \in \mathbb{R}^{2\times 2}$ is a diagonal gain matrix, with positive constants. Now, we propose the following robust torque-control input for the mobile robot $j$ by substituting (2.44) into (2.14) and adding a signum function,

$$\underline{\tau}_j = \overline{B}_j^{-1}\left[\overline{M}_j K_4 \underline{e}_{vjc} + f_j\left(\varepsilon_j\right) - \gamma.sign(\underline{e}_{vjc})\right] \tag{2.45}$$

where $\gamma = \begin{bmatrix} \gamma_1 & \gamma_2 \end{bmatrix}^T \in \mathbb{R}^{2\times 1}$ is a vector with positive constants that satisfies $\gamma_i > \bar{\tau}_{d,Max,i}$, $i = 1, 2$ and the operator $(.)$ refers to element by element matrix multiplication.

Substituting (2.45) into the dynamics of the $j^{th}$ mobile robot (2.12) and using (2.41) produces the closed loop error dynamics shown in (2.46),

$$\overline{M}_j \dot{\underline{e}}_{vjc} = -\left(\overline{M}_j K_4 + \overline{V}_{mj}\right)\underline{e}_{vjc} + \bar{\tau}_{dj} - \gamma.sign(\underline{e}_{vjc}) \tag{2.46}$$

A general TVF tracking control structure for an individual mobile robot $j, j \in [1, ..., N_R]$ is presented in Fig. 2.9.

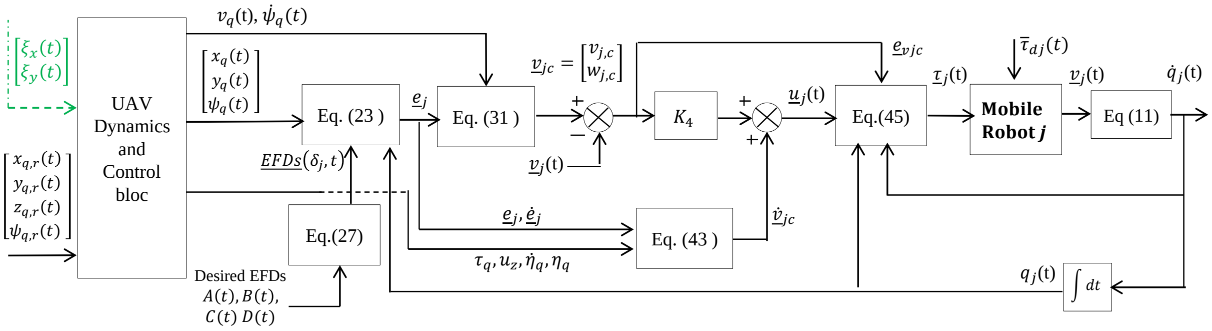


Figure 2.9: TVF tracking control design architecture of the $j^{th}$ UGV.

It is worth noting that the role of the leader-UAV is not only to provide its state information $[x_q, y_q, \psi_q]$ and the desired formation shape (i.e., EFDs vectors) for the UGVs followers, but also in providing its dynamics information and its control inputs $\ddot{\chi}_q, \dot{\chi}_q, \ddot{\eta}, \dot{\eta}, \tau_q$, with $\tau_q = [u_\phi, u_\theta, u_\psi]^T$. The UGVs dynamics and particularly the UAV dynamics have an important impact on the UGVs formation stability and tracking accuracy, especially when the UAV exhibits high dynamics. In the following, we will explain the UAV trajectory tracking control design.

It should be pointed out that The dynamic model of the quadrotor can be divided into two subsystems. An under-actuated subsystem, which consists of the position $x, y$, the roll and the pitch angles $\phi, \theta$ dynamics. And a full actuated sub-system, which consists of the altitude $z$ and the yaw angle $\psi$ dynamics. Thus, the quadrotor $x, y$ states are controlled by the $\phi, \theta$ angles respectively, while the $\phi, \theta$ states are controller by the control inputs $u_\phi, u_\theta$ respectively.

Refer to Fig. 2.10, the control strategy of the quadrotor system is divided into two parts in cascade. The first control part is the altitude $z$ and the yaw angle $\psi$ control, yielding the control inputs $u_z$ and $u_\psi$ respectively.

The second control part will be designed for the under-actuated sub-system to control in first step the quadrotor position $x, y$, yielding the virtual control input $u_x$ and $u_y$ respectively (see Section-2.2.2). By using (2.47), the reference roll and pitch angles $\phi_{q,r}$, $\theta_{q,r}$ are obtained. In the second step, the $\phi_{q,r}$, $\theta_{q,r}$ will feed-forward the roll and

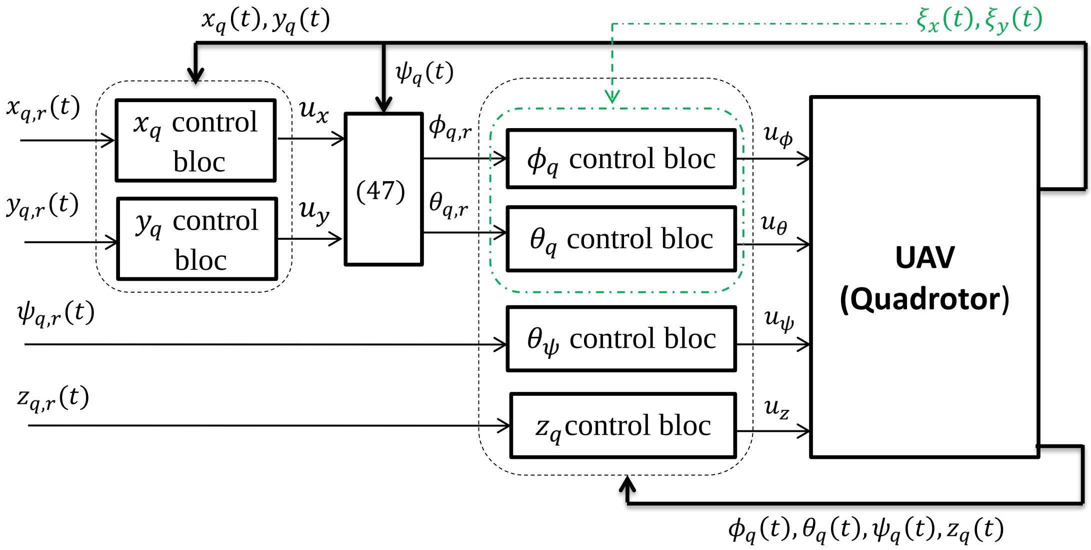


Figure 2.10: UAV control system block diagram.

pitch control bloc to provide the control inputs $u_\phi, u_\theta$ respectively.

$$\begin{aligned} \phi_{q,r} &= \arcsin\left[u_x \sin(\psi_q) - u_y \cos(\psi_q)\right] \\ \theta_{q,r} &= \arcsin\left[\frac{u_x \cos(\psi_q) + u_y \sin(\psi_q)}{\cos(\phi_{q,r})}\right] \end{aligned} \tag{2.47}$$

All the quadrotor states are controlled by the simplest control law which is a PID controller. However, other robust controllers may be selected. The PID-controller takes many structures but the most common one has the following form,

$$u_k = K_p\left[e_k + \frac{1}{T_i}\int_0^t e_k dt + T_d\frac{de_k}{dt}\right] \tag{2.48}$$

where the subscript $k$ refers to the state being controlled (i.e., $x, y, z, \phi, \theta$, and $\psi$), and $K_p, T_i, T_d$ refer to the PID gains.

**Remark 2.7** *If we replace the UAV-leader with a UGV, basically, the same developments can be applied to align the new UGVs-formation tracking controller with UGV-leader dynamics under the following changes,*

- *In the formation deployment control (2.24), the control design will remain the same, because it is not dependent to the leader dynamics. The control design in this stage depends only on the IPF i.e., $\mathcal{H}(x, y)$.*

- *In the formation tracking control based kinematic model (2.31)-(2.33), we replace $[x_q, y_q, \psi_q]$ by $[x_{ugv}, y_{ugv}, \theta_{ugv}]$ in (2.25)-(2.26). Consequentially, after redeveloping the derivative of the UGV formation tracking error (2.29), the parameters $v_q$, $\psi_q$, $\rho_q$ and $e_j^{(\theta_j,\psi_q)}$ in (2.29) and (2.31)-(2.33) will be replaced by $v_{ugv}$, $w_{ugv}$, $\theta_{ugv}$ and $e_j^{(\theta_j,\theta_{ugv})}$ respectively. The UGV-leader in this case has geometry as in Fig. 2.3, with behavior dynamics described by (2.11) and (2.12). In addition, $[x_{ugv}, y_{ugv}, \theta_{ugv}]$ and $v_{ugv}$, $w_{ugv}$ denote the leader-UGV generalized coordinates and its linear and angular velocities respectively.*

- *In the UGVs torque-control design based dynamic model (2.45), the dynamics information of the leader-UGV (i.e., $v_{ugv}$, $w_{ugv}$, $\theta_{ugv}$, $\underline{\tau}_{ugv}$) will be brought to the UGVs-torque-control input through the term $\underline{\dot{v}}_{jc} = f_{vjc}(v_{ugv}, w_{ugv}, \underline{\tau}_{ugv}, \underline{e}_j, \underline{\dot{e}}_j)$ in (2.43). With $\underline{\tau}_{ugv}$ denoting the UGV-leader torque control vector. As a result, (2.45) will be updated accordingly.*

**Remark 2.8** *In order to improve the UGVs formation stability and tracking accuracy, we can take into consideration the overall UGVs formation tracking error when designing the UAV control. Thus, we can feed-forward the UAV roll and pitch control blocs by the overall UGVs formation tracking error $\xi_x$, $\xi_y$ along the x and y axes respectively (refer to Fig. 2.10), with $\xi_x = \sum_{j=1}^{N_R} \left[e_j^x\right]^2$ and $\xi_y = \sum_{j=1}^{N_R} \left[e_j^y\right]^2$. As such, when the UGVs formation is drifting from the desired shape, the terms $\xi_x$, $\xi_y$ will take place and will drive slightly the UAV by affecting $u_\phi$ and $u_\theta$ in such a way to minimize the values of $\xi_x$, $\xi_y$. However, this will come at the cost of temporary shifting the UAV from tracking its reference trajectory. And when the UGVs progressively regain their desired formation, the terms $\xi_x$, $\xi_y$ would be decreasing so the UAV will get back to accurately track its reference trajectory. The insertion of the UGVs formation tracking error in the UAV control design yields a cross coupling between the UAV and the UGVs, which will result in the necessity to provide the UAV stability analysis.*

**Theorem 2.9** *[49] Let $K_4$ in (2.45) be a diagonal matrix with positive constants, $\gamma \in \mathbb{R}^{2\times 1}$ is vector that satisfies $\gamma > \overline{\tau}_{d,Max}$, and suppose assumptions $1,2,3,5$ are valid. We consider the pseudo velocity control-input $\underline{v}_{jc}(t)$ for the $j^{th}$ mobile robot as defined by (2.31)-(2.33). We apply the torque-control input given in (2.45) to the formation of mobile robots $j = 1,2,\dots N_R$, with dynamics described by (2.12). Therefore, the position, orientation and velocity tracking errors ($\underline{e}_j$ and $\underline{e}_{vjc}$, $j = [1,2,\dots N_R]$) are stable and ultimately bounded around the origin.*

***Proof of Theorem-2.9:*** Let $\tilde{Q}_j$ be a Lyapunov candidate function defined as,

$$\tilde{Q}_j = Q_j + \frac{1}{2}\underline{e}_{vjc}^T \overline{M}_j \underline{e}_{vjc} \tag{2.49}$$

where $Q_j$ is already defined in (2.34). $\tilde{Q}_j \geq 0$ and $\tilde{Q}_j = 0$ only if $\underline{e}_j = 0$ and $\underline{e}_{vjc} = 0$. Differentiating (2.49) with respect to time yields,

$$\dot{\tilde{Q}}_j = \dot{Q}_j + \left(\underline{e}_{vjc}\right)^T \overline{M}_j \dot{\underline{e}}_{vjc} + \frac{1}{2}\left(\underline{e}_{vjc}\right)^T \dot{\overline{M}}_j \underline{e}_{vjc} \tag{2.50}$$

It has been shown that $\dot{Q}_j \leqslant 0$ from the proof of Theorem-2.6. By substituting (2.46) into (2.50) yields,

$$\begin{aligned} \dot{\tilde{Q}}_j = {} & \dot{Q}_j - \underline{e}_{vjc}^T \overline{M}_j K_4 \underline{e}_{vjc} + \frac{1}{2}\underline{e}_{vjc}^T (\dot{\overline{M}}_j - 2\overline{V}_{mj})\underline{e}_{vjc} \\ & + \underline{e}_{vjc}^T \overline{\tau}_{dj}(t) - \underline{e}_{vjc}^T \gamma.sign(\underline{e}_{vjc}) \end{aligned} \tag{2.51}$$

After applying the *skew-symmetric property*, given by $X^T\left(\dot{\overline{M}}_j - 2\overline{V}_{mj}\right)X = 0, \forall X \in \mathbb{R}^2$ [62], and relying on Assumption-5, (2.51) can be expressed as,

$$\dot{\tilde{Q}}_j \leqslant \begin{bmatrix} \dot{Q}_j - \underline{e}_{vjc}^T \overline{M}_j K_4 \underline{e}_{vjc} \\ -\left|\underline{e}_{vjc}^T\right|_{abs}(\gamma - \overline{\tau}_{d,Max}) \end{bmatrix} = \tilde{\Pi} \leqslant 0 \tag{2.52}$$

where $|.|_{abs}$ denotes the absolute value of some vector. Examining (2.52), it is obvious that $\dot{\tilde{Q}}_j \leqslant 0$ since $\overline{M}_j$ and $K_4$ are definite positive matrices, and $\gamma$ satisfies $\gamma_i > \overline{\tau}_{d,Max,i}$, $i = 1,2$. Similar to the proof of *Theorem-2.6*, knowing that $\dot{\tilde{Q}}_j \leqslant 0$, yields that $\tilde{Q}_j$ is bounded $\left\|\tilde{Q}_j\right\| \leq \infty$, and consequently $\underline{e}_j$ and $\underline{e}_{vjc}$ are bounded too. In the other side,

$\int_0^\infty \tilde{\Pi} dt \leqslant \widetilde{Q}_j(\infty) - \widetilde{Q}_j(0)$ has a finite limit since $\dot{\widetilde{Q}}_j \leqslant 0$ and $\widetilde{Q}_j(t) > 0$, i.e., $\tilde{\Pi}$ is uniformly continuous, by Barbalat's lemma [63], $\tilde{\Pi} \to 0$ as $t \to \infty$, which implies from (2.52) that $\tilde{\Pi} \equiv 0 \Rightarrow \begin{bmatrix} e_j & e_{vjc} \end{bmatrix}^T = 0$. Then, from (2.38) and (2.52), we get the velocity tracking error $\underline{e}_{vjc}$ followed by the position and orientation tracking error $\underline{e}_j$ are uniformly stable. However, due to the signum term in (2.45), the tracking errors $\underline{e}_{vjc}$ and $\underline{e}_j$ will not exactly converge to zero, instead they will be ultimately bounded and close to a neighbourhood of the origin, as it is demonstrated in simulation section, and the ultimate bounds are dependent on the selected parameter $\gamma$, which in its turn depends on the estimation of the disturbance upper-bound $\bar{\tau}_{d,Max}$. Further, we can replace the signum term in (2.45) by a saturation function in order to reduce the chattering effect. However in that case, the system stability needs to be restudied.

## 2.7 Simulation and experimental results

### Simulation result

In this section, simulation studies are conducted to show the results of applying the proposed UAV-UGVs formation controls. The aim of the simulation results is, (i) to confirm the theoretical conjecture and to show the effectiveness of the proposed deployment and UGVs TVF tracking controllers, (ii) to highlight and to demonstrate the improvement brought to UGVs formation in terms of stability, tracking accuracy and robustness when UGVs dynamics and particularly UAV (dynamics and control inputs) information are considered in the torque control design, in the presence of unknown disturbance and unmodeled dynamics. The obtained simulation results were carried out using three identical mobile robots guided by a quadrotor in Matlab environment. The quadrotor's trajectory planning and the process of producing the desired formation shape are not within the scope of this chapter. Thus, the EFDs vectors and the quadrotor's reference trajectory $(x_{q,r}, y_{q,r}, z_{q,r}, \psi_{q,r})$ are predefined. The quadrotor is tracking its predefined reference trajectory using a PID controller (refer to Fig. 2.10).

To respond the simulation aims, two simulation cases were considered. In the case-1, we used the controllers (2.24) and (2.31)-(2.33) with perfect velocity tracking assumption (i.e., $\underline{v}_j = \underline{v}_{jc}$ and $\dot{\underline{v}}_j = \dot{\underline{v}}_{jc}$) which means that the UGVs and the UAV dynamics are ignored. In the case-2, the controllers (2.24), (2.31)-(2.33) and (2.45) are used, implying that the UGVs and the UAV dynamics/control information are incorporated in the UGVs torque-control design. The obtained results of case-2 are compared with the ones of the case-1. In both cases, two tasks are performed by the UGVs formation namely, deployment task and TVF tracking task.

Hereafter, the position states $x_j, y_j$ and the heading $\theta_j$ are expressed in Meter and Radian units respectively. The mobile robots initial posture are chosen as, $q_1 = \begin{bmatrix} -5.4 & 15.96 & 0 \end{bmatrix}^T$, $q_2 = \begin{bmatrix} 4.76 & 30.71 & 0.05 \end{bmatrix}^T$ and $q_3 = \begin{bmatrix} 11.52 & 7.36 & 1.58 \end{bmatrix}^T$. In deployment task, we consider that the initial desired shape (i.e., EFDs vectors) are given,

$$x_q(t=0) = 10m; y_q(t=0) = 20m; n_h = 2$$
$$A_1 = \begin{bmatrix} 7 \\ 0.5 \end{bmatrix}; B_1 = \begin{bmatrix} 3 \\ 0 \end{bmatrix}; C_1 = \begin{bmatrix} 1 \\ 0.2 \end{bmatrix}; D_1 = \begin{bmatrix} 7 \\ 1 \end{bmatrix}$$

The corresponding IPF denoted by $\mathcal{H}(x,y)$,

$$\mathcal{H}(x,y) = 10^{-3}x^4 - 1/2600x^3y - 9\times10^{-3}x^3 + 5\times10^{-4}x^2y^2 - 0.026x^2y$$
$$+0.39x^2 - 10^{-4}xy^3 + 4\times10^{-3}xy^2 - 0.1xy - 0.57x + 2.5\times10^{-3}y^3 - 0.04y^2$$
$$-1.8y + 27.3$$

The quadrotor's mechanical parameters were taken from [58](Table.E). During the Task-2, we suppose that the quadrotor reference trajectory is given as in Table 2.1, and the formation shape is described by the desired time-varying EFDs vectors,

$$A_2(t) = A_1; B_2(t) = B_1 - 3\sin(0.05t\pi);$$
$$C_2(t) = C_1 - \sin(0.05t\pi); D_2(t) = D_1.$$

The following gains and parameters were utilized for the controllers,

$$K_x = 1.2; K_y = 1.2; K_\theta = 1.5; K_v = 5; \lambda_1 = 0.5;$$
$$K_4 = 1.5\times \mathrm{I}_{2\times2}; M_s = 50; dt = 0.02s.$$

Table 2.1: Quadrotor reference trajectory.

| | $t \in [0,4s]$ | $t \in [4,16s]$ | $t \in [16,20s]$ |
|---|---|---|---|
| $x_{q,r}(m)$ | $10+5t$ | $30+15cos(0.26t-2.61)$ | 30-5t |
| $y_{q,r}(m)$ | 20 | $30+15sin(0.26t-2.61)$ | 50 |
| $z_{q,r}(m)$ | 5 | 7 | 6 |
| $\psi_{q,r}(Rad)$ | 0 | $0.26t$ | $\pi$ |

We used the following parameters of the mobile robots in the simulation, $m = 15kg$, $I = 0.71kg.m^2$, $R_w = 0.1m$, $L_w = 0.32m$ and $d_j = 0.05m$. The added friction to the mobile robots dynamics and the unknown external disturbance are given as,

$$\bar{\tau}_{dj} = [\ 0.7e^{-0.5t} + \delta 0.3\text{cos2t} \quad 0.3e^{-0.3\delta t} + 0.12\text{sin3t}\ ]^T$$

$$\bar{F}_j = [\ \delta 0.1v_j \quad (1-\delta)0.12w_j\ ]^T; j \in [1,3]$$

where $\delta$ is random umber between 0 and 1 with uniform distribution. We choose $\gamma = [\ 1.15 \quad 0.435\ ]^T > \bar{\tau}_{d,Max}$. Fig. 2.11 depicts the obtained mobile robots trajectories during the deployment process followed by the TVF tracking task for the case-2, using the formation controllers (2.24), (2.31)-(2.33), and (2.45). It can be seen that the three mobile robots have successfully reached the initial desired 2D desired curve, where the curve center $(a_0, c_0)$ corresponds to the quadrotor's initial coordinates $[x_q(t=0), y_q(t=0)]$. After the deployment task was completed, the mobile robots have tracked the quadrotor trajectory while maintaining the desired time-varying geometric shape.

In Fig. 2.12, we see the evolution of the IPF during the deployment task for each mobile robot in case-2 using (2.24). The distance between the desired 2D curve and each robot' position is expressed by the IPF. It's obvious that the IPFs are decreasing exponentially during the robots convergence process towards the desired 2D curve perimeter. The UAV-leader position and angles tracking errors $[e_q^x, e_q^y, e_q^z]$, $[e_q^\phi, e_q^\theta, e_q^\psi]$ and its corresponding control inputs $\tau_q = [u_\phi, u_\theta, u_\psi]^T$ are depicted in Fig. 2.13. In Fig. 2.14, we show the trajectory tracking errors of the three mobile robots along the x-axis, the y-axis and the heading error for two the cases. The comparison shows that the TVF is seemed to be greatly degraded and not maintained in case-1, where we

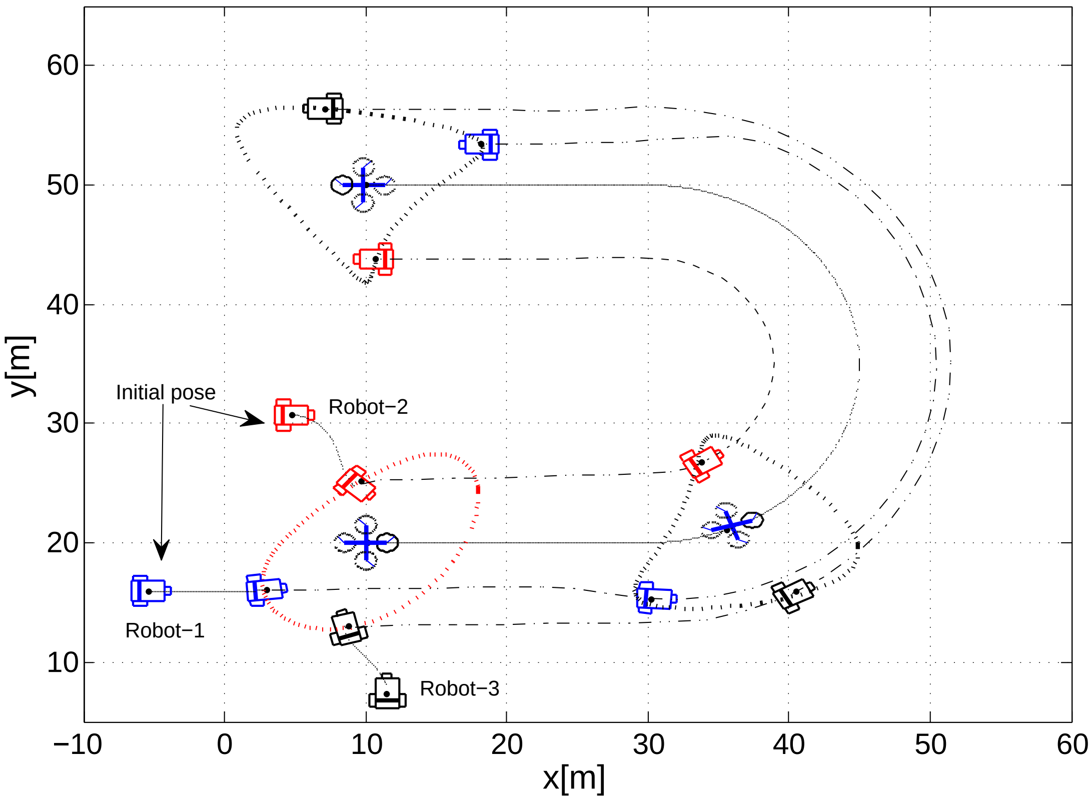


Figure 2.11: Formation deployment and TVF tracking by three mobile robots (case-2 using (2.24), (2.31)-(2.33), and (2.45) controllers.

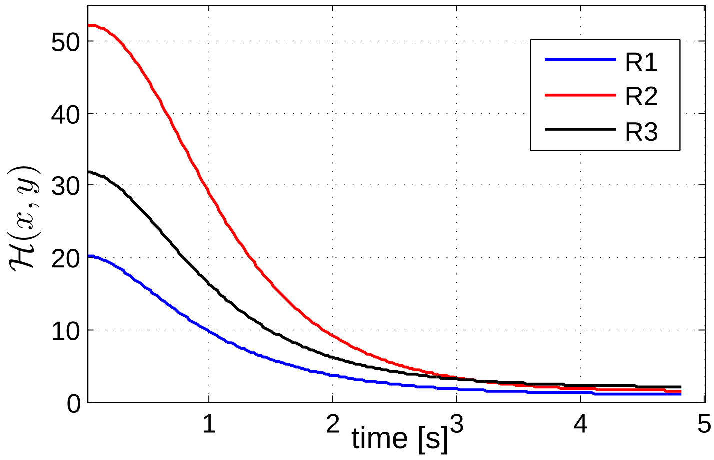


Figure 2.12: IPF $\mathcal{H}(x_j, y_j), j \in [1, 3]$ in the deployment task (case-2 using the controller (2.24).

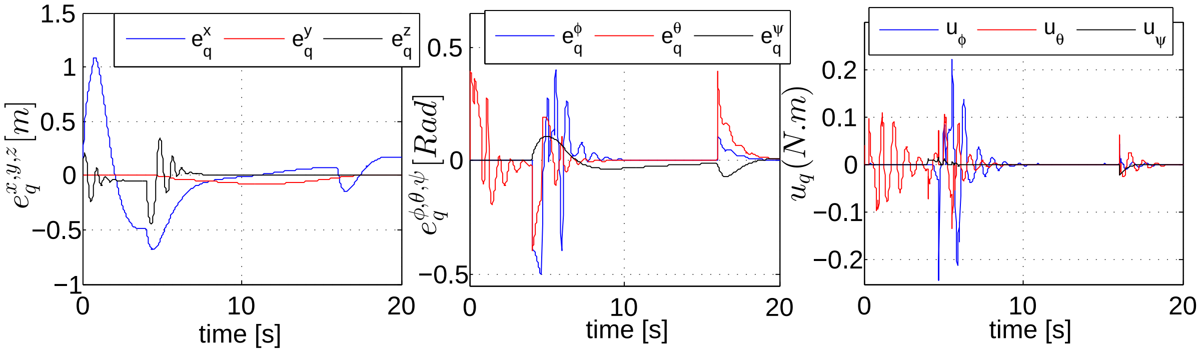
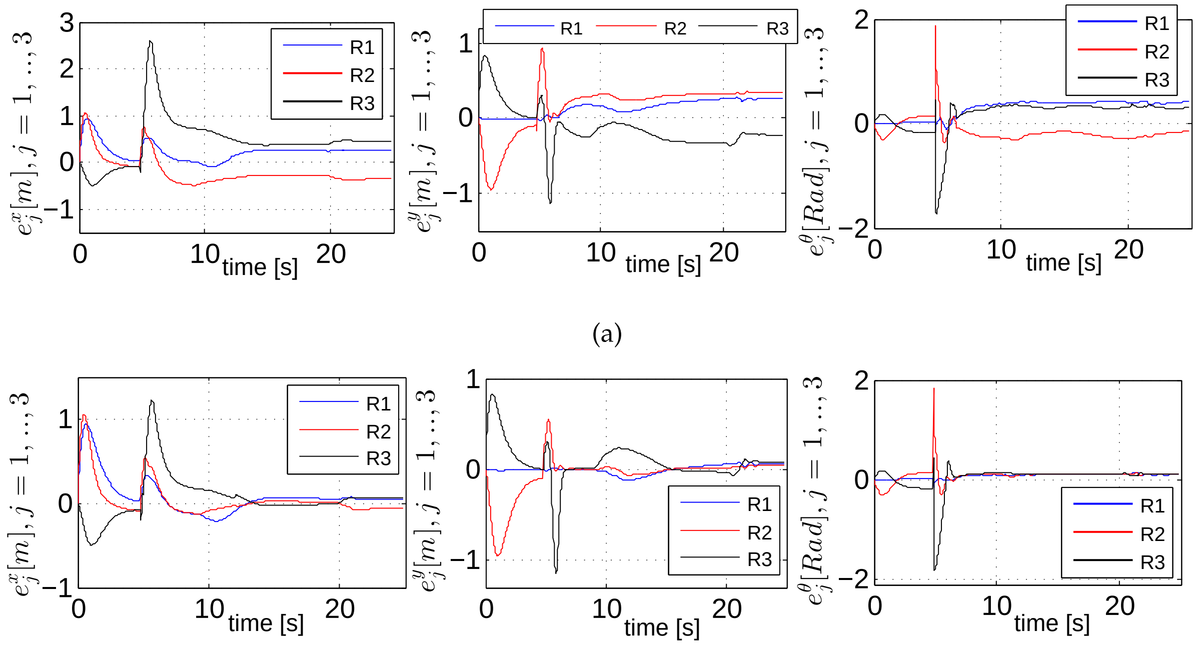


Figure 2.13: UAV position and angles tracking error $[e_q^x, e_q^y, e_q^z]$, $[e_q^\phi, e_q^\theta, e_q^\psi]$ respectively and the corresponding control inputs $\tau_q = [u_\phi, u_\theta, u_\psi]^T$.

notice unsteady tracking-errors state, in contrast with the case-2, in which, the settling time and the overshot decreased significantly and further the tracking-errors state are bounded around zero. The comparison demonstrates the improvement brought to the UGVs formation tracking behaviour in terms of smooth convergence and accurate tracking, when considering the UGVs and the UAV dynamics and control inputs information (i.e., case-2). The improvement can be explained further by the fact that the control law (2.45) includes an inner velocity tracking-loop and a discontinuous term introduced to attenuate the influence of the external disturbance. In Fig. 2.14(b), the errors tend to decrease till becoming almost nil at time $t = 4.9s$, which corresponds to the end of the deployment task. From $t = 4.9s$, we notice an increase of the $x$-axis errors and the heading errors for the three robots, this is due to the robots initial heading at the beginning of the second task (see Fig. 2.11), as well as the non-holonomic constraints that prevent the robots from moving along $y$-axis, then quickly the tracking-errors tend to almost a certain bound, so the time-varying formation is nearly maintained.

In Table 2.2, a numerical comparison in terms of the tracking-error overshot and the absolute tracking-error mean values along the $x$ and $y$ axes respectively is provided. It is straightforward to conclude that the UGVs formation tracking performance in terms of accurate formation tracking, stability, and disturbance attenuation have been significantly improved in case-2, compared with the case-1, which confirms the theoretical conjecture and the effectiveness of the UGVs robust torque-control.



(a)

(b)

Figure 2.14: Formation tracking errors, (a) Case-1 using (2.24), (2.31)-(2.33) controllers, (b) Case-2 using (2.24), (2.31)-(2.33) and (2.45) controllers.

Fig. 2.15 and Fig. 2.16 show the actual velocity and the torque control inputs $\underline{v}_j$, $\underline{\tau}_j$ of the three mobile robots respectively in case-2, where some oscillations are observed due to the selection of the control gains and the effect of the signum term in $\underline{\tau}_j$ that produces the chattering phenomenon.

## Experimental result

In this section, the aim is to demonstrate experimentally the feasibility and the effectiveness of the proposed deployment-controller (2.24) and the TVF tracking-controller (2.31)-(2.33) respectively. The validation was carried out using a group of three mobile robots type festo's Robotino$^{(\mathbb{R})}$ [64], in an area of $4m \times 6m$. Motion capture system (OptiTrack system) has been used to provide the positions and orientations of the mobile robots with high precision. It consists of a set of 4 cameras model Prime$^x$13 able to track in 3D a cloud of points in the defined workspace. The festo's Robotino is an omnidirectional mobile platform with three drive units (three omni-wheels). How-

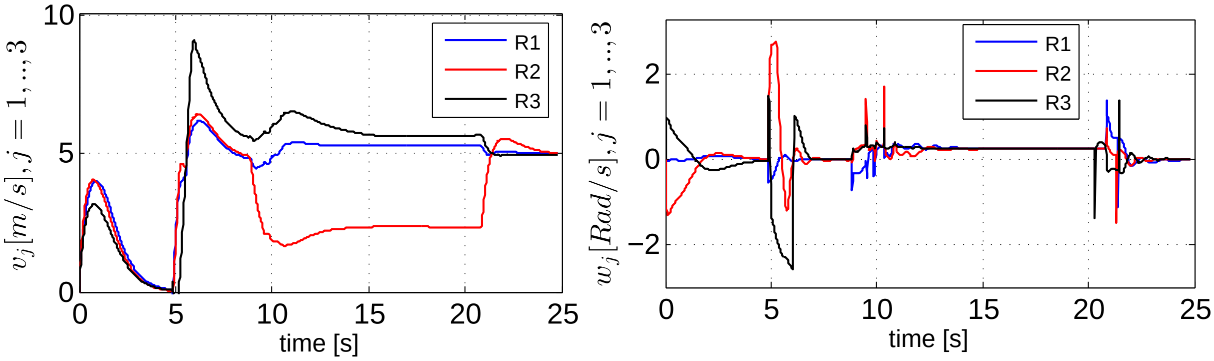


Figure 2.15: Actual linear and angular velocities in Case-2 using (2.24), (2.31)-(2.33) and (2.45) controllers.

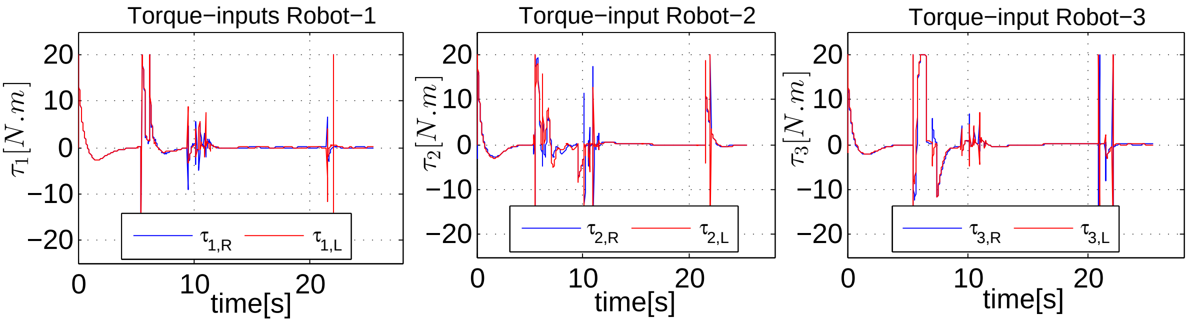


Figure 2.16: Applied torque control inputs (right and left wheel), in Case-2 using (2.24), (2.31)-(2.33) and (2.45) controllers.

ever, in this experiment, the Robotino will be controlled as if it is a non-holonomic mobile robot. Thus, a transformation between the three wheels angular velocities $(\dot{\varphi}_1, \dot{\varphi}_2, \dot{\varphi}_3)$ and the platform longitudinal and rotational velocities is used (2.53). As a result, the Robotino will be controlled by its longitudinal and rotational velocities $v_x$, $w$ respectively, ignoring its lateral velocity $v_y = 0$ [65].

$$\begin{bmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \end{bmatrix} = \frac{1}{R_w} \begin{bmatrix} -\frac{2\cos(\pi/6)}{3} & 0 & \frac{2\cos(\pi/6)}{3} \\ \frac{2\sin(\pi/6)}{3} & -\frac{2}{3} & \frac{2\sin(\pi/6)}{3} \\ \frac{1}{3L} & \frac{1}{3L} & \frac{1}{3L} \end{bmatrix}^{-1} \begin{bmatrix} v_x \\ v_y \\ w \end{bmatrix} \tag{2.53}$$

In (2.53), $R_w$ is the identical wheels radius and $L$ is the distance between the Robotino mass center and the wheel center[65].

The main control program runs on a ground station (PC), which communicates with the three robots Robotino via a Wi-Fi link (Fig. 2.18). Where, an external Router

Table 2.2: Comparison between the control performance obtained in case-1 and case-2 (TVF tracking task).

| | $Case - 1$ | | | $Case - 2$ | | |
|---|---|---|---|---|---|---|
| | $R1$ | $R2$ | $R3$ | $R1$ | $R2$ | $R3$ |
| $e_j^x(m)$ overshot | 0.51 | 0.76 | 2.62 | 0.32 | 0.55 | 1.23 |
| $e_j^y(m)$ overshot | 0.18 | 0.93 | 0.31 | -0.11 | -0.05 | 0.23 |
| Mean$\left[\left\|e_j^x(t)\right\|\right]$(m) | 0.24 | 0.33 | 0.54 | 0.14 | 0.11 | 0.16 |
| Mean$\left[\left\|e_j^x(t)\right\|\right]$(m) | 0.24 | 0.33 | 0.54 | 0.14 | 0.11 | 0.16 |

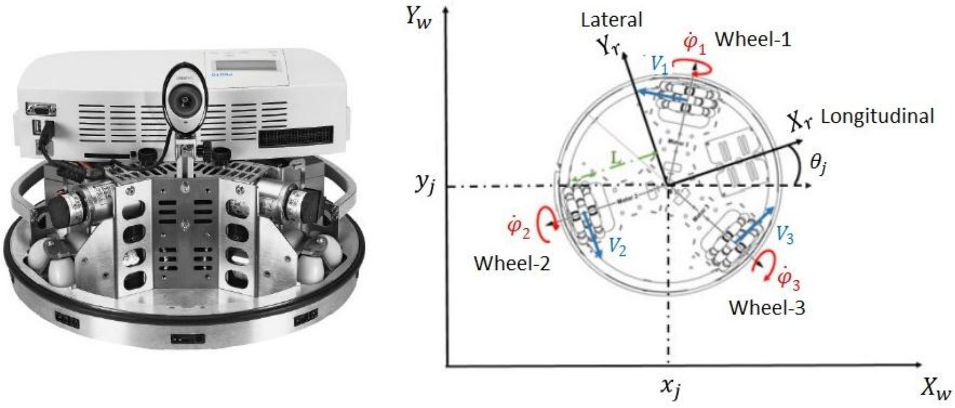


Figure 2.17: Front view of Robotino with its geometry plan.

is configured as a point access, that initiates a Wi-Fi network. Each of the Robotinto robots is equipped with an embedded PC and a WLAN interface set with a unique IP address, which is configured to get integrating to the Wi-Fi initiated by the Router in mode (Clients). In the other side, the ground control PC (central unit which runs the control algorithms) is configured to connect to the initiated router Wi-Fi. Then, specific functions of a Wi-Fi-library within C++ environment are used to handle the communication network, sensory data, and the control inputs sent from the ground PC to the three Robotino robots. The state information of the three Robotino is obtained from the Optitrack processing unit and sent to the ground control PC via an Ethernet port. In the case of poor communication, some UAV-UGVs connection links may get sometimes broken along time, which may result in formation tracking degradation, due to the fact of not receiving the leader's information (i.e., state, dynamics and control inputs). As a solution, the UAV-UGVs formation tracking may be explored under the constraint of partial access to UAV-leader's information (i.e., only some UGVs have access to UAV's information).

In the experiment, we used a virtual leader instead of a physical quadrotor due to the limitation of the flight space and the requirement to perform the experiments in the Optirack visual range, where its reference trajectory is predefined and shared among the formation robots.

In the first experience, deployment task control using (2.24) is carried out using three Robotino robots, where the initial desired EFDs vectors are given,

$$x_L(t=0) = 0.5; y_L(t=0) = 1.5; n_h = 1.$$

$$\tilde{A}_1 = [0.75]\,; \tilde{B}_1 = [0]\,; \tilde{C}_1 = [0]\,; \tilde{D}_1 = [0.75]$$

$$K_x = 1.5; K_y = 1.5; K_\theta = 0.3; K_v = 0.9; \lambda_1 = 0.2; M_s = 70; dt = 0.1s$$

Fig. 2.19 depicts the successful convergence process of the three robots Robotino towards the 2D initial desired curve starting from their initial position. Fig. 2.20 shows the evolution of the IPF of the three Robotino robots, where it can be seen that they decrease to zero with some weak oscillations which confirm the convergence process of the mobile robots. In Fig. 2.21, we see the tracking errors along the $x, y$ axes and the heading errors. Some oscillations are observed at the beginning of the convergence while the error magnitude is bounded.

In the second experiment, three Robotino robots are used to validate the second task (TVF tracking). Once the deployment task is realized, the three robots will track a reference trajectory while maintaining a TVF shape. The leader's reference trajectory is given as in Table 2.3,

Table 2.3: Reference trajectory of the virtual leader.

| | $t \in [0, 10s]$ | $t \in [10, 20s]$ |
|---|---|---|
| $x_L(m)$ | $0.1t - 1$ | $0.85 + 1.92cos(0.1\pi t)$ |
| $y_{q,r}(m)$ | $20$ | $30 + 15sin(0.26t - 2.61)$ |
| $\psi_L(Rad)$ | $0$ | $0.1(t-10) - \pi/6$ |

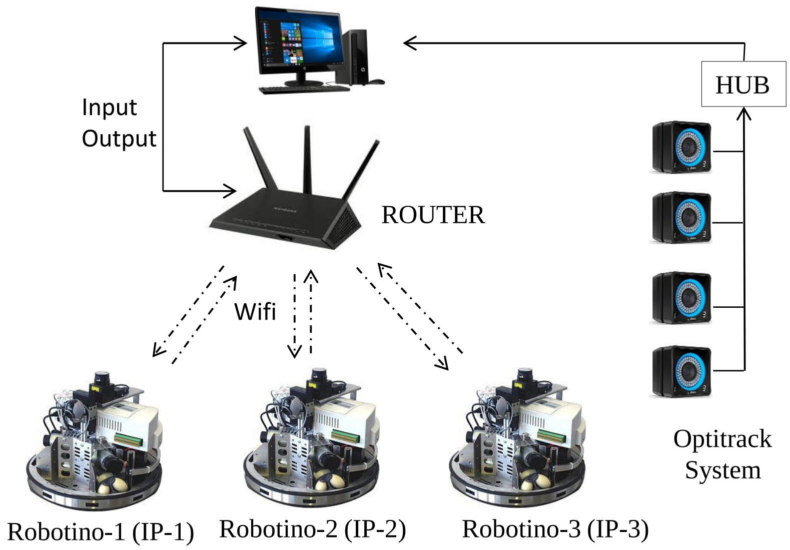


Figure 2.18: The experimental platform.

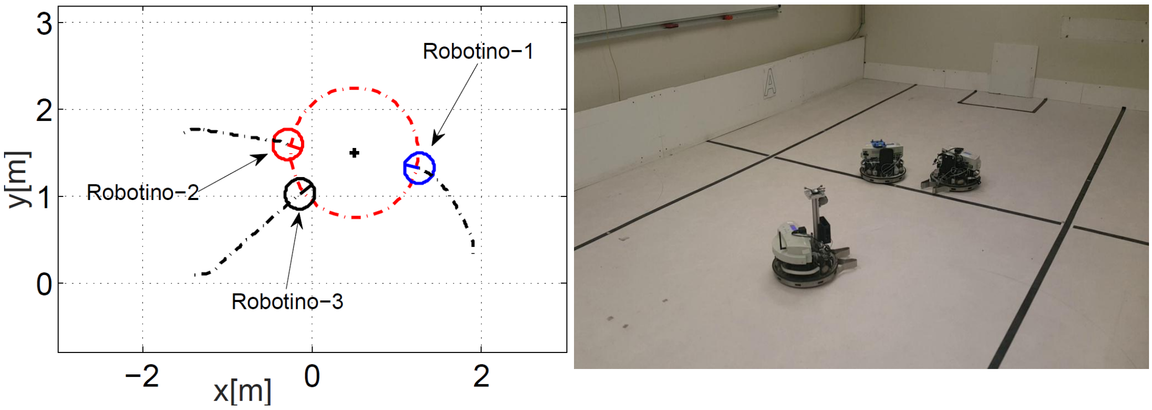


Figure 2.19: Convergence process of three Robotino robots (experiment-1).

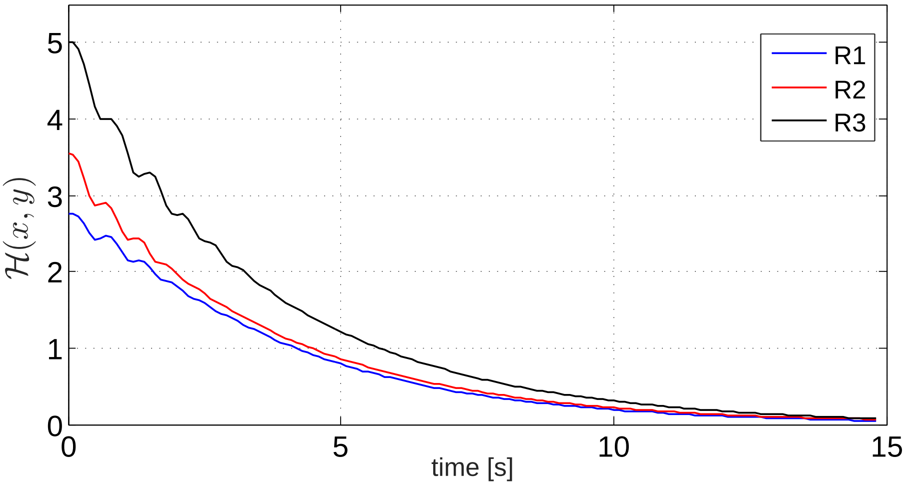


Figure 2.20: Evolution of the IPF of the three Robotino during the deployment task (experiment-1).

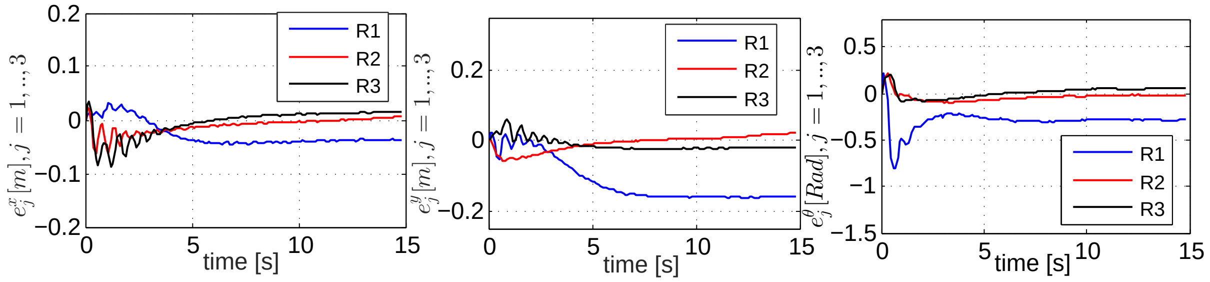


Figure 2.21: Tracking errors of the three Robotino robots in experiment-1 (i.e., deployment task)

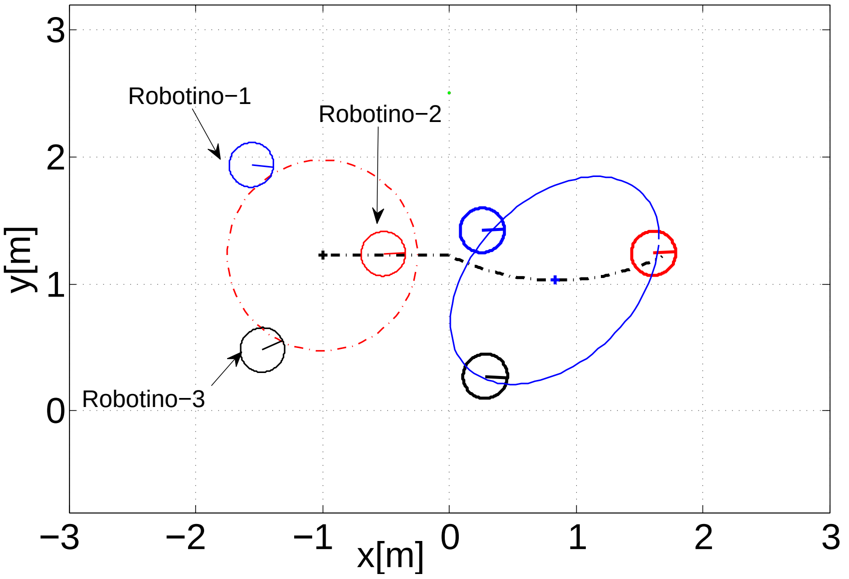


Figure 2.22: TVF tracking by three Robotino robots (experiment-2).

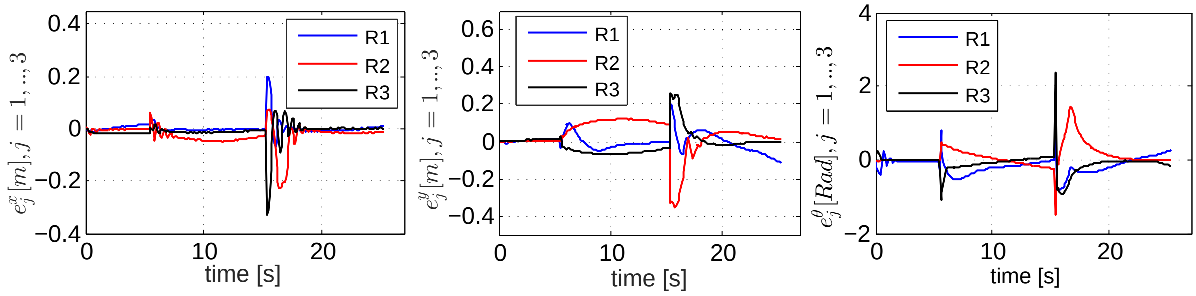


Figure 2.23: Tracking errors of the three Robotino robots in experiment-2.

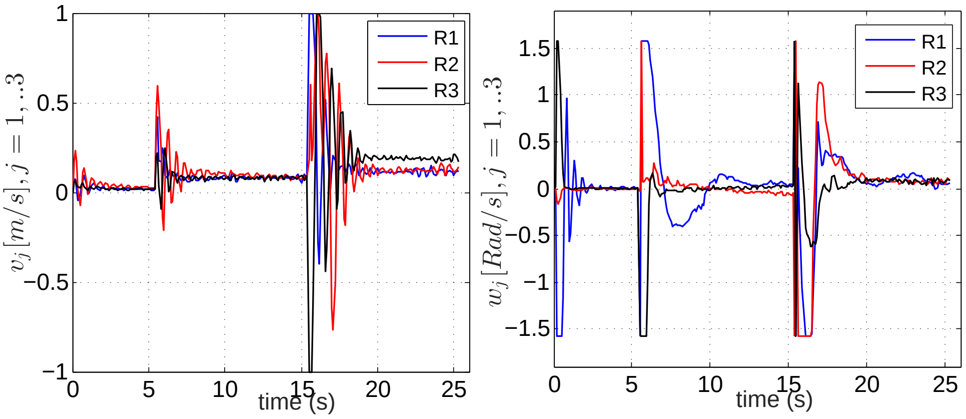


Figure 2.24: Velocity control-inputs of the three Robotino robots (experiment-2).

The used parameters, as well the EFDs vectors are given,

$$\tilde{A}_2\,(t) = \tilde{A}_1 - 0.2\sin(0.1\pi t); \tilde{B}_2(t) = \tilde{B}_2; \tilde{C}_2\,(t) = \tilde{C}_1; \tilde{D}_2\,(t) = \tilde{D}_1 + 0.3\cos(0.1\pi t);$$
$$K_x = 8; K_y = 11; K_\theta = 4; K_v = 0.8; dt = 0.1s.$$

Fig. 2.23 shows the tracking errors along the $x$ and $y$ axes as well as the heading errors, where we can notice that errors magnitude is weak and limited, however, at $t = 15.2s$ we observe some errors increases due to the change of the heading reference trajectory. Finally, in Fig. 2.24, the actual velocities control signals of the three Robotino robots are shown, in which weak chattering effect is visible on the linear velocities signals $v_j[m/s]; j = [1,3]$. This can bee explained by a backward and forward movements of the robots, due to the difficulty on choosing the suitable control parameters $K_x$ and $K_y$ during the experiment

## 2.8 Conclusion

In this chapter, a UAV-UGVs control system is considered. In which two novel nonlinear controllers have been proposed for controlling a UGVs system led by a UAV. Firstly, in deployment-task control, the key feature is to use the parametric representation of the desired formation shape (i.e., EFDs vectors) and its corresponding IPF. Thus, the formation shape dynamics are considered. The controller exhibits flexibility, scalability and offers smooth convergence to any free formation shape. Secondly, in the TVF tracking control, a combined robust velocity/torque controller based Backstepping is proposed, for which the UGVs-formation flexibility and robustness have been improved face modeling errors and external disturbance. The key idea is to introduce a virtual auxiliary control input $\underline{u}_j$ through a nonlinear feedback, and further adding a switching term function of the upper bound of the disturbance to guarantee the mobile robots velocity tracking. Lyapunov theory is used to show the stability and the boundedness of the UGVs formation tracking errors when using torque control-input. The effectiveness of the proposed controllers was demonstrated through numerical simulation, in which the importance of inserting an inner velocity-tracking loop is illustrated in the presence of modeling errors and external disturbance. The simulation results revealed good performance in terms of smooth convergence towards the desired TVF shape, accurate tracking, and further in terms of robustness against disturbance. The UGVs-formation control using the proposed pseudo velocity-control $\underline{v}_{jc}$ is further validated experimentally using three robots type festo's Robotino$^{(\mathbb{R})}$. In the next chapter, we extend the EFDs tool to model and control aerial-vehicles formation with application to quadrotors.

# CHAPTER 3

# QUADROTORS FORMATION CONTROL BASED 3-DIMENTIONAL DYNAMIC EFDS

## Contents

## 3.1 Introduction

In this chapter, we introduce an extension of the approach presented in [31, 49], where we propose a formation cooperative controller for a group of UAVs. We expanded the 2D−EFDs formation model introduced in Chapter-II to model 3D formation shapes. It is naturally to start first by designing the formation controller for holonomic robots, where the kinematics and the complexity relating to the dynamical models are relaxed. After that, we expand the developed controller to a group of UAV (i.e., quadrotors). The designed formation controller aims to track 3D planar free-form formation shapes. The UAVs formation is supposed to reach a predefined target and eventually to switch gradually to new time-varying formation shapes. During this process, the robots are coordinating among themselves by keeping a desired distance between each other and ensure an appropriate dispatching in the formation pattern.

The chapter is organized as follows, the next section will be devoted for introducing the concept of the EFDs tool and the IPF for 3D parametrized curves. In section−3, we explain the formation deployment control design for holonomics robots followed by an extension to UAVs formation deployment control in section-4. Then, in section−5, we showcase some simulation results to confirm the feasibility and the effectiveness of the proposed controller.

## 3.2 Problem formulation

### 3.2.1 Dynamic 3D EFDs and its correspondent Implicit Function

The EFDs are an alternative representation of Fourier descriptors using elliptical description of the curves. Thus, the EFDs model uses planar free-form curves as a sum of ellipses. EFDs are rotation, scale and translation invariant. The EFDs were used to represent any time-invariant $2D$ planar closed curves [30, 55] (see section-2.2.1). Similarly, from (2.1) we propose in the following to expand the $2D$ representation to a dynamic $3D$ one as,

$$\begin{cases} x\left(\delta,t\right) = a_0(t) + \sum\limits_{k=1}^{n_h} \left(A_k(t)\cos\left(k\delta\right) + B_k(t)\sin\left(k\delta\right)\right) \\ y\left(\delta,t\right) = c_0(t) + \sum\limits_{k=1}^{n_h} \left(C_k(t)\cos\left(k\delta\right) + D_k(t)\sin\left(k\delta\right)\right) \\ z\left(\delta,t\right) = e_0(t) + \sum\limits_{k=1}^{n_h} \left(E_k(t)\cos\left(k\delta\right) + F_k(t)\sin\left(k\delta\right)\right) \end{cases} \tag{3.1}$$

where $k$, $n_h$ and $\delta$ are already defined in (2.1). $x\left(\delta\right)$, $y\left(\delta\right)$ and $z\left(\delta\right)$ are the Cartesian coordinates of the points belonging to the curve, written as function of a normalized parameter $\delta$, with $\delta \in [0, 2\pi]$. The point $[a_0(t), c_0(t), e_0(t)]^T$ is the curve center and $A = [A_1, \ldots, A_{n_h}]^T \in \mathbb{R}^{n_h \times 1}$, $B(t) \in \mathbb{R}^{n_h \times 1}$, $C(t) \in \mathbb{R}^{n_h \times 1}$, $D(t) \in \mathbb{R}^{n_h \times 1}$, $E(t) \in \mathbb{R}^{n_h \times 1}$ and $F(t) \in \mathbb{R}^{n_h \times 1}$ are denoted the $3D$ curve descriptors (i.e. EFDs descriptors).

**Remark 3.1** *Notice that the representation (3.1) can be used in Leader-Follower based MVS control approach, where the leader's coordinates are the curve center $[a_0, c_0, e_0]^T$ and the followers state-offset with respect to the leader's position is given by the second term of $x(\delta, t)$, $y(\delta, t)$ and $z(\delta, t)$ in (3.1). Furthermore, it is important to note that the curve shape generated by (3.1) can be either a planar curve or a non-planar one. By using an arbitrary EFDs vectors values, we usually obtain a non-planar curve. To get a free-form 3D planar curve, at first we need to describe our desired curve in $2D$ frame using the $2D$ version of EFDs (2.1), then by using a predefined matrix transformation $T$ (see Fig. 3.1), i.e., including a rotation and translation, we can describe this latter on the $3D$ frame,*

$$\begin{pmatrix} \tilde{x}(\delta,t) \\ \tilde{y}(\delta,t) \\ \tilde{z}(\delta,t) \\ 1 \end{pmatrix} = T. \begin{pmatrix} x(\delta,t) \\ y(\delta,t) \\ 0 \\ 1 \end{pmatrix}. \tag{3.2}$$

*where $\tilde{x}(\delta, t)$, $\tilde{y}(\delta, t)$ and $\tilde{z}(\delta, t)$ are the $3D$ planar curve points coordinates. The parameters of the latter 3D curve denoted $\tilde{a}_0$, $\tilde{c}_0$, $\tilde{e}_0$, $\tilde{A}$, $\tilde{B}$, $\tilde{C}$, $\tilde{D}$, $\tilde{E}$ and $\tilde{F}$ are deduced from (3.2) by identification with (3.1).*

The EFDs is a very interesting mathematical tool to form and to model with high flexibility any dynamical planar 3D-curve. 3D-planar curves can also be represented

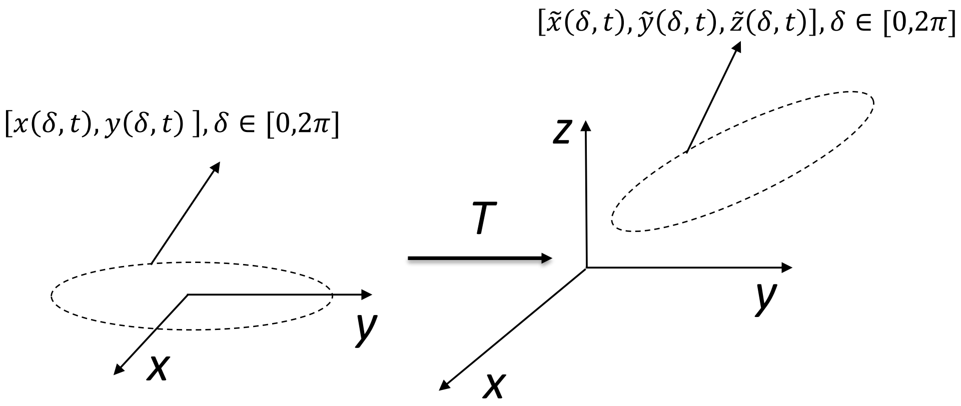

$[\tilde{x}(\delta,t),\tilde{y}(\delta,t),\tilde{z}(\delta,t)],\delta\in[0,2\pi]$
$[x(\delta,t),y(\delta,t)],\delta\in[0,2\pi]$
z
T
y
x
y
x

Figure 3.1: Constructing a 3D parametrized planar curve.

using Implicit Polynomial Function $IPF$ denoted by $\mathcal{H}(x,y,z)$. The $IPF$ function of a $3D$ planar curve is used to define an algebraic distance between any point in the space and the curve surface. It can be used to find whether a point is inside or outside the planar curve. This $IPF$ function can be found by implicitization of the EFDs. In this study, the $IPF$ function of a $2D$ closed planar curve is obtained using the method detailed in [1]. Similar to (2.5), the description of a $3D$ parametrized planar curve with an $IPF$ function takes the following form,

$$\mathcal{H}(x,y,z)=\sum_{0<i+j+k<d_p} a_{ijk}x^iy^jz^k=0 \tag{3.3}$$

where $a_{ijk}$ are real coefficients and $d_p=2n_h$ is the degree of the polynomial function, as detailed in [1].

The EFDs of a planar closed 3D-curve can be estimated similar to the method explained in section-2.2.1 using an extended version of (2.2), [31,56].

**Example of EFDs estimation:** Refer to Fig. 3.2, relying on 20 3D-coordinate points that describe geometrically a 3D planar curve in blue color, the 3D−EFDs are estimated using extended version of (2.2) with a selected number of harmonics of $n_h=2$. The estimated EFDs are $a_0=61.9$, $c_0=66.3$, $e_0=20.1$, $A=[7.2,-0.1]^T$, $B=[-19.3,3.1]^T$, $C=[17.2,0.4]^T$, $D=[1.8,0.1]^T$, $E=[1.1,0]^T$ and $F=[0.4,0.1]^T$. The estimated EFDs-vectors are then used to reconstruct the 3D curve in red color using (3.1). The $IPF$ function of the parametrized curve is too long to be carried on the thesis document.

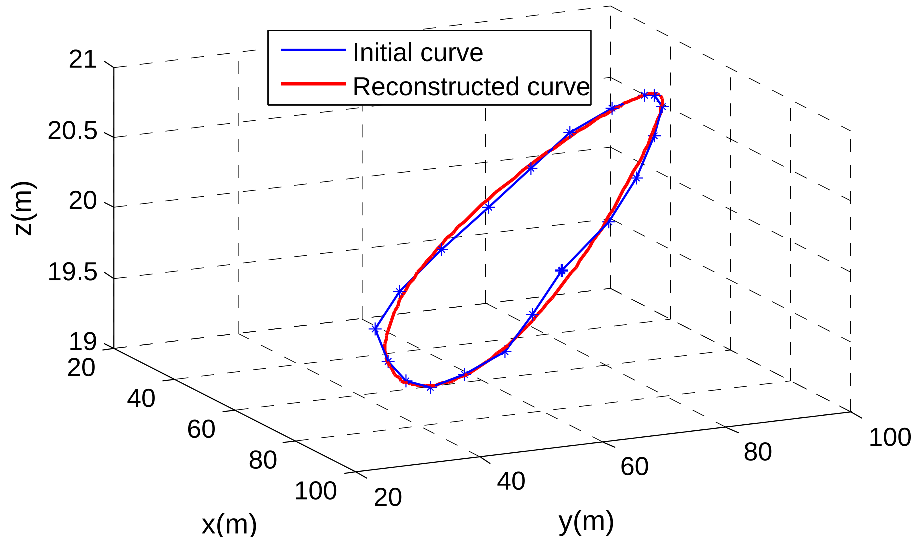


Figure 3.2: Example of EFDs-vectors estimation based on a set of points coordinates.

### 3.2.2 Holonomic robot model:

In this section, we consider the robot as a $3D$ mass point particle to rend the robots-formation control problem more simplified. This latter assumption allows eliminating the complexity related to the robots dynamics and its non-holonomic constraints (i.e., UAVs) from the analysis, and enables studying the efficiency and the feasibility of the proposed formation control approach. We first present the robot kinematic model and then we detail in the next section the formation control strategy. The $i_{th}$ robot in the formation is featured by a first order integrator as follows,

$$\begin{cases} \dot{x}_i = u_{x,i} \\ \dot{y}_i = u_{y,i} \\ \dot{z}_i = u_{z,i} \end{cases} \tag{3.4}$$

where $\chi_i = (x_i, y_i, z_i)^T$, $i \in [1, .., N_R]$ and $\dot{x}_i, \dot{y}_i, \dot{z}_i$ are the cartesian coordinates and the their time-derivatives (i.e., velocities) of the $i_{th}$ robot respectively, with respect to the world coordinate frame. $\underline{u}_i = [u_{x,i}, u_{y,i}, u_{z,i}]^T$ is its corresponding control input. It is straightforward to see from (3.4) that the robot dynamic exhibits holonomic property.

## 3.3 Formation control based IPFs and EFDs

We focus in this section on expanding the formation controller proposed in [30] to a 3D robots formation control. In [30], the formation controller is valid only for 2D robots to achieve deployment process towards 2D planar curves.

Initially, we assume that $N_R$ $3D$-holonomic robots group featured by dynamics as in (3.4) are randomly positioned in the defined space. The proposed robots formation control aims to converge the robots group towards the contour of a desired 3D planar curve (i.e., geometric configuration) relying on the knowledge of the $IPF$ that features the desired 3D-curve. During this process, the robots are expected to avoid collision between each other and to keep a certain coordination in the formation. Then, the control input $\underline{u}_i$ for the $i_{th}$ robot will be obtained by the sum of the sub-control component designed for the deployment control denoted by $\underline{u}_i^D$ and coordination control input $\underline{u}_i^C$ respectively,

$$\underline{u}_i = \begin{pmatrix} u_{x,i} \\ u_{y,i} \\ u_{z,i} \end{pmatrix} = \underline{u}_i^D + \underline{u}_i^C; i \in [1, N_R] \tag{3.5}$$

We can define a $3D$ desired formation shape for a group of $3D$ robots by many ways depending on the task to be realized. It could be the perimeter of a friend $UAV$ or an enemy target to be encircled or supervised or the boundary of a considered geographical area. This boundary (i.e., desired $3D$ curve) can be defined based on an extracted and processed images provided by a UAV equipped by an onboard camera for instance. The desired $3D$ formation shape (i.e., 3D curve) is in practice a sequence of points coordinate. We use (2.2) to estimate its EFDs and the method detailed in [1] to compute its $IPF$ function.

### 3.3.1 Formation deployment control

Firstly, we focus on designing a formation controller that ensures moving the robots to the desired 3D planar curve. As depicted in Fig. 3.3, the formation deployment sub-control input $\underline{u}_i^D$ is composed of two components, parallel input $\underline{u}_i^{DP}$ and normal

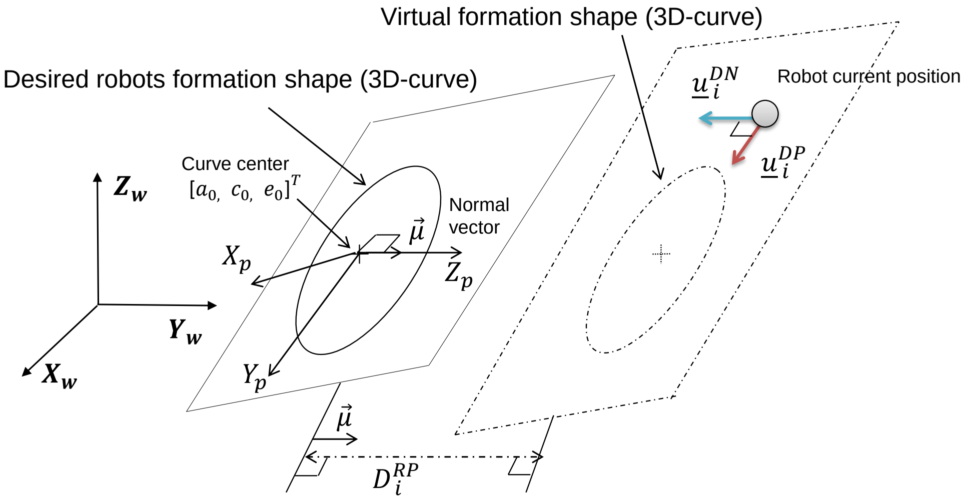


Figure 3.3: Illustrative diagram of the robots formation deployment control [2].

input $\underline{u}_i^{DN}$ respectively,

$$\underline{u}_i^{D} = \underline{u}_i^{DP} + \underline{u}_i^{DN} \tag{3.6}$$

The first control-input component $\underline{u}_i^{DP}$ of (3.6) will be designed based on the framework of [30] by using the *IPF* of the desired formation shape (i.e., desired planar curve) as a potential function. This control input will ensure only the robot convergence to the virtual 3D−curve which is parallel to the desired 3D−curve as it is shown in Fig. 3.3.

The sub-control input $\underline{u}_i^{D}$ will be designed in sort to minimize the following potential function [2],

$$\mathcal{F}\left(x,y,z\right) = \mathcal{H}\left(x,y,z\right) + \frac{1}{2}\lambda_2\left\|D_i^{RP}\vec{\mu}\right\|^2 \tag{3.7}$$

where $\mathcal{H}\left(x,y,z\right)$ is the IPF, $\lambda_2$ is a positive constant number and $D_i^{RP}$ is the normal distance between the $i_{th}$ robot position and the plane that holds the desired 3D−curve (see Fig. 3.3). $\vec{\mu}$ is unitary vector that is normal to the surface holding the desired 3D-curve.

The IPF function $\mathcal{H}\left(x,y,z\right)$ is adapted to find whether a point is on a curve or not. It could be used to measure the distance between a point and a closed curve [30, 31]. The position error function between the $i_{th}$ robot and the 3D **virtual** planar curve is given by the algebraic distance to the curve of the desired formation shape as,

$$e_i^{virt} = \mathcal{H}\left(x_i, y_i, z_i\right) \tag{3.8}$$

By using (3.8), if the robot is approximately on the virtual curve, the error is zero, while it is negative when the robot is inside the virtual curve and positive when outside. The sub-control input $\underline{u}_i^{DP}$ is designed to force this error to decrease exponentially based on Lyaponov theory, i.e.

$$\dot{\mathcal{H}}\left(x_i, y_i, z_i\right) = -\lambda_1 \mathcal{H}\left(x_i, y_i, z_i\right) \tag{3.9}$$

where $\lambda_1$ is a positive constant number. Using chain rule of differentiation and substituting (3.4) in (3.9), we find,

$$\begin{pmatrix} \mathcal{H}_x & \mathcal{H}_y & \mathcal{H}_z \end{pmatrix} \begin{pmatrix} u_{x,i}^{DP} \\ u_{y,i}^{DP} \\ u_{z,i}^{DP} \end{pmatrix} = -\lambda_1 \mathcal{H}(x_i, y_i, z_i) \tag{3.10}$$

The parallel formation deployment sub-control input $\underline{u}_i^{DP}$ for the $i_{th}$ robot can be determined using pseudo inverse operator, namely

$$\begin{pmatrix} u_{x,i}^{DP} \\ u_{y,i}^{DP} \\ u_{z,i}^{DP} \end{pmatrix} = -\lambda_1 \frac{1}{\left\|\nabla_{\underline{x}} \mathcal{H}\left(x_i, y_i, z_i\right)\right\|^2} \mathcal{H}(x_j, y_j, z_j) \begin{pmatrix} \mathcal{H}_x(x_i, y_i, z_i) \\ \mathcal{H}_y(x_i, y_i, z_i) \\ \mathcal{H}_z(x_i, y_i, z_i) \end{pmatrix} \tag{3.11}$$

where $\mathcal{H}_x$, $\mathcal{H}_y$ and $\mathcal{H}_z$ are the partial derivatives of $\mathcal{H}\left(x, y, z\right)$ with respect to $x$, $y$ and $z$, and we have $\nabla_{\underline{x}} \mathcal{H} = \left[\begin{array}{ccc} \mathcal{H}_x & \mathcal{H}_y & \mathcal{H}_z \end{array}\right]^T$.

Now, back to (3.7), it is straightforward to derive the sub-control input $\underline{u}_i^{DN}$ that ensures moving the robots normally to the plane that contains the desired 3D curve as [2],

$$\underline{u}_i^{DN} = -\nabla_{\underline{x}} \left(\frac{1}{2} \lambda_2 \left\| D_i^{RP} \vec{\mu} \right\|^2\right) = -\lambda_2 D_i^{RP} \vec{\mu} \tag{3.12}$$

where $\lambda_2$, $D_i^{RP}$ are already defined in (3.7), $\vec{\mu} = R_w^{plane} \underline{e}_z$ with $\underline{e}_z = [0, 0, 1]^T$ is a unitary vector expressed on word coordinate frame and $R_w^{plane}$ is a rotation matrix relating the world coordinates frame to the plane that contains the $3D$ desired curve (see Fig. 3.3).

### 3.3.2 Formation coordination control

The robots are supposed to coordinate among themselves during the robots deployment-process. Therefore, the robots must avoid collision each to other and ensuring a given desired spacing between each robot and its neighbours, which enables also having a certain distribution of the robots group in the achieved formation shape, which is the control aim in this section. The coordination control input is designed as the sum of virtual forces of linear springs exercised from each robot to its two nearest neighbours as in Fig. 3.4.

The spring is proposed to have a normal length which is equal to the secure distance $D_s$. The produced force by the spring is linearly proportional to the difference between the actual distance and the secure distance between the robots. The spring produces a force directed from the $i_{th}$ robot to its two neighbours neighbours as follows,

$$\begin{pmatrix} u^C_{x,i} \\ u^C_{y,i} \\ u^C_{z,i} \end{pmatrix} = K_{rr}\left(D_s - D_{ij}\right)\begin{pmatrix} x_i - x_j \\ y_i - y_j \\ z_i - z_j \end{pmatrix} + K_{rr}(D_s - D_{ik})\begin{pmatrix} x_i - x_k \\ y_i - y_k \\ z_i - z_k \end{pmatrix} \tag{3.13}$$

where $K_{rr}$ is an adaptable spring constant. The desired coordination stiffness could be obtained by tuning $K_{rr}$. Therefore, $K_{rr}$ may take relatively grand values if the robots are far away from the desired configuration, while it takes small values when the robots are reaching the desired configuration. We have $j$ and $k$ the indices for the robots that are the nearest two neighbours of $i_{th}$ robot. $D_{ij}$ and $D_{ik}$ are the current

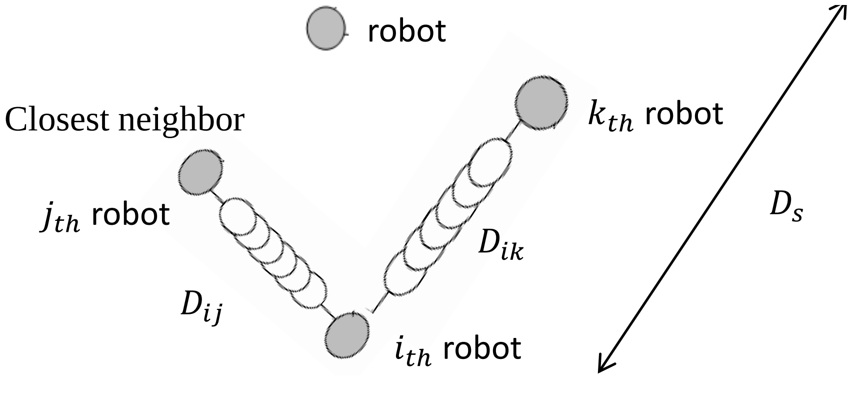


Figure 3.4: Modelling of the coordination control.

distances of the $i_{th}$ robot from the $j_{th}$ and $k_{th}$ robots respectively. $[x_j, y_j, z_j]^T$ and $[x_k, y_k, z_k]^T$ are the position coordinates of the robots $j$ and $k$ respectively with respect to the world coordinate frame. It is worth noting that is possible to set the number of nearest neighbours greater than 2.

## 3.4 Extension to quadrotors formation control

In this section, we will expand the developed formation controller (3.6) to a quadrotors formation. The dynamics of the quadrotor can be viewed as a double sub-dynamics in cascade, such the rotational dynamics (inner-loop) which is fast and the translational dynamics (outer-loop) which is slow.

The quadrotors formation control will be divided into two parts. The first control part will be applied to the translational dynamics to drive the quadrotors to a desired 3D-planar curve, while tracking the curve center $[a_0(t), c_0(t), e_0(t)]^T$ trajectory (viewed as a virtual leader). The second control part concerns the angles tracking, where the outputs of the first control-part will feed-forward the second control-part.

Recall that the control objective is to design a quadrotors formation-controller that converges each agent towards the contour of the desired $3D$ planar curve. The nonlinear translational dynamics of a the quadrotor are given as in (2.8), (2.9). We denote $\chi_i = [x_i, y_i, z_i]^T$ the position of the mass center of the quadrotor with respect to the inertial frame and $\eta_i = [\varphi_i, \theta_i, \psi_i]^T$ the vector of Euler angles. Let $u_{z,i}$ and $\tau_{q,i} = [u_{\varphi_i}, u_{\theta_i}, u_{\psi_i}]^T$ be the total thrust and the torque control vector respectively. For simplicity, we denote by $q_i = [\chi_i, \dot{\chi}_i, \eta_i, \dot{\eta}_i]^T$ the quadrotor state vector.

Since the equations in (2.8), (2.9) are nonlinear, the solution is difficult to be found in closed form because of trigonometric functions related each other in no-elementary way. For this reason, the linearization is performed on a simplified model called to *small oscillations* [66]. This simplification is made by approximating the sine function with its argument and the cosine function with unity. The approximation is valid if the argument is small. Then, we linearize the translational dynamics around the equilibrium point $\bar{q}_i = [\bar{\chi}_i, 0_{3\times1}, \bar{\eta}_i, 0_{3\times1}]^T$,

$$\begin{cases} \ddot{x}_i = \frac{1}{M_q}\theta_i \\ \ddot{y}_i = -\frac{1}{M_q}\phi_i \\ \ddot{z}_i = \frac{u_{z,i}}{M_q} \end{cases} \tag{3.14}$$

In (3.14), $M_q$ is the quadrotor mass. The the translational dynamics of the quadrotor is approached to a double integrator one, i.e., $\ddot{\chi} = \underline{u}_i$, in order to reduce the complexity of the formation controller design, however, the rotational dynamics remains as described in (2.9).

It is known that the quadrotor is driven along x-axis, y-axis and z-axis by changing the pitch and roll angles and the total thrust, thus, the control input is chosen to be $\underline{u}_i = [\frac{1}{M_q}\theta_i, -\frac{1}{M_q}\phi_i, \frac{u_{z,i}}{M_q}]^T$.

Similar to the followed steps resulting (3.11), the reference linear velocities of the $i_{th}$ quadrotor are given,

$$\begin{pmatrix} \dot{x}_i^r \\ \dot{y}_i^r \\ \dot{z}_i^r \end{pmatrix} = -\lambda_1 \frac{1}{\left\|\nabla_{\underline{x}}\mathcal{H}\left(x_i, y_i, z_i\right)\right\|^2}\mathcal{H}(x_j, y_j, z_j) \begin{pmatrix} \mathcal{H}_x(x_i, y_i, z_i) \\ \mathcal{H}_y(x_i, y_i, z_i) \\ \mathcal{H}_z(x_i, y_i, z_i) \end{pmatrix} \tag{3.15}$$

Then, the reference quadrotors trajectory is obtained by integrating (3.15). Further, $PD$-controller is used to track the reference linear velocities. Thus, the parallel deployment sub-control $\underline{u}_i^{DP}$ is given,

$$\underline{u}_i^{DP} = K_P\left(\begin{bmatrix} x_i^r \\ y_i^r \\ z_i^r \end{bmatrix} - \chi_i\right) + K_D\left(\begin{bmatrix} \dot{x}_i^r \\ \dot{y}_i^r \\ \dot{z}_i^r \end{bmatrix} - \dot{\chi}_i\right) \tag{3.16}$$

where $K_P$ and $K_D$ are selected positive gains. Now, the deployment control input $\underline{u}_i^D = \underline{u}_i^{DC} + \underline{u}_i^{DN}$, with $\underline{u}_i^{DN}$ is designed as in (3.12). Next, from (3.13), (3.14), and knowing that $\underline{u}_i = \underline{u}_i^D + \underline{u}_i^C = [\frac{1}{M_q}\theta_i, -\frac{1}{M_q}\phi_i, \frac{u_{z,i}}{M_q}]^T$, we can deduce the reference pitch and roll signals as,

$$[\theta_i^r, \phi_i^r]^T = M_q[u_{x,i}, -u_{y,i}]^T \tag{3.17}$$

where $u_{x,i}$ and $u_{y,i}$ are the first and the second components of $\underline{u}_i$. Notice that the yaw angle $\psi_i$ is not considered in this study, since it doesn't affect the translational

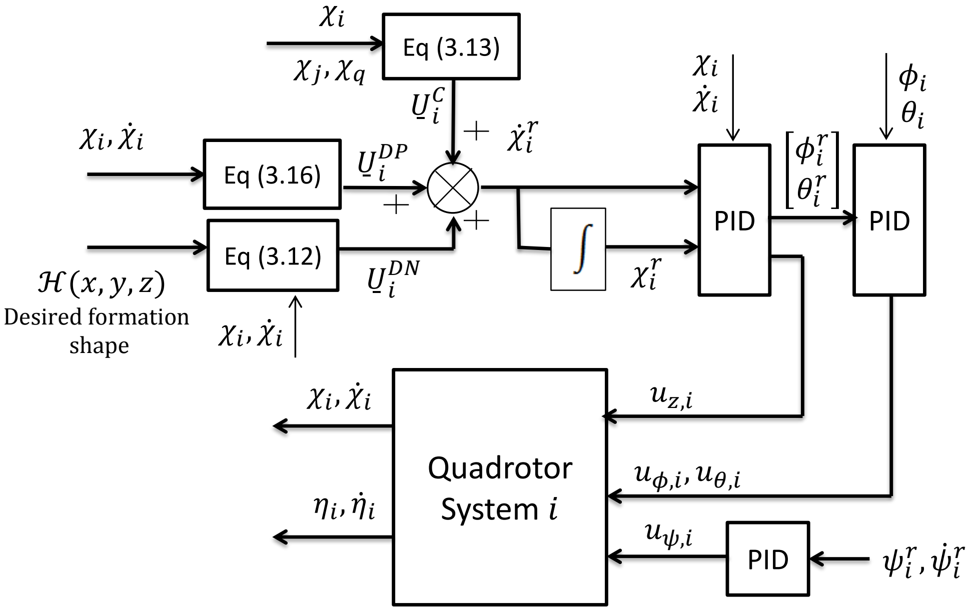


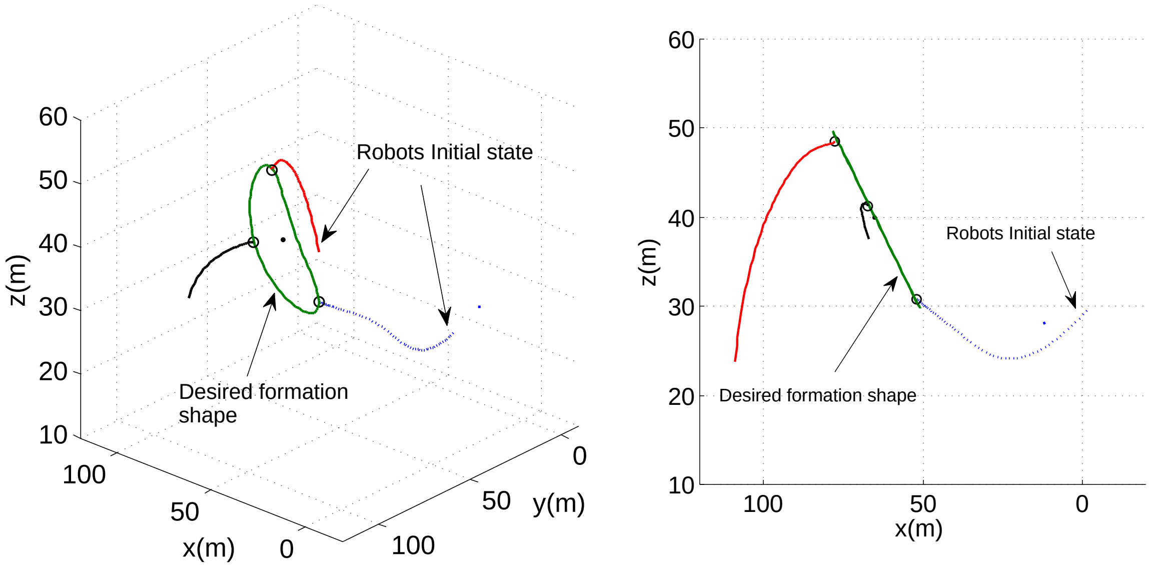
Figure 3.5: Deployment control diagram of the quadrotors formation.

displacement of the quadrotors along the $x$ and $y$ axes. Thus, the yaw angle is assumed constant and will be controlled separately. Finally, the reference roll and pitch angles will be tracked through a PID controller to realize the deployment task by the quadrotors.

## 3.5 Simulation results

In this section, simulations results are provided to show the feasibility and the effectiveness of the proposed deployment-controller (3.11), (3.12) and (3.13). The simulation results could be obtained with any desired 3D planar formation (i.e., EFDs vectors) and any number of the robots. Two simulation examples are carried out in Matlab environment. In Example-1, the agents are 3D holonomics robots featured by first integrator model. The aim is to check the feasibility of the proposed formation deployment control and coordination without considering the agents dynamics. In Example-2, the proposed deployment controller is extended and applied to a group of quadrotors.

**Example-1:** In this example, three robots are employed to see the performances of the controller in formation deployment and coordination. The desired 3D$-$curve is



(a)

(b)

Figure 3.6: Deployment process towards 3D−curve by three robots using (3.11), (3.12), and (3.13).

described by the following EFDs vectors,

$$A = \begin{bmatrix} 16.21 & -0.89 \end{bmatrix}^T; B = \begin{bmatrix} 4.22 & 0.54 \end{bmatrix}^T; C \begin{bmatrix} -2.39 & -2.85 \end{bmatrix}^T;$$
$$D = \begin{bmatrix} 8.56 & -1.74 \end{bmatrix}^T; E = \begin{bmatrix} 1.38 & 1.65 \end{bmatrix}^T; F = \begin{bmatrix} -4.94 & 1 \end{bmatrix}^T$$

The parameters used in the first simulation example, $dt = 0.01$, $D_s = 25m$, $\lambda_1 = 1.4$ and $\lambda_2 = 1.1$. The adaptable parameter $K_{rr}$ used is,

$$K_{rr} = \begin{cases} 2, |\mathcal{H}(x_i, y_i, z_i)| < \mathcal{H}_0 \\ 6, |\mathcal{H}(x_i, y_i, z_i)| \geq \mathcal{H}_0 \end{cases}$$

where $\mathcal{H}_0 = 15$ is a threshold algebraic distance between the robot position and the contour of the desired formation.

In Fig. 3.6, we can see a scenario in which three robots placed randomly in the space, where the desired curve is represented using EFDs with $n_h = 2$ harmonics and an $IPF$ function with a degree $d_p = 4$. The robots under the control input (3.6) and (3.13) are correctly and successfully achieving the desired 3D-curve.

Fig. 3.7 and Fig. 3.8 show the different control inputs acting on the robot−2 and the corresponding IPF function of the three robots respectively. We see that the control input (3.11) is acting on the robot−2 (chosen arbitrary among the three robots) in sort

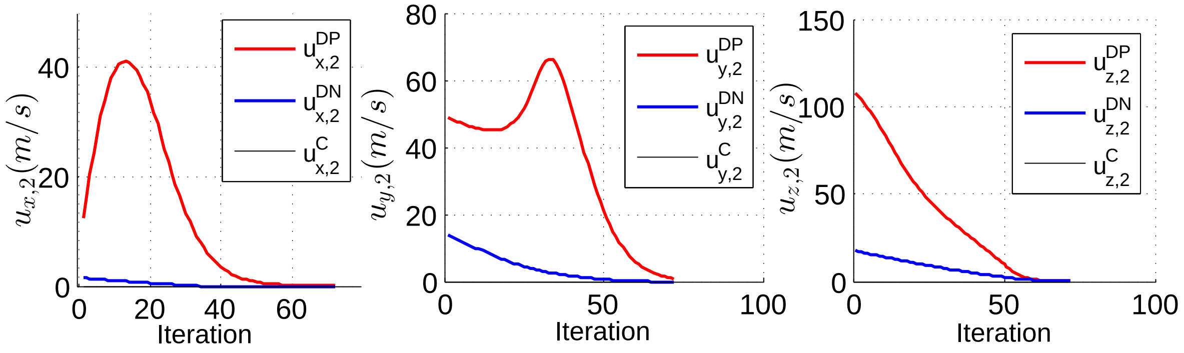


Figure 3.7: Control inputs of robot $i = 2$

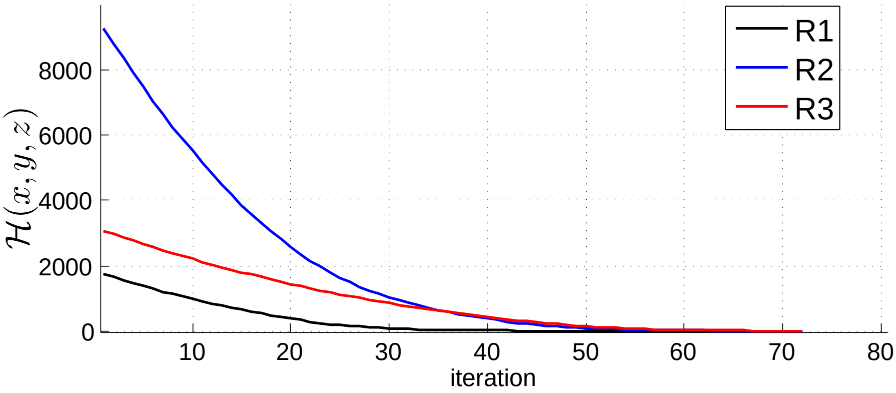


Figure 3.8: Variation of $\mathcal{H}(x,y,z)$ during the deployment precess ((Example-1)).

of decreasing its corresponding IPF function, which confirms the effectiveness and the conjecture of the proposed controller based on Lyapunov theory. We can remark, that the coordination control component is nil since the robots are too far each to other in the realized formation.

**Example-2:** In this example, five (05) quadrotors are considered to achieve deployment behaviour relying on the proposed formation deployment. The quadrotrs are featured by dynamical model described by (2.8) and (2.9). The quadrotors deployment control scheme is shown in Fig. 3.5. The desired 3D-planar curve is given by the following EFDs vectors,

$$A = \begin{bmatrix} 16.21 & 2.89 \end{bmatrix}^T; B = \begin{bmatrix} 5.22 & 0.54 \end{bmatrix}^T; C\begin{bmatrix} -2.39 & -2 \end{bmatrix}^T;$$
$$D = \begin{bmatrix} 11.16 & -1.74 \end{bmatrix}^T; E = \begin{bmatrix} -1.38 & -1.15 \end{bmatrix}^T; F = \begin{bmatrix} 6.44 & -1 \end{bmatrix}^T$$

The parameters used in Example-2 simulation are, $dt = 0.01s$, $\lambda_1 = 7$ and $\lambda_2 = 1.6$.

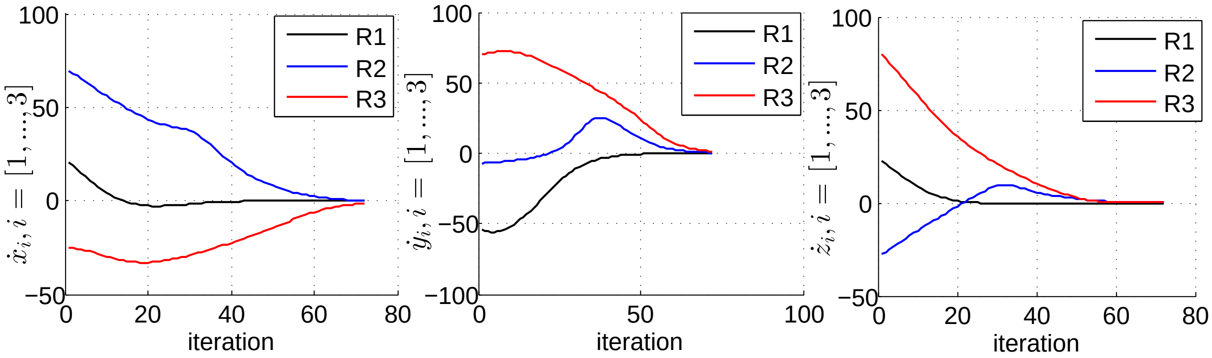


Figure 3.9: Robots velocity inputs $\dot{\chi}_i = [\dot{x}_i, \dot{y}_i, \dot{z}_i]^T$, $i \in \{1, 3\}$.

The adaptable parameter $k_{rr}$ is given,

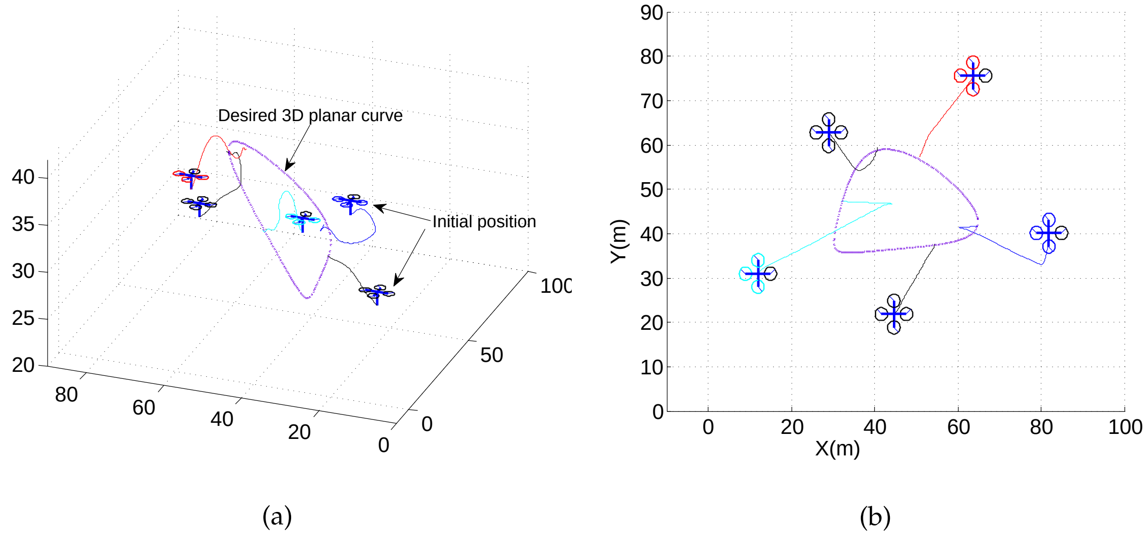
$$K_{rr} = \begin{cases} 2, |\mathcal{H}(x_i, y_i, z_i)| < \mathcal{H}_0 \\ 6, |\mathcal{H}(x_i, y_i, z_i)| \geq \mathcal{H}_0 \end{cases}$$

where $\mathcal{H}_0 = 200$. Further we set, $K_{rr} = 3$ and $D_s = 5m$ in (3.13). The used qudarotor's mechanical and electrical parameters in simulation were taken from [58]. The quadrotors initial positions and the Euler angles vectors given by, $\chi_i$, $\eta_i$, with $i \in [1, 5]$ are set manually using the matlab function *ginput*.

In Fig. 3.10, we show snapshots from different angle-view of the quadrotors during the the deployment precess towards the desired 3D-planar curve. It can be seen clearly that the quadrotors have successfully converged to the contour of the 3D-planar curve.

Fig. 3.11 and Fig. 3.12 depict the obtained IPF of the four quadrotors over time and the quadrotors tracking errors over time respectively. The decreasing behaviour of the IPF $\mathcal{H}(x, y, z)$ is well observed. In addition, the state tracking errors along the $x$, $y$ and $z$ axes followed by the Euler's tracking errors are shown to converge to a neighbourhood of the origin. These results confirm the conjecture of the proposed control theory.

Fig. 3.13 illustrates the virtual forces acting on quadrotor-3, namely $U_i^{DP}$, $U_i^{DN}$, and $U_i^{C}$, with $i = 3$. The sum of these forces yields the quadrotor's reference trajectory rate, i.e., $\dot{\chi}_i^r(t)$ leading to the planar 3D-curve as shown in Fig. 3.5. The quadrotrs control inputs $u_{z,i}$, $u_{\phi,i}$, $u_{\theta,i}$, $u_{\psi,i}$, $i \in [1, 4]$, are depicted in Fig. 3.14. These control in-



(a)

(b)

Figure 3.10: Snapshots of the quadorotors during the deployment process.

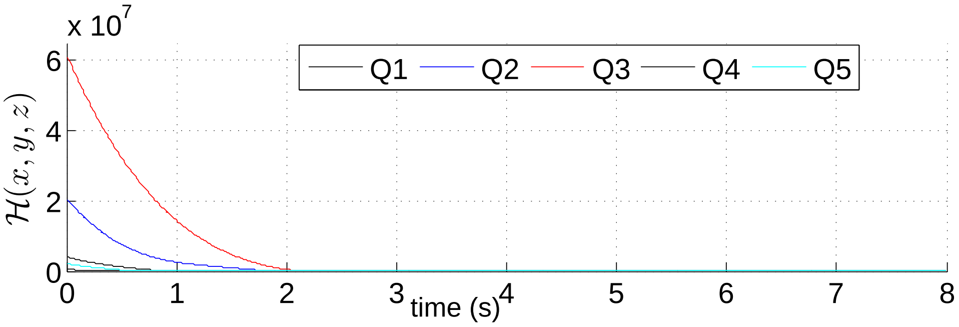


Figure 3.11: Variation of $\mathcal{H}(x,y,z)$ during the deployment process (Example-2).

puts are obtained through the PID control blocs. Despite the fact that the deployment controller was fist designed for first-integrator based dynamic systems, it is important to notice the feasibility as well as the validity of the proposed deployment controller on systems having such a complex, coupled and nonlinear dynamics, i.e. quadrotors.

## 3.6 Conclusion

We have presented in this chapter a novel formation control of 3D holonomic robots (i.e., deployment control) to track arbitrary desired time-varying 3D shapes (i.e., planar curves). We used an extension version of the 2D dynamic EFDs to model the time-varying formation into 3D planar curve enabling more flexibility on the choice

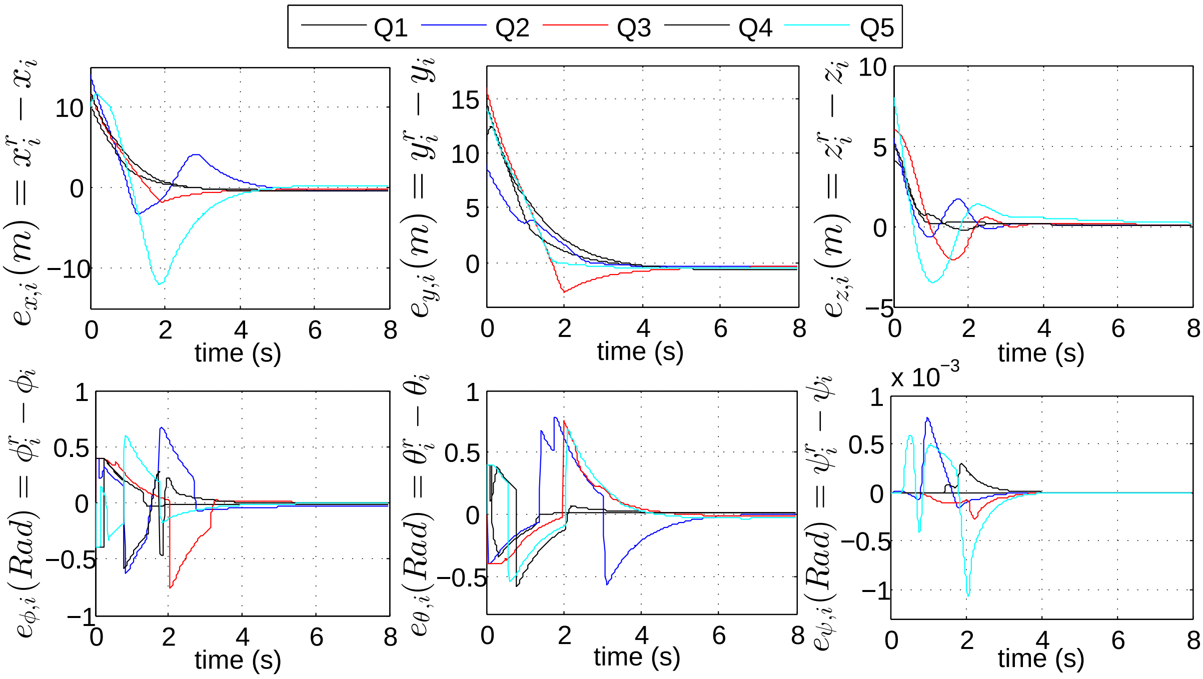


Figure 3.12: The quadrotors tracking errors, $\tilde{\chi}_i = \chi_i^r - \chi_i$, $\tilde{\eta}_i = \eta_i^r - \eta_i$, $i = [1,5]$ (Example-2).

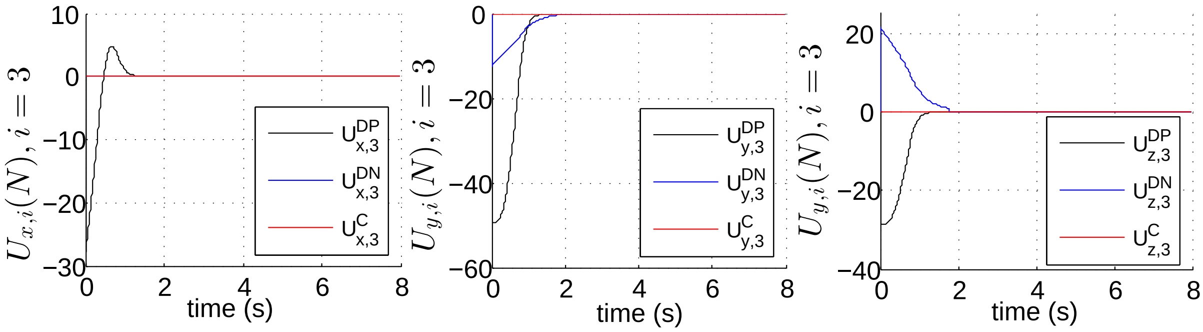


Figure 3.13: The control forces $U_i^{DP}$, $U_i^{DN}$, and $U_i^{C}$, of quadrotor $i = 3$ chosen arbitrary (Example-2).

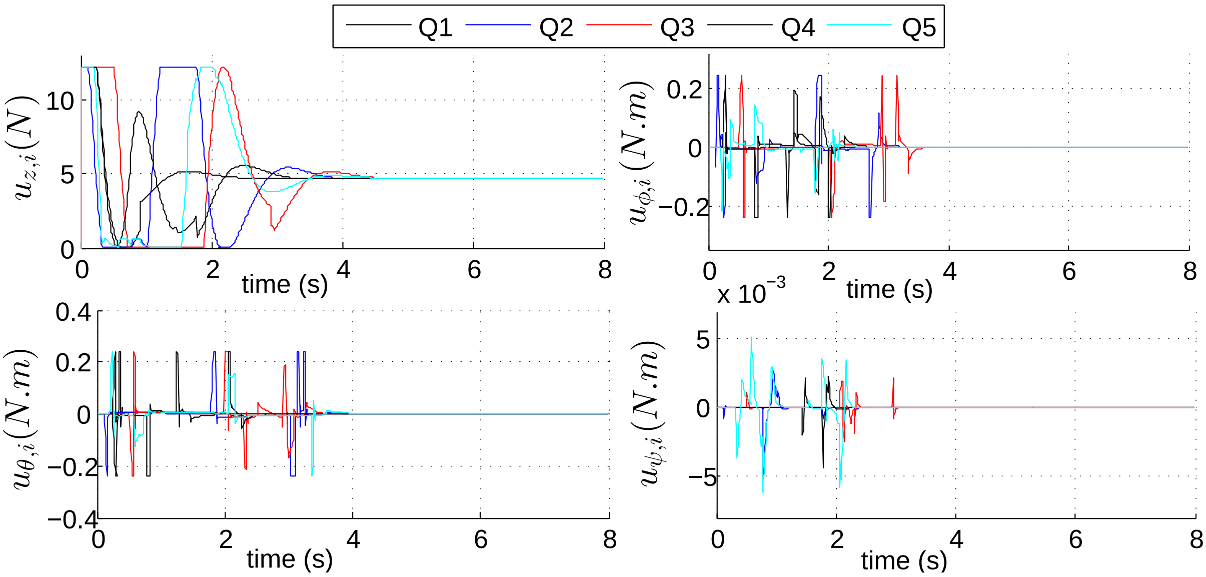


Figure 3.14: The quadrotors control inputs $u_{z,i}$, $u_{\phi,i}$, $u_{\theta,i}$, $u_{\psi,i}$, $i \in [1,5]$ (Example-2).

of the desired formation shape. The correspondent implicit function IPF was used as a potential function to design the deployment controller. Next, the developed controller was applied and extended to a quadrotors formation. In which, the formation controller is divided into two cascade parts. In the first part, the proposed deployment control was used to control the translational dynamics based on the linearized version of the quadrotor dynamical model, which yields the total thrust as well as the reference roll and pitch angles. The second control part is PID Control, which has the role of stabilizing the rotational dynamics. The obtained simulation results revealed the feasibility of the method and the stability of the whole system for first-integrator based dynamics systems, and then extend to second order nonlinear systems, i.e., quadrotors. In the next chapter, we will explore the distributed MASs formation tracking control relying on consensus tracking framework, in which the communication among the agents/vehicles and the information-sharing play an important role on the overall system performance and stability.

# CHAPTER 4

# COOPERATIVE CONTROL OF MULTI-AGENTS/MULTI-ROBOTS SYSTEMS MAS-MRS

## Contents

## 4.1 Introduction

Among the research branches of the cooperative control, the formation control of a MASs/MRSs is considered as an interesting and very active research topic. This is due to its main role for the success of any potential task executed by a MRSs [3]. Furthermore, formation control is applied to numerous areas and wide range of applications such as, target enclosing, sensor networks, cooperative surveillance and localization [10,12,13,37].

In general context, depending on whether the formation has a reference trajectory or not, formation control can be categorized into two sub-branches. Firstly, the **formation stabilization** (leaderless), that refers to design cooperative controllers for a MASs to only achieve a desired geometric shape. Secondly, the distributed **formation tracking** or leader-follower formation tracking. In this latter, the followers agents seek achieving TVF configuration while tracking the trajectory of the real/virtual leader [6].The formation control consists on designing algorithms for a group of networked agents with on-board sensing to achieve time-invariant or time-varying geometric configuration among the agents (i.e., circle, square) autonomously, while keep being stable[3,6].

This chapter presents a unified framework of TVF control design for homogeneous $LTI$ MASs based on an **observer viewpoint** from undirected to directed topology, and from a leader with zero control input to a one with **unknown** and **bounded** control input. The followers can form a TVF shape which is specified by piecewise continuously differential vectors. The leader's trajectory (case of real leader), which is available to only a subset of followers, is also time-varying. The agents are equipped with on-board sensors and communicating capabilities (i.e., transceivers) to measure and exchange the relative output measurements and the relative observer outputs among themselves depending on the interaction topology structure.

In this chapter, we reveal the control design from the distributed observer-type viewpoint. So all the controllers (i.e., protocols) are fully **distributed**. Hence, no global information (i.e., neither the total number of the agents nor the smallest positive eigenvalue of the Laplacian matrix) are required in the design of the controllers.

In addition, we deal in this study with time-varying formation and agents featured by general linear dynamics, which is more general compared with many existing studies conducted for particular types of systems dynamics and invariant-formation such as, first order dynamics [32, 67], second order dynamics [33, 34, 68] and time-invariant formation tracking with general linear dynamics [69]. Finally, the conducted studies on TVF tracking in this chapter rely only on the knowledge of the agents dynamics (agents have identical dynamics), the relative output measurements $y_i - y_j$, the internal observer relative-outputs $v_i - v_j$ and the leader relative-output $y_i - y_0$. Therefore, TVF tracking based relative outputs is more applicable in real industry than the states one as in [8, 41]. This is due to the fact that some system states may be unavailable or inaccessible in reality.

First, useful mathematical preliminaries, Lemmas, and terminologies on graph theory are first reviewed in the following of this section. Next, Section-4.2 reveals the mathematical framework of the consensus control protocols for LTI MASs, which represents the background of the TVF tracking problem. In Section-4.3, Definitions and problem statement are presented. In Section-4.4, the TVF tracking problem is developed and solved step by step under the assumption of a leader whose control input is zero. Next, in Section-4.5, the TVF tracking problem is solved considering a leader with nonzero control input. Finally, numerical simulation are provided in Section-4.6 to verify and to confirm the conjecture of the developed TVF tracking controllers.

### 4.1.1 Notation

The following notions will be used throughout this chapter. $\mathbb{R}^{n\times m}$ and $\mathbb{R}^n$ denote the $n \times m$ real matrix space and $n-$dimentional Euclidean vector space, respectively. Let $diag\,(a_1, \ldots, a_n)$ be a diagonal matrix with $(a_1, \ldots, a_n)$ being the diagonal entries. Let $\underline{0}_n$ and $\underline{1}_n$ be the all$-$zero and all$-$one column vectors in $\mathbb{R}^n$. $I_n$ is used to represent the identity matrix in $\mathbb{R}^{n\times n}$. Let the superscript $T$ be the transpose for a matrix. A Matrix $G$ is Hurwitz (or stable) if all its eigenvalues have strictly negative real parts. We denote $G > 0$ if the matrix $G$ is symmetric and positive definite. For a symmetric matrix A, we denote by $\lambda_{min}(\mathrm{A})$ and $\lambda_{max}(\mathrm{A})$ the minimal and maximal (real part) eigenvalues of matrix A, respectively and $||x||$ as the 2-norm of a vector $x$.

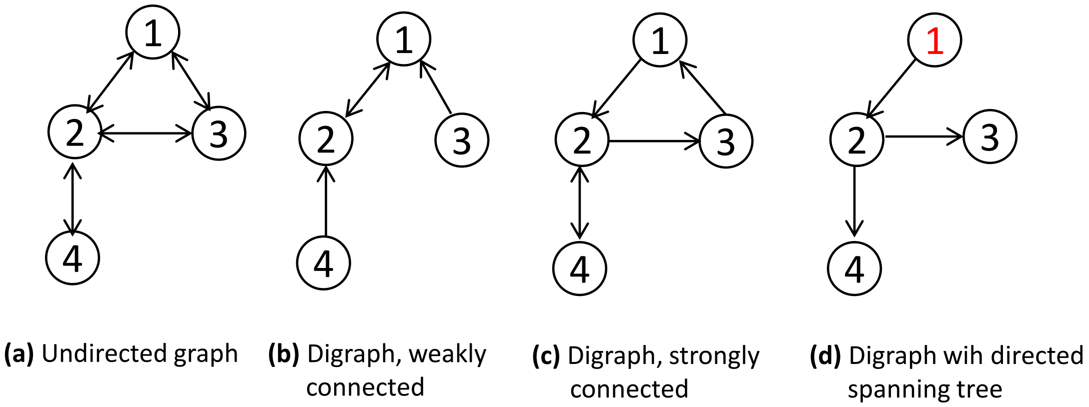
### 4.1.2 Graph theory and mathematical preliminaries

Some useful preliminaries and concepts of graph theory are presented on the following. A team of agents interacts with each other via communication or sensing networks (i.e., exchanging information locally) to achieve collective objectives. It is convenient to model the information exchanges among agents by directed or undirected graphs.

The information exchange among a networked linear multi-agent system is described by a weighted graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{A})$ that represents the connections between agents, where $\mathcal{V} = \{1, \cdots, N\}$ and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denoting the node-set and the edge-set respectively. We denote by $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$ the weighted adjacency matrix defined as $a_{ij} > 0$ if there exists an edge $(j, i) \in \mathcal{E}$, and $a_{ij} = 0$ otherwise. An edge $(i, j) \in \mathcal{E}$ where $i$ is a parent node and $j$ is a child node, means that agent $j$ can receive information from agent $i$, but not necessary conversely and agent $i$ is neighbour of agent $i$. The set of the neighbours of node $i$ is denoted by $\mathcal{N}_i = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}\}$, whose cardinality is called the in-degree of node $i$. Thus, the incidence matrix $\mathcal{D} \in \mathbb{R}^{N \times N}$ of a graph is a diagonal matrix with positive entries representing the in-degree of all the nodes. A graph is defined as being *balanced* when it has the same number of ingoing and outgoing edges for all the nodes. The Laplacian matrix $\mathcal{L} = [l_{ij}] \in \mathbb{R}^{N \times N}$ associated with the graph $\mathcal{G}$ is defined as,

$$\begin{cases} l_{ii} = \sum\limits_{i \neq j} a_{ij}, \\ l_{ij} = -a_{ij}, i \neq j \end{cases} \tag{4.1}$$

The Laplacian matrix $\mathcal{L}$ can be further defined as $\mathcal{L} = \mathcal{D} - \mathcal{A}$. The graph is said undirected if the following property is satisfied, $(i, j) \in \mathcal{E}$ implies $(j, i) \in \mathcal{E}, \forall i, j \in \mathcal{V}$. Otherwise, the graph is directed (or digraph). A directed path is a sequence of ordered edges of the form $(i_1, i_2)$, $(i_3, i_4)$, $\ldots, (i_{k-1}, i_k)$, where $i_p \in \mathcal{V}, p = \{1, \cdots, k\}$.



**(a)** Undirected graph **(b)** Digraph, weakly connected **(c)** Digraph, strongly connected **(d)** Digraph wih directed spanning tree

Figure 4.1: An illustrative example of different types of graphs.

If there exists a directed path from every node to every other node, the digraph (directed graph) is called strongly connected [32],[70]. Thus, any undirected graph is connected. A directed spanning tree of a digraph is a directed tree formed by graph edges that connects all the nodes of the graph. It is said that a digraph contains a directed spanning tree if there exists at least one node called *root-node* such that there exists a directed path from this latter to all other nodes. A digraph is strongly connected, implies the existence of at least a directed spanning tree, but not necessary conversely. A digraph being strongly connected is stronger condition than having a directed spanning tree.

The Kronecker product of matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ is described by $A \otimes B$ is defined as, [20],

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}$$

The Kronecker product satisfies the following properties,

$$\begin{aligned} &A \otimes (B + C) = A \otimes B + A \otimes C \\ &(A \otimes B)(C \otimes D) = (AC) \otimes (BD) \\ &(A \otimes B)^T = \left(A^T \otimes B^T\right) \\ &(A \otimes B)^{-1} = \left(A^{-1} \otimes B^{-1}\right) \end{aligned}$$

where the matrices are assumed to be compatible for multiplication.

**Definition 4.1** *A square matrix $\mathcal{L} = \left[l_{ij}\right] \in \mathbb{R}^{n\times n}$ is called a singular (non-singular) $M-$matrix, if all its off-diagonal elements are non-negative (positive) and all eigenvalues of $\mathcal{L}$ have non-negative (positive) real parts [21].*

*The following properties (conditions) are equivalent for the nonsingular $M-$matrices. Let $\mathcal{L} = \left[l_{ij}\right] \in \mathbb{R}^{n\times n}$ be a nonsingular $M-$matrix, then,*

*1. The leading principal minors of $\mathcal{L}$ are all positive.*

*2. The eigenvalues of $\mathcal{L}$ have positive real parts.*

*3. $\mathcal{L}^{-1}$ exists and has nonnegative elements.*

*4. There exist vectors $w, v > 0$ such that the matrices $\mathcal{L}v$, $\mathcal{L}^T w$ are both positive definite.*

*5. $[\mathcal{L}]x = 0$, $x \in \mathbb{R}^n$ implies that $x = 0$.*

The **M-matrices** and their properties are important in the analysis of graphs and are used in the MASs stability analysis due to the fact that the graph Laplacian matrix $\mathcal{L} = \mathcal{D} - \mathcal{A}$ which model the interaction among the agents (i.e., information exchange) is an $M-$matrix.

**Example:** For the graphs depicted in Fig. 4.1, we compute the corresponding Laplacian matrices as,

$$\mathcal{L}_a = \begin{bmatrix} 1 & -2 & -4 & -3 \\ 1 & -1 & -5 & -5 \\ -2 & -2 & -2 & -3 \\ -10 & -8 & -4 & -4 \end{bmatrix}; \mathcal{L}_b = \begin{bmatrix} 1 & -2 & -4 & -3 \\ 1 & -1 & -5 & -5 \\ -2 & -2 & -2 & -3 \\ -10 & -8 & -4 & -4 \end{bmatrix}$$

$$\mathcal{L}_c = \begin{bmatrix} 1 & -2 & -4 & -3 \\ 1 & -1 & -5 & -5 \\ -2 & -2 & -2 & -3 \\ -10 & -8 & -4 & -4 \end{bmatrix}; \mathcal{L}_d = \begin{bmatrix} 1 & -2 & -4 & -3 \\ 1 & -1 & -5 & -5 \\ -2 & -2 & -2 & -3 \\ -10 & -8 & -4 & -4 \end{bmatrix}$$

From the definition of the Laplacian matrix and also the above example, it is easy to see that $\mathcal{L}$ is diagonally dominant and has nonnegative diagonal entries. Since $\mathcal{L}$ has zero row sums, 0 is an eigenvalue of $\mathcal{L}$ with an associated eigenvector $\underline{1}_4$. According to

Gershgorin's disc theorem [71], all nonzero eigenvalues of $\mathcal{L}$ are located within a disk in the complex plane centred at $d_{max}$ and having radius of $d_{max}$, where $d_{max}$ denotes the maximum in-degree of all nodes. According to the definition of $M-$matrix in the last subsection, we know that the Laplacian matrix $\mathcal{L}$ is a singular $M-$matrix.

### 4.1.3 Stability theory and technical tools

The following lemmas are keys to our development and stability analysis in the next sections.

**Lemma 4.2** *[21,72]: For a non-singular M-matrix $\widetilde{\mathcal{L}}_1 \in \mathbb{R}^{N\times N}$, there exists a positive diagonal matrix $G = diag\{g_1, \ldots, g_N\}$, such that $G\widetilde{\mathcal{L}}_1 + \widetilde{\mathcal{L}}_1^T G > \lambda_0 I_N > 0$ where $[g_1, \ldots, g_N]^T = \widetilde{\mathcal{L}}_1^{-1} 1_N$ and $\lambda_0 > 0$ is the smallest eigenvalue of $G\widetilde{\mathcal{L}}_1 + \widetilde{\mathcal{L}}_1^T G$.*

**Lemma 4.3** *(Young's Inequality) [73]: Let p and q be positive real numbers satisfying $1/_p + 1/_q = 1$, the following inequality holds for nonnegative real numbers a and b as, $ab \leq \frac{a^p}{p} + \frac{b^q}{q}$.*

**Lemma 4.4** *[20,51]: If a directed graph $\mathcal{G}$ contains a directed spanning tree, then zero is a simple eigenvalue with $\underline{1}$ as right eigenvector of the associated Laplacian matrix $\mathcal{L}$. Furthermore, the rest of the eigenvalues are positive real parts.*

**Lemma 4.5** *Consider the autonomous system,*

$$\dot{x} = f(x) \tag{4.2}$$

*where $f : D \to \mathbb{R}^n$ is a locally Lipschitz map from a domain $D \in \mathbb{R}^n$ into $\mathbb{R}^n$. Suppose that $x_0 = 0$ is an equilibrium point of (4.2). Let $V : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function such that $V(x_0 = 0) = 0$ and $V(x) > 0, \forall x \neq 0$, $V(x)$ is radially unbounded (i.e., $V(x) \to \infty$ as $\|x\| \to \infty$), and $\dot{V} < 0, \forall x \neq 0$. Then, $x_0 = 0$ is globally asymptotically stable[63].*

**Lemma 4.6** ***LaSalle Invariance Principle*** *[74] Let $\Omega \subset D$ be a compact set that is positively invariant with respect to (4.2). Let $V : D \to \mathbb{R}$ be a continuously differentiable function such that, $\dot{V} \leqslant 0$ in $\Omega$. Let E be the set of all points in where $\dot{V} = 0$ and M be the largest invariant set in E. Then, every solution starting in $\Omega$ approaches M as $t \to \infty$.*

Note that in many applications the construction of $V(x)$ will guarantee the existence of a set $\Omega$. In particular, if $\Omega_c = x \in \mathbb{R}^n : V(x) \leqslant c$ is bounded and $\dot{V}(x) \leqslant 0$, then we can choose $\Omega = \Omega_c$.

**Lemma 4.7** ***Barbalat's Lemma*** *[63] If the differentiable function $f(t)$ has finite limit as $t \to \infty$, and $\dot{f}(t)$ is uniformly continuous, then $\dot{f}(t) \to 0$ as $t \to \infty$.*

A simple sufficient condition for a differentiable function to be uniformly continuous is that its derivative is bounded. The Lemma-4.7 can be stated differently as, if the function $\dot{f}(t)$ is uniformly continuous $\mathbb{R}^+ \to \mathbb{R}$, and $\int_0^{+\infty} \dot{f}(\tau)d\tau$, exists and finite, then, $\lim\limits_{t\to\infty} \dot{f}(t) = 0$.

## 4.2 Overview on MASs consensus control

It has been shown that many existing formation control approaches such as leader-follower, behavioural and virtual structure approach can be unified within the general framework of consensus control [18]. The consensus problem has been extensively studied for different continuous or discrete dynamics node under fixed or switching topologies. The difference between formation control and consensus control of a MASs is that in consensus control, an agreement among the agents in term of some variable of interest (i.e., states) has to be reached, while in formation control, a desired geometric configuration is required to be achieved by the agents. In other words, the MASs formations can be achieved through reaching consensus on the center point of the formation. Thus, the MASs formation control is an extension of a MASs consensus control.

In this section we present a general overview of the main frameworks on consensus control as it represents the basis of the distributed output TVF tracking of MASs in the next sections. Consensus means that a team of agents reaches an agreement on a common value by interacting with each other via a sensing or communication network. For the consensus control problem, the main task is to design appropriate distributed controllers, usually called consensus protocols, based on local information of neighboring agents to achieve consensus.

Two pioneer papers on consensus control are [75] and [71]. A theoretical explanation was provided in [75] for the alignment behavior observed in the Vicsek model [76] and a general framework of the consensus problem for networks of integrators was proposed in [71]. Since then, the consensus problem has been extensively studied by numerous researchers from various perspectives (i.e., the type of node dynamics, the type of the exchanged data, the distribution property, and the type of the communication graph).

The consensus control was developed for first order dynamics node i.e., $\dot{x} = u$ [32,71], second order dynamics, i.e., $\ddot{x} = u$, [33], high-order integrator dynamics, i.e., $x^{(n)} = u$ [77] and for GLS dynamics[20], where the node's dynamics is described as in (4.4).

The consensus problem neither the TVF tracking of MASs with each node being a GLS dynamics is more challenging than the other types of node's dynamics. The importance of studying consensus for general linear MASs lies in at least two aspects. First, the general linear agents are more representative, which contain the first-order, second-order, and high-order integrators as special cases. Second, consensus of general linear MASs can be regarded as local consensus of nonlinear MASs, and it paves a way for studying more complex MASs, e.g., systems subject to different types of uncertainties.

Mathematically, the distributed consensus problem for MASs featured by (4.4) with neighbouring relative-state exchanging can be stated as follows, *the consensus problem is solved for MASs featured by* (4.4) *under fixed directed interaction topology if for any given agents initial states* $x_i(0), i \in [1, N]$, *the following limits hold,* $\lim_{t \to \infty} ||x_i(t) - x_j(t)|| = 0; i, j \in [1, N]$. Thus, the static consensus protocol (i.e., control law) that solves the aforementioned consensus problem is given as [20],

$$u_i = cK \sum_{j=1}^{N} a_{ij}(x_i - x_j), i = 1, ..., N, \tag{4.3}$$

where $x_i$, $u_i$, refer to the state and control input of agent $i$ (see (4.4)), $c > 0 \in \mathbb{R}$ is called the (constant) coupling gain or weight, $K \in \mathbb{R}^{p \times n}$ is the feedback gain matrix, and $a_{ij}$ is the $(i, j)$-th entry of the adjacency matrix $\mathcal{A}$ associated with the graph $\mathcal{G}$

among the agents. Note that the condition for reaching consensus depends on three correlated factors: *(i)* the agent dynamics, *(ii)* the design parameters of the consensus protocol, *(iii)* and the nonzero eigenvalues of the Laplacian matrix $\mathcal{L}$ of the communication graph [20]. Reader can refer to [20] for the stability analysis and the sufficient conditions for reaching the consensus problem with relative state feedback and also for relative output-feedback cases under fixed directed graphs. In [78,79], the state-feedback consensus problem is studied and solved in fully distributed fashion under *Leader-Follower* scheme, where the agent's protocol construction does not depend on the knowledge of the eigenvalues of the Lapalcian matrix. The frameworks [78,79] were extended to the case of output-feedback consensus control [51], where sufficient conditions for consensus to be reached were derived in fully distributed fashion.

## 4.3 Definitions and problem statement

Consider a MASs consisting of $(N+1)$ identical agents (one Leader and $N$ followers) with time-continuous general Linear Time-Invariant (LTI) dynamics that could be regarded as the linearized model of a nonlinear systems. The dynamics of the *i-th* agent is given,

$$\begin{aligned} \dot{x}_i(t) &= Ax_i(t) + Bu_i(t), \\ y_i(t) &= Cx_i(t), i \in [0, N] \end{aligned} \tag{4.4}$$

In (4.4), $x_i = [x_{i1}, \ldots, x_{in}]^T \in \mathbb{R}^n$, $u_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}^q$ are the i-*th* agent's state vector, control input and measured output, respectively. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$ and $C \in \mathbb{R}^{q\times n}$ are constant matrices.

Without loss of generality, the agents in (4.4) indexed by $1, \ldots, N$ are the followers and the agent indexed by 0 is the leader, whose role is to decide the whole system moves. The leader receives no information from followers and its measured output is available to only a subset of followers.

**Definition 4.8** *The desired geometric TVF shape to be maintained by the identical homogeneous N followers is described by a relative offset vector $h(t) = \left[h_i^T(t), \ldots, h_N^T(t)\right]^T \in \mathbb{R}^{nN}$, where $h_i(t) \in \mathbb{R}^n$ being a piecewise continuously differentiable and compatible with agent* *dynamics given in* (4.4) *for* $i \in [1, N]$,

$$\dot{h}_i(t) = Ah_i(t) + Bu_{i,r}(t), i \in [1, N] \tag{4.5}$$

where $u_{i,r}(t) \in \mathbb{R}^p$ is a free reference input for generating $h_i(t)$, the relative offset or coordinate of follower $i$ with respect to the leader. The desired TVF format can be written as,

$$\dot{h}_i(t) = (A + BK_h) h_i(t), i \in [1, N] \tag{4.6}$$

where $K_h \in \mathbb{R}^{p \times n}$ can be a constant or a time-varying matrix. The gain matrix $K_h$ provides the freedom to design any desired TVF shape while satisfying compatibility with agent's dynamics. The TVF form (4.6) gives the ability to provide rotation, translation and scalability operations by appropriate chose of $K_h$. The importance of this formation form lies on the flexibility and applicability in many scenarios and contexts such as target-enclosing and obstacles avoidance.

**Remark 4.9** *To explain the Desired TVF shape, an illustrative example is given in Fig.* 4.2*. First, in Fig.* 4.2*(a), the agent labelled i is required to maintain a time-varying relative state vector* $h_i(t)$ *relative to the leader state* $x_0(t)$*, and as a result, all the follower agents maintain a specific shape (i.e, in this example a hexagon form) around the leader while tracking its trajectory. Secondly, in figure Fig.* 4.2*(b), the followers form an hexagon shape while tracking the leader simultaneously. Thus,* $Ch_i(t)$ *represent in this case the relative output offset of agent i relative to the leader. As a result, each agent reference trajectory comes from* $y_0(t) + Ch_i(t)$*.*

**Definition 4.10** *The multi-agent system MASs described in* (4.4) *is said to achieve the output TVF tracking control if for any given initial states* $x_i(0), i \in [0, N]$*, the following limits hold,*

$$\lim_{t \to \infty} ||y_i(t) - y_0(t) - Ch_i(t)|| = 0, i = [1, ..., N] \tag{4.7}$$

It can be seen from Definition-4.10, that the goal behind the TVF tracking problem based L-F approach is to ensure that the followers achieve an agreement on leader's output $y_0(t)$, and simultaneously each follower keeps a desired relative offset $Ch_i(t)$ with respect to $y_0(t)$. As a result, the desired formation shape denoted by $h(t)$ is

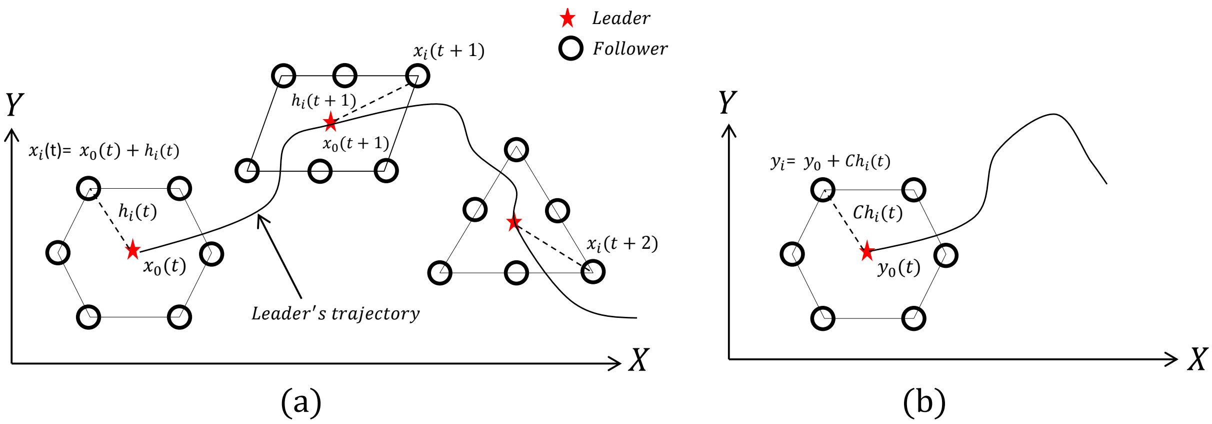


Figure 4.2: An illustrative example of the desired TVF shape.

achieved. It is worth noting that selecting $\sum\limits_{i=1}^{N} h_i\left(t\right) = 0$, implies the leader will lie inside the desired TVF which corresponds to target enclosing and trajectory tracking problem providing that the leader is real and has nonzero control input. Moreover, in the case where $h(t) \equiv 0$, the TVF tracking problem becomes the Consensus tracking problem [20].

**Example 4.11** ***The desired TVF format:*** *We show in this example how the desired TVF shape changes over time through simulation. Let consider a MASs consisting of a leader labelled by 0 and 6 followers labelled from* 1 *to* 6. *Now, let the state vector of agent follower i be* $x_i = \left[x_{i,1}^T, \ldots, x_{i,4}^T\right]^T \in \mathbb{R}^4$. *The identical agents dynamics is supposed to be defined by the following matrices A and B,*

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

*We choose* $C = \left[\begin{array}{cc} I_2 & 0_2 \end{array}\right]$.

Inspired by [48, 80], the desired formation shape $h(t)$ will be constructed by three parameters that are: the formation shape $\bar{h}$, the formation scale $r(t)$ and the formation rotating frequency around its center $\bar{\omega}(t)$. For the agent $i, i \in [1,6]$, the offset $\bar{h}_i = [\bar{h}_{i1}, ..., \bar{h}_{i6}]^T$ can be defined as,

$$\bar{h}_i = \begin{cases} r\sin(\bar{\omega}t + (i-1)\,\pi/3) - r\cos(\bar{\omega}t + (i-1)\,\pi/3) \\ 2r\sin(\bar{\omega}t + (i-1)\,\pi/3) \\ r\bar{\omega}[\cos(\bar{\omega}t + (i-1)\,\pi/3) + \sin(\bar{\omega}t + (i-1)\,\pi/3)] \\ 2r\bar{\omega}\cos(\bar{\omega}t + (i-1)\,\pi/3) \end{cases}$$

Now, different formation shapes (i.e. parallelogram, circle, triangle,...) could be designed as a combination of $\bar{h}_i$. Let $h(t)$ to be designed as,

$$h(t) = \begin{matrix} \left[\bar{h}_1^T, \frac{\bar{h}_1^T+\bar{h}_3^T}{2}, \bar{h}_3^T, \bar{h}_4^T, \frac{\bar{h}_4^T+\bar{h}_6^T}{2}, \bar{h}_6^T\right]^T, 0 \le t \le 15 \\ \left[\bar{h}_1^T, \frac{\bar{h}_1^T+\bar{h}_3^T}{2}, \bar{h}_3^T, \frac{\bar{h}_3^T+\bar{h}_5^T}{2}, \bar{h}_5^T, \frac{\bar{h}_1^T+\bar{h}_5^T}{2}\right]^T, 15 < t \le 30 \\ \left[\bar{h}_1^T, \bar{h}_2^T, \bar{h}_3^T, \bar{h}_4^T, \bar{h}_6^T, \bar{h}_6^T\right]^T, 30 < t \le 45 \end{matrix}$$

From the description of $h(t)$, the desired formation shape is a triangle as $t \in [0s, 15s]$, then a parallelogram shape as $t \in [15s, 30s]$ and as hexagons shape when $t \in [30s, 45s]$. We set $\bar{\omega}(t) = 0.15$, and we set $r(t)$ as,

$$r\,(t) = \begin{cases} 3, t \le 15 \\ 3 + 3e^{-0.18(t-15)}, t > 15 \end{cases}$$

By setting $r(t)$ time-varying and $\bar{\omega}(t) \neq 0$, the formation shape will rotate around the leader and change scale over time. The above facts offer high flexibility on the choice of the desired MASs formation as shown in Fig. 4.3. It can be checked that the formation constraint (4.5) is satisfied with $u_{h,i}\,(t) = K_h h_i(t)$. Thus, using (4.6) and knowing $A$ and $B$, the matrix $K_h$ is found to be,

$$K_h = \begin{bmatrix} -\bar{\omega}^2 & 0 & 0 & 0 \\ 0 & -\bar{\omega}^2 & 0 & 0 \end{bmatrix}$$

## 4.4 Distributed TVF tracking with zero leader input

It is naturally to start first in section-4.4.1 solving the output-feedback TVF tracking problem for MASs with a leader whose control input is zero, and further with an

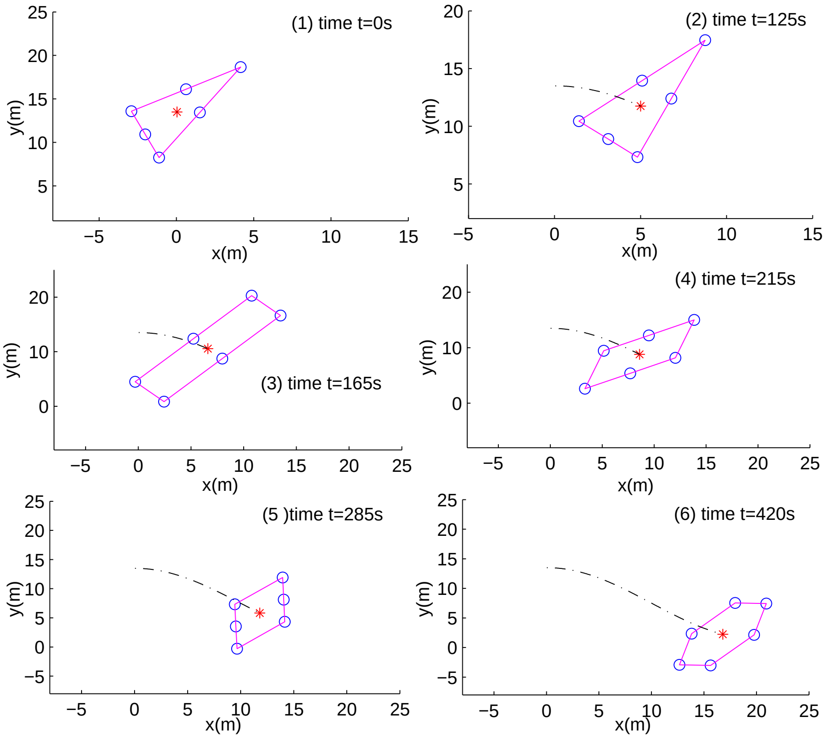


Figure 4.3: An illustrative example of TVF changing mechanism.

undirected interaction topology (i.e., undirected graph), in which the agents are exchanging the relative output measurements bidirectionally, which is mathematically less challenging. Moreover, this assumption (i.e., undirected graph among the agents) is a strong requirement and may be not feasible in permanence in real application, due to the movements of the agents during the TVF. In addition, it implies an increase of the utilized communicating resources. In section-4.4.2, we consider the directed graph case with full access to leader's measured output, which is relatively less challenging. The fact that all the follower agents have knowledge of the leader's control input render the formation stability-analysis easier to demonstrate. In the other side, the graph among the agents is required to be directed rather than undirected, hence, less communication resources are utilized compared to the undirected graphs. After that, in section-4.4.3, we treat the case when the followers have partial access to leader's output and the interaction topology is supposed to be directed and required only to have a spanning tree. Hence, the communication-links number among the agents will be reduced compared to the case when the digraph is being strongly connected. The above controllers are valid only for a leader whose control input is zero, which is a severe constraint and less practical in real applications.

As detailed previously, the whole networked MASs is consisting of $(N+1)$ homogeneous $LTI$ agents ($N$ followers and one leader) of a general linear dynamics described as in (4.4). Physically, the followers are supposed to be equipped with wireless capabilities (i.e., transceivers) of certain range, enabling them to exchange their measured outputs and other signals via the communication topology.

Without loss of generality, the agents in (4.4) indexed by $1, \dots, N$ are the followers and the agent indexed by 0 is the leader, whose role is to decide the whole system moves. The leader receives no information from followers and its measured output is available to only a subset of followers.

In the following, we denote by $\mathcal{A} \in \mathbb{R}^{(N+1)\times(N+1)}$ and $\mathcal{L} \in \mathbb{R}^{(N+1)\times(N+1)}$ the Adjacency matrix and the Laplacian matrix respectively, associated with the communication graph $\mathcal{G}$ among the whole system. Because the node indexed by 0 is the leader which has no neighbors, $\mathcal{A}$ and $\mathcal{L}$ can be partitioned as,

$$\mathcal{A} = \begin{bmatrix} 0 & 0_{1\times N} \\ \widetilde{\mathcal{A}}_2 & \widetilde{\mathcal{A}}_1 \end{bmatrix}; \mathcal{L} = \begin{bmatrix} 0 & 0_{1\times N} \\ \widetilde{\mathcal{L}}_2 & \widetilde{\mathcal{L}}_1 \end{bmatrix} \tag{4.8}$$

In the above equation (4.8), $\widetilde{\mathcal{L}}_2 \in \mathbb{R}^{N\times 1}$. $\widetilde{\mathcal{L}}_1 \in \mathbb{R}^{N\times N}$ is representing the Lapalcian matrix associated to the subgraph $\widetilde{\mathcal{G}}_1$ among the followers. By the definition of Laplacian matrix, it is straightforward to see that $\widetilde{\mathcal{L}}_2 = -\widetilde{\mathcal{A}}_2$, where $\widetilde{\mathcal{A}}_2 = [a_{i0}]\,, i \in [1, N]$ defines whether a follower receives information from the leader if $a_{i0} > 0$ and doesn't receive information if $a_{i0} = 0$.

**Assumption 4.12** *The pair $(A, B)$ is stabilizable.*

**Assumption 4.13** *The pair $(A, C)$ is detectable.*

Its worth noting that the stabilizability and detectability of a LTI system are less strong requirements compared to the controllability and observability. Thus, the above assumptions are reasonable and suitable for physical systems.

The desired TVF shape information $h(t)$ is presented in Section-4.3. In order to reach the TVF control, a variety of protocols based on absolute or relative states have been proposed, e.g., in [34], [36]. For instance, a TVF protocol based on absolute/relative state information of neighboring agents is given in [36] as,

$$u_i(t) = K_1 x_i(t) + K_2 \sum_{j=1}^{N} c_{ij}(t) a_{ij} \left[ (x_i(t) - x_j(t)) - (h_i(t) - h_j(t)) \right] \tag{4.9}$$

However, the state information $x_i(t)$ and $x_j(t)$ in (4.9) may not be always available in most practical applications. Whereas, the output information, (i.e., $y_i(t)$ and $y_j(t)$) is usually accessible all the time. Hence, the output-based adaptive observer-type protocols are studied in this chapter. Furthermore, we deal particularly with the problem of reducing the exchanged information among the networked agents when designing the cooperative controller.

### 4.4.1 Undirected output TVF tracking

**Assumption 4.14** *The communication subgraph $\widetilde{\mathcal{G}}_1$ among followers is undirected with the its corresponding adjacency matrix $\widetilde{\mathcal{A}}_1$ and Laplacian matrix $\widetilde{\mathcal{L}}_1$. The graph $\mathcal{G}$ of the whole system contains a spanning tree with the adjacency matrix $\mathcal{A}$ and Laplacian matrix $\mathcal{L}$*

In this sub-section, we deal first with the case when the leader's input control is zero ($u_0(\mathrm{t}) = 0$), which means that the leader could be a virtual one. This assumption is common in previous work [46], [47]. However, it imposes severe limitations on the desired formation trajectory generated from the equation $\dot{x}_0(t) = Ax_0(t)$. This is due to the fact that the matrix $A$ is unchangeable. We assume that the interaction topology is time-invariant. Let recall the Definition-4.10. The objective is to develop a fully distributed protocol for the followers in order to reach and maintain a TVF shape defined as in Definition-4.8, while simultaneously tracking the leader's trajectory. It is assumed that each agent exchanges the relative outputs, rather than the relative states, of its neighbors with respect to itself via the undirected communication graph. The term (t) is omitted in the following for writing convenience, and each follower can get access to the relative output measurements as,

$$y_{ij} = y_i - y_j, y_{i0} = y_i - y_0, \{i,j\} \in [1,N] \tag{4.10}$$

Inspired by the proposed **edge-based** adaptive consensus control given in [78], an extension to the edge-based adaptive TVF tracking control is given as follows [81], [48],

$$\begin{aligned} u_i &= K_h h_i + K v_i \\ \dot{v}_i &= (A + BK)\, v_i + F\left[\sum_{j=1}^{N} a_{ij} c_{ij} \left(\bar{c}_{ij} - y_{ij}\right) + a_{i0} c_i \left(\bar{c}_i - y_{i0}\right)\right] \\ \dot{c}_{ij} &= k_{ij} a_{ij} \left(\bar{c}_{ij} - y_{ij}\right)^T \Gamma \left(\bar{c}_{ij} - y_{ij}\right) \\ \dot{c}_i &= k_i a_{i0} \left(\bar{c}_i - y_{i0}\right)^T \Gamma \left(\bar{c}_i - y_{i0}\right), i \in [1, N] \end{aligned} \tag{4.11}$$

where the terms $\bar{c}_{ij}$ and $\bar{c}_i$ are given as,

$$\bar{c}_i = C\left(v_i + h_i\right), \bar{c}_{ij} = \bar{c}_i - \bar{c}_j \tag{4.12}$$

In (4.11), $v_i \in \mathbb{R}^n$ is the distributed observer state (the protocol internal state) introduced to estimate the TVF tracking error $\tilde{x}_i$ defined in (4.13), $y_{ij}$ and $y_{i0}$ are already defined in (4.10), $k_{ij} = k_{ji}, k_i$ are positive constants, $a_{ij}$ is the $(i,j)-$th entry of the adjacency matrix $\widetilde{\mathcal{A}}_1$, $c_{ij}(t)$ denotes the time-varying coupling weight between the follower $i$ and $j$ satisfying $c_{ij}(0) = c_{ji}(0) > 0$, while $c_i(t) > 0$ denotes the time-varying coupling weight between follower $i$ and the leader. $K \in \mathbb{R}^{p\times n}, F \in \mathbb{R}^{n\times q}, \Gamma \in \mathbb{R}^{q\times q}$ are feedback gain matrices what will be determined later, while the gain matrix $K_h$ characterize the desired TVF shape (see (4.6)). $a_{i0}$ is the $i-$th entry of $\widetilde{\mathcal{A}}_2$ which satisfies $a_{i0} > 0$ if follower $i$ can get information from leader, otherwise $a_{i0} = 0$.

**Remark 4.15** *The TVF tracking control* (4.11) *dynamically updates the coupling weights $c_{ij}(t)$ and $c_i(t)$ for each edge (i.e., each communication link), thus, the controller is not dependant to any global information such as the total number of agents or the minimal positive eigenvalue of the Laplacian matrix $\mathcal{L}$ as in [35], [82], [38] and [8]. Therefore, the controller is fully distributed and applicable to a large scale multi-agents system. However, this is valid for only undirected interaction topologies among followers.*

From Definition-4.10, let define the TVF tracking error $\tilde{x}_i \in \mathbb{R}^n$ for each follower-agent as,

$$\tilde{x}_i = \left(x_i - x_0 - h_i\right), i = [1, \cdots, N] \tag{4.13}$$

As we included an observer $v_i \in \mathbb{R}^n$ in (4.11) to estimate the TVF tracking error of each follower, the observer-estimating error $e_i \in \mathbb{R}^n$ could be defined as,

$$e_i = (\tilde{x}_i - v_i)\,, i = [1, \ldots, N] \tag{4.14}$$

The following theorem presents a result of designing an observer based adaptive protocol (see definition-4.10) for solving the Leader-Follower TVF tracking problem under undirected interaction topology.

**Theorem 4.16** *[48], [81] The fully distributed TVF tracking problem is solved with Assumptions 4.12, 4.13 and 4.14 under the protocol (4.11) if $(A + BK)$ is Hurwitz, $\Gamma = I$ and $F = -PC^T$, where $P^{-1} > 0$ is a solution to the following LMI:*

$$P^{-1}A + A^T P^{-1} - 2C^T C < 0 \tag{4.15}$$

*Furthermore, the coupling weights $c_{ij}(t)$, $c_i(t)$, $\{i, j\} \in [1, N]$ converge to some finite steady-state value.*

***Proof of Theorem-4.16:*** From (4.7) and (4.13), we have $(y_i - y_0 - Ch_i) = C\tilde{x}_i$, thus, from Definition-4.10, the goal is to prove that $\lim\limits_{t\to\infty} \tilde{x}_i = 0, i = [1, ..., N]$. The formation tracking error dynamics $\dot{\tilde{x}}_i$ is derived from (4.4) and (4.6) as,

$$\dot{\tilde{x}}_i = A\tilde{x}_i + BKv_i \tag{4.16}$$

Now, we substitute (4.16) in (4.11) yields,

$$\begin{aligned}\dot{e}_i &= Ae_i + FC\left[\sum_{j=1}^{N} a_{ij}c_{ij}\left(e_i - e_j\right) + a_{i0}c_i e_i\right]\\ \dot{c}_{ij} &= k_{ij}a_{ij}\left(e_i - e_j\right)^T C^T \Gamma C\left(e_i - e_j\right)\\ \dot{c}_i &= k_i a_{i0}(e_i)^T C^T \Gamma C e_i\end{aligned} \tag{4.17}$$

Let consider the Lyapunov candidate function,

$$V_1 = \sum_{i=1}^{N} e_i{}^T P^{-1} e_i + \sum_{i=1}^{N}\sum_{j=1, i\neq j}^{N} \frac{\left(c_{ij} - \alpha\right)^2}{2k_{ij}} + \sum_{i=1}^{N} \frac{\left(c_i - \alpha\right)^2}{k_i} \tag{4.18}$$

The parameter $\alpha$ is a constant to be determined later. The time derivative of $V_1$ considering the dynamics $\dot{e}_i$ in (4.17),

$$\begin{aligned} \dot{V}_1 &= 2\sum_{i=1}^{N}(c_i-\alpha)\,a_{i0}{e_i}^T C^T \Gamma C e_i + 2\sum_{i=1}^{N}{e_i}^T P^{-1} \mathrm{F} C\left[\sum_{j=1}^{N}a_{ij}c_{ij}\left(e_i-e_j\right)+a_{i0}c_i e_i\right] \\ &+2\sum_{i=1}^{N}{e_i}^T P^{-1} e_i + \sum_{i=1}^{N}\sum_{j=1,i\neq j}^{N}\left(c_{ij}-\alpha\right)a_{ij}\left(e_i-e_j\right)^T C^T\Gamma C\left(e_i-e_j\right). \end{aligned} \tag{4.19}$$

Because of $k_{ij}=k_{ji}$, $c_{ij}(0)=c_{ji}(0)$ and $a_{ij}=a_{ji}$ due to the property of undirected graph among followers (see Assumption-4.14) with the matrix $\Gamma$ being symmetric, it follows from protocol (4.11) that $c_{ij}(\mathrm{t})=c_{ji}(t),\forall t\geq 0$. Then, the following equality holds,

$$\begin{aligned} &\sum_{i=1}^{N}\sum_{j=1,i\neq j}^{N}\left(c_{ij}-\alpha\right)a_{ij}\left(e_i-e_j\right)^T C^T\Gamma C\left(e_i-e_j\right) = \\ &2\sum_{i=1}^{N}\sum_{j=1,i\neq j}^{N}\left(c_{ij}-\alpha\right)a_{ij}{e_i}^T C^T\Gamma C\left(e_i-e_j\right) \end{aligned} \tag{4.20}$$

Now, let substitute $F=-P^{-1}C^T$, $\Gamma=\mathrm{I}$ and (4.20) in (4.19) yields,

$$\begin{aligned} \dot{V}_1 &= \sum_{i=1}^{N}{e_i}^T\left(P^{-1}A+A^TP^{-1}\right)e_i - 2\alpha\sum_{i=1}^{N}a_{i0}{e_i}^T C^T C e_i \\ &-2\alpha\sum_{i=1}^{N}\sum_{j=1,i\neq j}^{N}a_{ij}{e_i}^T C^T C\left(e_i-e_j\right) \\ &= e^T\left[I_N\otimes\left(P^{-1}A+A^TP^{-1}\right)-2\alpha\widehat{\mathcal{L}}_1\otimes C^TC\right]e \end{aligned} \tag{4.21}$$

where $\widehat{\mathcal{L}}_1=\widetilde{\mathcal{L}}_1+\mathcal{D}$, with $\widetilde{\mathcal{L}}_1$ being the Laplacian matrix corresponding to the subgraph $\widetilde{\mathcal{G}}_1$ among followers, $\mathcal{D}=\operatorname{diag}\left\{\widetilde{\mathcal{A}}_2\right\}=\operatorname{diag}\left\{a_{10},\ldots,a_{N0}\right\}$ and $e=[e_1^T,\ldots,e_N^T]^T\in\mathbb{R}^{Nn\times 1}$. It is clear that $\mathcal{D}>0$ with at least one diagonal entry being positive since at least one follower gets information from the leader (see Assumption-4.14). Then, $\widehat{\mathcal{L}}_1$ is positive-definite matrix [78], [79]. Now, let $U\in\mathbb{R}^{N\times N}$ be a unitary matrix such that $U^T\widehat{\mathcal{L}}_1U=\Lambda\triangleq diag\{\lambda_1,\ldots,\lambda_N\}$ with $\lambda_i, i\in[1,\ldots,N]$ being the eigenvalues of $\widehat{\mathcal{L}}_1$. Define $\tilde{e}\triangleq\left[\tilde{e}_1^T,\ldots,\tilde{e}_N^T\right]=\left(U^T\otimes I_n\right)e$. It follows from (4.21) that,

$$\begin{aligned} \dot{V}_1 &= \tilde{e}^T\left[I_N\otimes\left(P^{-1}A+A^TP^{-1}\right)-2\alpha\Lambda\otimes C^TC\right]\tilde{e} \\ &= \sum_{i=1}^{N}\tilde{e}_i^T\left(P^{-1}A+A^TP^{-1}-2\alpha\lambda_iC^TC\right)\tilde{e}_i \end{aligned} \tag{4.22}$$

Let choose the constant $\alpha$ sufficiently large such that $\alpha\lambda_i\geq 1, i\in[1,\ldots,N]$. Then it follows from (4.22) that,

$$\dot{V}_1 \leq \sum_{i=1}^{N} \tilde{e}_i^T \left( P^{-1}A + A^T P^{-1} - 2C^T C \right) \tilde{e}_i \leq 0 \tag{4.23}$$

The inequality obtained in (4.23) comes form the LMI (4.15). Since $\dot{V}_1 \leq 0$, so we conclude that $V(t)$ is bounded and so are $c_{ij}(t)$, $c_i(t)$. Moreover, from (4.11) and $\Gamma = I$ it follows that $\dot{c}_{ij}(t) > 0, \dot{c}_i(t) > 0$, thus each coupling weight $c_{ij}(t)$, $c_i(t)$ increases monotonically and converges to some finite value as $t \to \infty$. In the other side, $V_1 \equiv 0$ means that $\tilde{e} = 0$, and so $e = 0$, since $\tilde{e} = \left(U^T \otimes I_n\right) e$. By LaSalle's Invariance principle [74], it turns out that $\lim\limits_{t\to\infty} e(t) = 0$ implies $v(\mathrm{t}) \to \tilde{x}(t)$ as $t \to \infty$. Now, let substitute (4.14) into (4.16) yields,

$$\dot{\tilde{x}}_i = (A + BK)\, \tilde{x}_i - BKe_i \tag{4.24}$$

We now that $(A + BK)$ is Hurwitz from Theorem-4.16 and $\lim\limits_{t\to\infty} e_i = 0, i = [1, ..., N]$, it is straightforward to conclude that $\lim\limits_{t\to\infty} \tilde{x}_i = 0, i = [1, ..., N]$. The proof is completed.

### 4.4.2 Directed output TVF tracking with full access to leader

**Assumption 4.17** *The graph $\mathcal{G}$ among the whole system ($N$ followers and a leader) contains a directed spanning tree where the leader is acting as the root node.*

It is important to notice that under Assumption-4.17 the matrix $\widetilde{\mathcal{L}}_1$ is non-singular $M-$matrix [21,72], which means $\widetilde{\mathcal{L}}_1$ satisfies the following property, if $\widetilde{\mathcal{L}}_1 X = 0, \forall X \in \mathbb{R}^N$ implies $X = 0$ (see Definition-4.1).

Similar to section-4.4.1, each follower receives a weighted combination of relative outputs between itself and its neighbours via the directed communication graph. Furthermore, all the followers have access to the leader's measured output $\mathrm{a}_{i0} > 0, i = [1, ..., N]$ whose input is zero $u_0 = 0$, differently from the case of section-4.4.1, where only a subset of followers have access to leaders' output and the graph among the followers is undirected. Due to the fact that the graph $\mathcal{G}$ in this section is assumed directed, the property $c_{ij} = c_{ji}$, namely $a_{ij} = a_{ji}$ does no more match in the convergence proof. In the following, the parameters $c_{ij}$ and $c_i$ in (4.11) will be replaced by a one parameter $c_i(t) > 0$ that will be denoted the time-varying coupling weight associated to the $i-th$ follower agent (or node). The protocol given in (4.11) will be modified based on the above finding as,

$$\begin{aligned} u_i &= K_h h_i + K v_i \\ \dot{v}_i &= (A+BK)\, v_i + F\,(c_i+\mu_i) \left[ \sum_{j=1}^{N} a_{ij}\left(\bar{c}_{ij} - y_{ij}\right) + a_{i0}\left(\bar{c}_i - y_{i0}\right) \right] \\ \dot{c}_i &= \left(\bar{c}_i - y_{i0}\right)^T \Gamma \left(\bar{c}_i - y_{i0}\right), i \in [1, N] \end{aligned} \tag{4.25}$$

where $c_i\,(0) > 0$, with $\bar{c}_i$, $\bar{c}_{ij}$, $y_{ij}$, $y_{i0}$ defined in (4.10) and (4.12), and $\mu_i(e_i)$ is a smooth function of $e_i(t)$ that we will determine later. The rest of parameters are the same as in (4.11).

**Theorem 4.18** *[81], [80] The fully distributed TVF tracking problem is solved with Assumptions 4.12, 4.13 and 4.17 under the protocol (4.25) if $(A+BK)$ is Hurwitz, $\Gamma = I$ and $F = -PC^T$, and $\mu_i = e_i P^{-1} e_i$ where $P^{-1} > 0$ is a solution to the LMI (4.15). Furthermore, the coupling weights $c_i(t)$, $\{i\} \in [1, N]$ converge to some finite steady-state value.*

***Proof of Theorem-4.18:*** Similar to the proof of Theorem-4.16, we have to prove that $\lim\limits_{t\to\infty} e_i = 0$ which implies $v_i \to \tilde{x}_i, i = [1, ..., N]$, then from (4.24) $\dot{\tilde{x}}_i = (A+BK)\,\tilde{x}_i - BKe_i$ we get $\lim\limits_{t\to\infty} \tilde{x}_i = 0$ as the matrix $(A+BK)$ is Hurwitz. Let substitute (4.24) into (4.25) yields,

$$\begin{aligned} \dot{e}_i &= Ae_i + FC\,(c_i+\mu_i) \left( \sum_{j=1}^{N} \tilde{l}_{ij} e_i + a_{i0} e_i \right) \\ \dot{c}_i &= e_i^T C \Gamma C e_i \end{aligned} \tag{4.26}$$

where $\tilde{l}_{ij}$ are the entries of $\tilde{\mathcal{L}}_1$. Let define the following Lyapunov candidate function,

$$V_2 = \frac{1}{2} \sum_{i=1}^{N} g_i\,(2c_i + \mu_i)\,\mu_i + \frac{1}{2} \sum_{i=1}^{N} g_i (c_i - \alpha)^2 \tag{4.27}$$

In (4.27), $g_i > 0, i = [1, N]$ and $\alpha$ are positive constants that will be determined later. It results from $c_i\,(0) > 0$ and $\dot{c}_i\,(t) > 0$ that $c_i\,(t) > 0, \forall t \geq 0$. Moreover, we have by definition $\mu_i(t) > 0$, thus we get $V_2$ is positive definite. Let now, compute its derivative along time,

$$\dot{V}_2 = \sum_{i=1}^{N} \left[ g_i \left( 2c_i + \mu_i \right) \dot{\mu}_i + g_i \mu_i \dot{c}_i + g_i \left( c_i - \alpha \right) \dot{c}_i \right] \tag{4.28}$$

Let $\hat{c} = [c_1, \ldots, c_N]^T$, $\hat{\mu} = [\mu_1, \ldots, \mu_N]^T$ and $\mathcal{D} = diag\{a_{10}, \ldots, a_{N0}\}$. We rewrite (4.28) in compact form using Kronecker product,

$$\dot{V}_2 = e^T \left[ \begin{array}{l} G\left(\hat{c} + \hat{\mu}\right) \otimes \left(P^{-1}A + A^T P^{-1}\right) + \left(\hat{c} + \hat{\mu}\right) \left( G\widehat{\mathcal{L}}_1 + \widehat{\mathcal{L}}_1^T G \right) \left(\hat{c} + \hat{\mu}\right) \\ \otimes P^{-1}FC + G\left(\hat{c} + \hat{\mu} - \alpha I\right) \otimes C^T \Gamma C \end{array} \right] e \tag{4.29}$$

where $e = [e_1, \ldots, e_N]^T$ and $\widehat{\mathcal{L}}_1 = \widetilde{\mathcal{L}}_1 + \mathcal{D}$ with $\widetilde{\mathcal{L}}_1$ is the Laplacian matrix that correspond to the subgraph $\widetilde{\mathcal{G}}_1$ among followers agents. All the eigenvalues of the matrix $\widehat{\mathcal{L}}_1$ are positive real parts since the whole graph $\widetilde{\mathcal{G}}$ satisfy Assumption-4.17 and $D > 0$ implying that $\widehat{\mathcal{L}}_1$ is non-singular M-matrix [21,72]. Relying on the above finding, and using Lemma-4.2, there exist $G = diag\left\{g_1, \ldots, g_N\right\} > 0$ such that $G\widehat{\mathcal{L}}_1 + \widehat{\mathcal{L}}_1^T G \geq \lambda_0 I$ where $\lambda_0$ is the smallest eigenvalue of $G\widehat{\mathcal{L}}_1 + \widehat{\mathcal{L}}_1^T G$. Using this result and $F = -PC^T$, $\Gamma = \mathrm{I}$, (4.29) turns in the following inequality,

$$\dot{V}_2 \leq e^T \left[ \begin{array}{l} G\left(\hat{c} + \hat{\mu}\right) \otimes \left(P^{-1}A + A^T P^{-1}\right) - \lambda_0 (\hat{c} + \hat{\mu})^2 \\ \otimes C^T C + G\left(\hat{c} + \hat{\mu} - \alpha I\right) \otimes C^T C \end{array} \right] e \tag{4.30}$$

In the other side, by using Lemma-4.3, yields the following inequality,

$$e^T \left[ G\left(\hat{c} + \hat{\mu}\right) \otimes C^T C \right] e \leq e^T \left[ \left( \frac{\lambda_0}{2} (\hat{c} + \hat{\mu})^2 + \frac{G^2}{2\lambda_0} \right) \otimes C^T C \right] e \tag{4.31}$$

Now, substituting (4.31) into (4.30) yields,

$$\dot{V}_2 \leq e^T \left[ \begin{array}{l} G\left(\hat{c} + \hat{\mu}\right) \otimes \left(P^{-1}A + A^T P^{-1}\right) \\ -\left( \frac{\lambda_0}{2} (\hat{c} + \hat{\mu})^2 - \frac{G^2}{2\lambda_0} + \alpha G \right) \otimes C^T C \end{array} \right] e \tag{4.32}$$

We choose $\alpha \geq \max\limits_{i \in [1,N]} \frac{5g_i}{\sqrt{2\lambda_0}}$ into (4.33) and using the LMI (4.15) yields,

$$\dot{V}_2 \leq e^T \left[ G\left(\hat{c} + \hat{\mu}\right) \otimes \left( P^{-1}A + A^T P^{-1} - 2C^T C \right) \right] e \leq 0 \tag{4.33}$$

Since $\dot{V}_2 \leq 0$, then $V_2$ and $c_i(t)$ are bounded. Each coupling weight $c_i(t)$ increases monotonically and converges to some finite value finally. Similar to the proof of

Theorem-4.16, $V_2 \equiv 0$ is equivalent to $e = 0$. By LaSalle's Invariance principle [74], it turns out that $\lim_{t \to \infty} e(t) = 0$, implying $v(\mathrm{t}) \to \tilde{x}(t)$ as $t \to \infty$. As we have $(A + BK)$ is Hurwitz, it follows from (4.24) that $\lim_{t \to \infty} \tilde{x}_i = 0, i = [1, N]$. The proof is completed.

**Remark 4.19** *It is important to highlight in protocol* (4.25) *that each follower agent exchange with its neighbours the relative measurement* $[(y_i - Ch_i) - (y_j - Ch_j)]$ *of a cardinal q and the relative observer state* $(v_i - v_j)$ *of cardinal n. Moreover, all the followers require the knowledge of their relative output measurement with the leader* $y_{i0} = y_i - y_0$*. The above finding result in stringent communication constraint and increase the communication burden heavily.*

### 4.4.3 Directed output TVF tracking with partial access to leader and reduced information exchange

In this section, the goal is to solve the Leader-Follower output TVF tracking problem with partial access to leader's output whose input is zero and satisfying reduced network information exchange among the agents. We assume that the interaction topology is time-invariant. We propose the following distributed protocol [83],

$$\begin{aligned} u_i &= K_h h_i + K v_i \\ \dot{v}_i &= (A + BK)\, v_i + F(c_i + \mu_i)\ \tilde{\pi}_i \\ \dot{c}_i &= \tilde{\pi}_i^T \Gamma \tilde{\pi}_i, i \in [1, N] \end{aligned} \tag{4.34}$$

where $K$, $F$, $\Gamma$ are feedback matrices defined in (4.11) and (4.25) to be determined later. $K_h$ is defined in (4.6). $v_i \in \mathbb{R}^n$ is a distributed observer (the i$-th$ agent internal protocol state) introduced to estimate the formation tracking error $\tilde{x}_i$. $c_i(t)$ is the adaptive coupling weight associated to the $i$-th agent with $c_i(0) > 0$, and $\mu_i(\varrho_i)$ to be determined later, is a smooth and monotonically increasing function with regard to signal $\varrho_i$. $\tilde{\pi}_i(t)$ is the only available network information for the $i-$th agent from its neighbours, synthesized as a single signal, containing the neighbouring output measurements $y_j$ and desired output offset $Ch_j$, the neighbouring distributed observer outputs $Cv_j$ and eventually the leader's output $y_0$ if agent $i$ has access to leader,

$$\tilde{\pi}_i = \bar{v}_i - \bar{y}_i \tag{4.35}$$

where $\bar{v}_i$ and $\bar{y}_i$ are the available network measurement and the neighbouring network protocol outputs respectively,

$$\bar{v}_i = \sum_{j=1}^{N} a_{ij} C \left( v_i - v_j \right) + a_{i0} C v_i \tag{4.36}$$

$$\bar{y}_i = \sum_{j=1}^{N} a_{ij} \left( (y_i + Ch_i) - \left( y_j + Ch_j \right) \right) + a_{i0} \left( y_i - y_0 + Ch_i \right) \tag{4.37}$$

The term $a_{i0}$ satisfies $a_{i0} > 0$ if the i-*th* agent can get information from the leader and $a_{i0} = 0$ otherwise. Note in (4.37) that it is valid to write $\sum\limits_{j=1}^{N} a_{ij} \equiv \sum\limits_{j \in N_i} a_{ij}$.

**Remark 4.20** *It is important to note from* (4.34) *and* (4.35)*, that the i-th agent receives from each j neighbour agent only a vector of cardinal p and further, it has been shown that only a subset of followers can receive the leader's measured output via the term $a_{i0}$.*

Let define the following signal,

$$\varrho_i = \sum_{j=1}^{N} a_{ij} \left( v_i - v_j \right) + a_{i0} v_i - \sum_{j=1}^{N} a_{ij} \left( \tilde{x}_i - \tilde{x}_j \right) - a_{i0} \tilde{x}_i \tag{4.38}$$

It can be verified that $C\varrho_i = \bar{v}_i - \bar{y}_i$. We denote $v = \left[ v_1^T, ..., v_N^T \right]^T$, $\tilde{x} = \left[ \tilde{x}_i^T, ..., \tilde{x}_N^T \right]^T$, $\bar{v} = [\bar{v}_1^T, ..., \bar{v}_N^T]^T$, $\bar{y} = \left[ \bar{y}_1^T, \ldots, \bar{y}_N^T \right]^T$ and $\varrho = \left[ \varrho_1^T, .., \varrho_N^T \right]^T$, then by using the Kronecker product we have,

$$\begin{cases} \bar{v} = \left[ \widetilde{\mathcal{L}}_1 \otimes C \right] v \\ \bar{y} = \left[ \widetilde{\mathcal{L}}_1 \otimes C \right] \tilde{x} \end{cases} \tag{4.39}$$

where $\widetilde{\mathcal{L}}_1$ is defined in (4.8), and since $C\varrho_i = \bar{v}_i - \bar{y}_i$ then, from (4.39) we have $\left[ I_N \otimes C \right] \varrho = \left( \widetilde{\mathcal{L}}_1 \otimes C \right) (v - \tilde{x})$, therefore we obtain,

$$\varrho = \left( \widetilde{\mathcal{L}}_1 \otimes I_n \right) (v - \tilde{x}) \tag{4.40}$$

In (4.40), notice that $\varrho = 0$ implies that $v = \tilde{x}$ since we have $\tilde{\mathcal{L}}_1$ is a non-singular $M-$matrix.

**Algorithm 4.21** *the following steps show that the application of the distributed adaptive protocol* (4.34) *on the MASs defined in* (4.4) *to solve the output TVF Tracking problem under reduced network information exchange and partial access to leader's output:*

- *Step-1):* choosing a desired formation shape described by $h(t)$, in such a way being compatible with the agents dynamics by satisfying (4.6), therefore a matrix $K_h$ could be extracted. The matrix $K_h$ expands the feasible desired TVF shape, where the formation shape becomes Time-Invariant if $A + BK_h = 0$.

- *Step-2):* choose a feedback matrix $K$ is such away $A + BK$ is Hurwitz. The existence of the matrix $K$ relies on Assumption-4.12.

- *Step-3):*choose the matrices $\Gamma = \mathrm{I}_q$ and $F = -Q^{-1}C^T$, where the matrix $Q \in \mathbb{R}^{n\times n}$ satisfies the following Linear Matrix Inequality LMI,

$$\mathcal{W} = QA + A^TQ - 2C^TC < 0 \tag{4.41}$$

  In (4.41), $Q$ is symmetric positive definite matrix ($Q > 0$). It is important to note that the existence of the matrix $Q$, solution of the LMI (4.41), relies on Assumption-4.13, as it is demonstrated in [20].

- *Step-4):*choose $\mu_i = \varrho_i^T Q \varrho_i$, where $C\varrho_i = \tilde{\pi}_i = \bar{v}_i - \bar{y}_i$. Note that $\mu_i, i = [1, N]$ is a positive scalar.

The proposed Leader-Followers TVF tracking protocol (4.34) is illustrated in diagram form in Fig. 4.4.

**Theorem 4.22** *[83] Based on Algorithm-4.21, the protocol* (4.34) *solves in a fully distributed fashion, the output-feedback TVF Tracking problem under Assumptions* (4.12), (4.13) *and* (4.17), *satisfying reduced network information exchange and partial access to leader's measured output. In addition, the leader's input is assumed zero and the coupling weight $c_i(t) = \varrho_i^T C^T \Gamma C \varrho_i$, $i = [1, \ldots N]$ converge to a finite-steady value.*

***Proof of Theorem-4.22:*** To prove that the protocol given in (4.34) solves the output

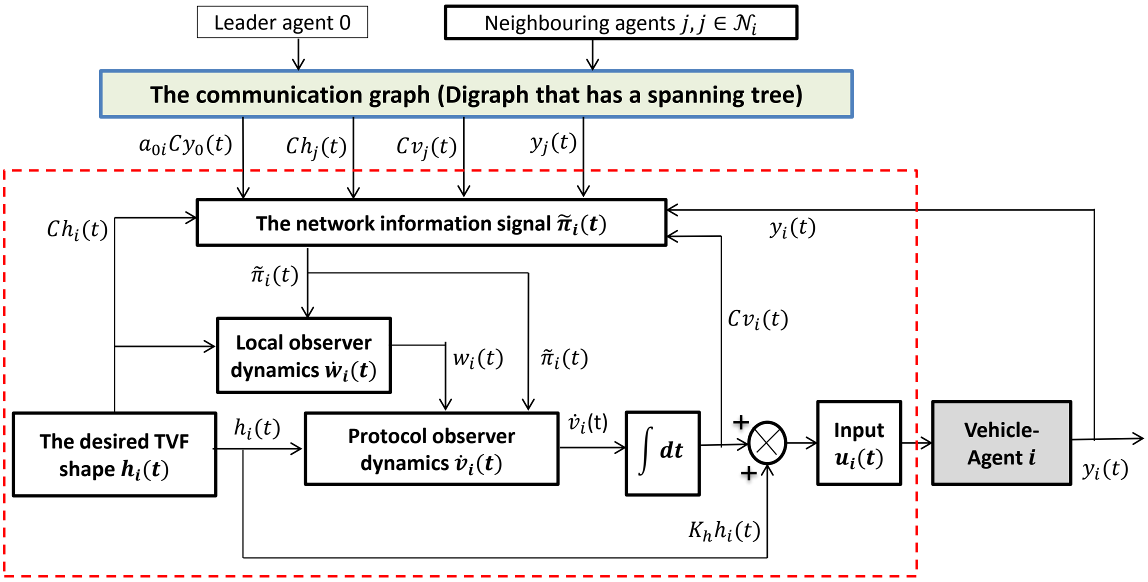

Leader agent 0
Neighbouring agents $j, j \in \mathcal{N}_i$
The communication graph (Digraph that has a spanning tree)
$a_{0i}Cy_0(t)$
$Ch_j(t)$
$Cv_j(t)$
$y_j(t)$
$Ch_i(t)$
The network information signal $\tilde{\pi}_i(t)$
$y_i(t)$
$\tilde{\pi}_i(t)$
$Cv_i(t)$
Local observer dynamics $\dot{w}_i(t)$
$w_i(t)$
$\tilde{\pi}_i(t)$
The desired TVF shape $h_i(t)$
$h_i(t)$
Protocol observer dynamics $\dot{v}_i(t)$
$\dot{v}_i(t)$
$\int dt$
+
+
Input $u_i(t)$
Vehicle-Agent $i$
$y_i(t)$
$K_h h_i(t)$

Figure 4.4: Bloc diagram of the proposed TVF tracking protocol (4.34).

TVFT problem defined in (see Definition-4.10), we have to demonstrate the following two implications (i), (ii) as,

$$\left.\begin{array}{c}\lim\limits_{t\to\infty} \varrho = 0 \Rightarrow (v - \tilde{x}) \to 0, (i) \\ \lim\limits_{t\to\infty} v = 0, (ii)\end{array}\right\} \Rightarrow \lim\limits_{t\to\infty} \tilde{x} \to 0$$

. Using (4.4), (4.6), (4.38) and (4.34), the derivative of $\tilde{x}_i$ and $\varrho_i$ are written as,

$$\begin{cases} \dot{\tilde{x}}_i = A\tilde{x}_i + BKv_i, i = [1, \dots N] \\ \dot{\varrho}_i = A\varrho_i + FC\left(\sum\limits_{j=1}^{N} \widetilde{l}_{ij}(c_j + \mu_j)\varrho_j + a_{i0}(c_i + \mu_i)\varrho_i\right) \end{cases} \tag{4.42}$$

where $\widetilde{l}_{ij}$ are the entries of $\widetilde{\mathcal{L}}_1$. Therefore, from (4.39), (4.40), we can rewrite (4.42) in a compact form as follows,

$$\begin{cases} \dot{\varrho} = \left(\widetilde{\mathcal{L}}_1 \otimes I_n\right)(\dot{v} - \dot{\tilde{x}}_i) \\ \dot{v} = [I_N \otimes (A + BK)]\, v + [(\hat{c} + \hat{\mu}) \otimes FC]\, \varrho \\ \dot{\tilde{x}} = [I_N \otimes A]\, \tilde{x} + [I_N \otimes BK]\, v \end{cases} \tag{4.43}$$

where $\hat{c} = diag\{c_1, \dots, c_N\}$ and $\hat{\mu} = diag\{\mu_1, \dots, \mu_N\}$ having positive entries. Then, it is straightforward to check from (4.43) that,

$$\dot{\varrho} = \left[I_N \otimes A + \widetilde{\mathcal{L}}_1 (\hat{c} + \hat{\mu}) \otimes FC\right] \varrho \tag{4.44}$$

Let $V_3$ be a candidate Lyapunov function defined as,

$$V_3(t) = \frac{1}{2}\sum_{i=1}^{N} g_i\left(2c_i + \mu_i\right)\mu_i + \frac{1}{2}\sum_{i=1}^{N} g_i {d_i}^2 \tag{4.45}$$

In (4.45), $g_i > 0, i = [1, \dots, N]$ could be chosen such as $[g_1, \dots, g_N]^T = \widetilde{L}_1^{-1} 1_N$ (see Lemma-4.2), $d_i = c_i - \delta$ with $\delta > 0$ is a constant that will be determined later. Let $\hat{d} = diag\left\{d_1, \dots, d_N\right\}$. In the other side, from (4.34), $\dot{c}_i(t) \geq 0$, $c_i(0) > 0$, then $c_i(t) > 0, \forall t \geq 0$. Noting further from *Algorithm-4.21*, $\mu_i = \varrho_i^T Q \varrho_i \geq 0$, then we have $V_3(t)$ is positive definite function with respect to $\hat{d}$ and $\varrho$. The derivative of $V_3(t)$ with respect to time,

$$\dot{V}_3(t) = \sum_{i=1}^{N} g_i\left(c_i + \mu_i\right)\dot{\mu}_i + \sum_{i=1}^{N} g_i\left(c_i + \mu_i - \delta\right)\dot{c}_i \tag{4.46}$$

By substituting $\mu_i$, $\dot{c}_i = \varrho_i^T C^T \Gamma C \varrho_i$ and $\dot{\mu}_i = 2{\varrho_i}^T P \dot{\varrho}_i$ from (4.34), (4.42) into (4.46), we get,

$$\begin{aligned}\dot{V}_3\left(t\right) = &\sum_{i=1}^{N} g_i\left(c_i + \mu_i\right) 2\varrho_i^T Q[A\varrho_i + \sum_{j=1}^{N} \widetilde{l}_{ij} FC\left(c_j + \mu_j\right)\varrho_j + a_{i0} FC\left(c_i + \mu_i\right)\varrho_i] \\ &+ \sum_{i=1}^{N} g_i\left(c_i + \mu_i - \delta\right)\varrho_i^T C^T \Gamma C \varrho_i.\end{aligned} \tag{4.47}$$

We have $2\varrho_i^T PA\varrho_i = \varrho_i^T PA\varrho_i + \varrho_i^T A^T P\varrho_i$, then rewrite (4.47) in compact form,

$$\dot{V}_3\left(t\right) = \varrho^T \left[\begin{array}{l} G\left(\hat{c} + \hat{\mu}\right) \otimes \left(QA + A^T Q\right) + \left(\hat{c} + \hat{\mu}\right) 2G\widetilde{L}_1\left(\hat{c} + \hat{\mu}\right) \otimes QFC \\ +G\left(\hat{c} + \hat{\mu} - \delta I_N\right) \otimes C^T \Gamma C \end{array}\right] \varrho \tag{4.48}$$

It holds that,
$\varrho^T \left(\left(\hat{c} + \hat{\mu}\right)\left[2G\widetilde{\mathcal{L}}_1\right]\left(\hat{c} + \hat{\mu}\right) \otimes PFC\right)\varrho = \varrho^T \left(\left(\hat{c} + \hat{\mu}\right)\left[G\widetilde{\mathcal{L}}_1 + \widetilde{\mathcal{L}}_1^T G\right]\left(\hat{c} + \hat{\mu}\right) \otimes PFC\right)\varrho,$
then substituting $F = -Q^{-1}C^T$, $\Gamma = \mathrm{I}_q$ and using Lemma-4.2, yields the following inequality,

$$\begin{aligned}&\dot{V}_3\left(t\right) \leq \varrho^T [G\left(\hat{c} + \hat{\mu}\right) \otimes \left(QA + A^T Q\right) \\ &-\lambda_0 (\hat{c} + \hat{\mu})^2 \otimes C^T C + G\left(\hat{c} + \hat{\mu} - \delta I_N\right) \otimes C^T C]\varrho\end{aligned} \tag{4.49}$$

where $\lambda_0 > 0$ and $G > 0$ are defined in Lemma-4.2. Now, we use Lemma-4.3 with $a = \sqrt{\lambda_0}\left(\hat{c} + \hat{\mu}\right)$, $b = \frac{G}{\sqrt{\lambda_0}}$ and $p = q = 2$, yields

$$\varrho^T\left[G\left(\hat{c} + \hat{\mu}\right) \otimes C^T C\right]\varrho \leq \varrho^T \left[\left(\frac{\lambda_0 (\hat{c} + \hat{\mu})^2}{2} + \frac{G^2}{2\lambda_0}\right) \otimes C^T C\right]\varrho \tag{4.50}$$

We substitute (4.50) into (4.49),

$$\dot{V}_3(t) \leq \varrho^T \left[ G(\hat{c}+\hat{\mu}) \otimes \left( QA + A^T Q \right) - \left( \frac{\lambda_0}{2} (\hat{c}+\hat{\mu})^2 - \frac{G^2}{2\lambda_0} + \delta G \right) \otimes C^T C \right] \varrho \quad (4.51)$$

In (4.51), we know that $a + b \geq 2\sqrt{ab}, \forall a, b \in \mathbb{R}^+$, then by choosing $\delta \geq \left( \frac{5g_s}{2\lambda_0} \right); g_s = Max \{g_i, i \in [1, \ldots, N]\}$ we get,

$$\dot{V}_3(t) \leq \varrho^T \left[ G(\hat{c}+\hat{\mu}) \otimes \mathcal{W} \right] \varrho \leq 0 \quad (4.52)$$

where the matrix $\mathcal{W} < 0$ is given in (4.41), yields that $V_3(t)$ is bounded, and consequently, $\varrho_i$ and $c_i$ are bounded too. Given the fact that $\dot{c}_i(t) \geq 0$, it is straightforward to conclude that $c_i(t), i \in [1, \ldots, N]$ increase monotonically and converge to final values. In the other side, when $\dot{V}_1(t) \equiv 0$ implies that $\varrho = 0$. Therefore, relying on LaSalle's Invariance Principle [74], $\varrho$ converge asymptotically to zero, $\lim_{t\to\infty} \varrho \to 0, (i)$.

It remains now to prove that $\lim_{t\to\infty} v = 0$. From the second line of (4.34), we have $\tilde{\pi}_i = C\varrho_i$ and we know $(A + BK)$ is Hurwitz, and as we got $\lim_{t\to\infty} \varrho \to 0$, we obtain that, $\lim_{t\to\infty} v_i = 0, (ii)$, thus $\lim_{t\to\infty} \tilde{x}_i \to 0, i \in [1, \ldots, N]$, that is the TVF tracking problem 4.22 is solved with protocol (4.34).

**Remark 4.23** *It is worth noting in Step-4 of Algorithm-4.21, that the construction of $\mu_i$ requires the observation of the signal $\varrho_i$ based on the only available network signal $\tilde{\pi}_i = \bar{v}_i - \bar{y}_i = C\varrho_i$. Thus, a local observer $w_i$ is introduced to estimate $\varrho_i$.*

The dynamics of $w_i$ is expressed as,

$$\dot{w}_i = Aw_i + Bu_i - BK_h h_i + F\left[Cw_i - \tilde{\pi}_i\right] \quad (4.53)$$

Now, we have to prove that $w_i \to \varrho_i$ as $t \to \infty$. Let define the local observation error $\tilde{w}_i = w_i - \varrho_i$, and let $w = [w_1^T, ..., w_N^T]^T$, thus the closed loop dynamics of $\tilde{w}_i$ can be written from (4.43) and (4.53) in compact form using the Kronecker product as,

$$\dot{\tilde{w}} = \left[I_N \otimes (A + FC)\right] \tilde{w} + (I_N \otimes BK)\, v - \left[ \widetilde{L}_1 (\hat{c}+\hat{\mu}) \otimes FC \right] \varrho \quad (4.54)$$

The goal is to prove that $\tilde{w} \to 0$ as $t \to \infty$. It follows from the LMI (4.41) with $F = -Q^{-1}C^T$ that,

$$(A + FC)^T Q + Q(A + FC) = A^T Q + QA - 2C^T C < 0 \quad (4.55)$$

Consequently, the matrix $(A + FC)$ is Hurwitz. In the other side, it has been found before that $\lim_{t\to\infty} \varrho \to 0$ and $\lim_{t\to\infty} v = 0$, therefore, from (4.54), we have $\tilde{w} \to 0 \Rightarrow w_i \to \varrho_i$ as $t \to \infty$.

**Remark 4.24** *It is worth noting that no global information is required on the construction of protocol* (4.34)*, such as (the smallest eigenvalue of the Laplacian matrix or the total number of agents) resulting on a fully distributed formation controller. In contrast to* [48] *where the proposed protocol is valid only for undirected interaction topologies, our proposed protocol deals with directed topologies. Furthermore, the design of the desired formation shape is simple and does not require further feasibility-conditions compared to* [42, 44] *where matrix $B$ is required to be of full rank. An important aspect of protocol* (4.34)*, is the reduced information exchange property, where each agent exchanges with its neighbours only the outputs $Cv_j$, $Ch_j$, $y_j$ of cardinal $q$ (which is in general lower than the state vector cardinal $n$). In* [44] *and* [53]*, the protocol implementation requires either to exchange neighbouring states $x_j$, or the local state observation $w_j$. In* [42, 45] *the distributed and local observers outputs $Cv_j$, $Cw_j$ are needed to implement the protocol, while in our proposed protocol* (4.34)*, the local observers outputs are not need in the protocol implementation. In addition, we deal in our work with Leader-follower formation tracking where the communication topology is required to have a directed spanning tree rooted by the leader, while the frameworks in* [44, 45] *deal with formation stabilization (i.e., Leaderless) which is less challenging task, and further the communication topology is assumed to be strongly connected, which is a severe condition compared to the requirement of the existence of a directed spanning tree. Based on the the above facts, and in contrast to the mentioned frameworks, the proposed protocol* (4.34) *is more practical in real applications, exhibits a light interaction burden, and yields a reduction of the network information exchange among the networked agents.*

## 4.5 Distributed TVF tracking under bounded/unknown leader's input

In the previous section (Sec.4.4.3), the TVF tracking problem is solved under reduced network information exchange but with the assumption of a leader with zero control input. This imposes severe limitation on the leader's movement, and therefore to the formation trajectory. In real applications, the final formation trajectory among MASs needs to be regulated. In this section, the leader's input is nonzero, bounded and unknown. Thus, the leader is real and able to implement any actions control (trajectories). This is more challenging than the case of zero control input $u_0(t) = 0$. In addition, $u_0(t)$ will not be accessible to any follower.

**Assumption 4.25** *the control input of the leader is considered unknown and satisfies,* $||u_0(t)|| < e_0$*, with* $e_0$ *being a positive constant scalar.*

To solve the output TVFT problem for Leader-Follower MASs consisting of $N$ followers and one leader under Assumption-4.25 in a fully distributed fashion, we expand the protocol (4.34) to the following protocol [83],

$$\begin{aligned} u_i &= K_h h_i + K v_i - \alpha \mathcal{X}\left(B^T S \zeta_i\right) \\ \dot{v}_i &= (A + BK) v_i + F(c_i + \mu_i) \tilde{\pi}_i - \alpha B\left[\mathcal{X}\left(B^T S \zeta_i\right) - \mathcal{X}\left(B^T P \varrho_i\right)\right] \\ \dot{c}_i &= \tilde{\pi}_i^T \Gamma \tilde{\pi}_i, i \in [1, N] \end{aligned} \tag{4.56}$$

where $S \in \mathbb{R}^{n \times n} > 0$, $\alpha$ is positive scalar that will be determined later, $c_i(0) \geq 1$, the signal $\zeta_i, i = [1 \ldots N]$ and the nonlinear function $\mathcal{X}(.)$ are defined as,

$$\zeta_i = \sum_{j=1}^{N} a_{ij}\left[(x_i - h_i) - (x_j - h_j)\right] + a_{i0}(x_i - x_0 - h_i) \tag{4.57}$$

$$\mathcal{X}(z) = \begin{cases} \frac{z}{\|z\|}, if \rightarrow z \neq 0, \\ 0, if \rightarrow z = 0 \end{cases} \tag{4.58}$$

The rest of the parameters in (4.56) are the same as in (4.34). We present the following **Theorem** to solve the directed output TVFT problem for a Leader-Follower MASs with a real leader whose input is unknown and bounded.

**Remark 4.26** *comparing the protocol (4.34) with protocol (4.56), it can be seen that two new terms on the form of switching (discontinuous) function have been added to $u_i$ and $v_i$ to compensate the non-zero leader's input, thus $u_0(t)$ is viewed as an external perturbation. The design of protocol (4.56) is partly inspired from [51],[52].*

The proposed Leader-Followers TVF tracking protocol (4.56) is illustrated in diagram form in Fig. 4.5.

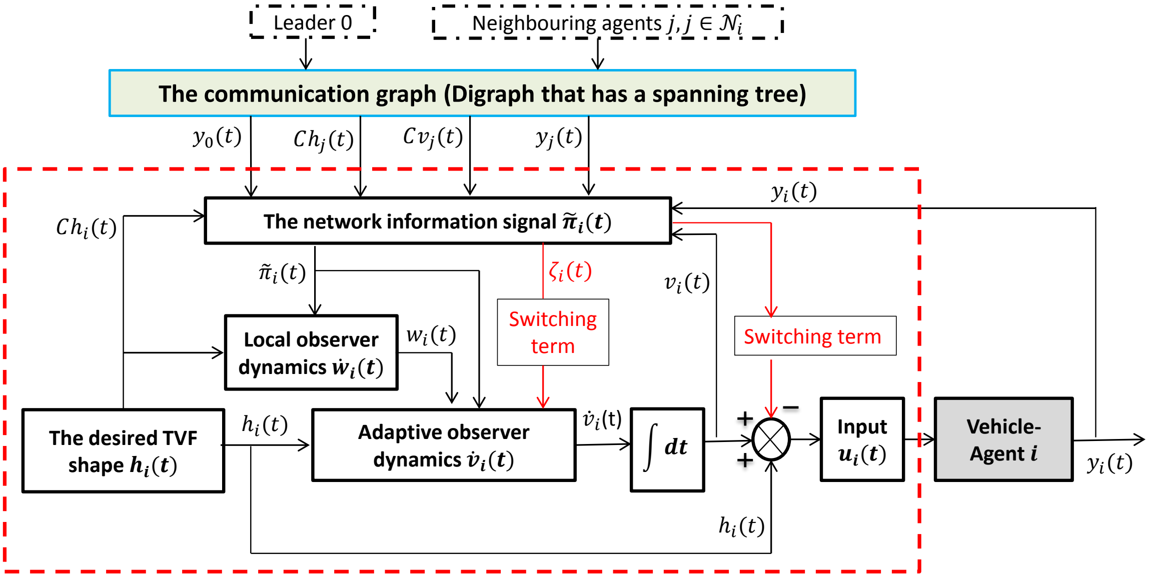

Leader 0
Neighbouring agents $j, j \in \mathcal{N}_i$
The communication graph (Digraph that has a spanning tree)
$y_0(t)$
$Ch_j(t)$
$Cv_j(t)$
$y_j(t)$
$y_i(t)$
$Ch_i(t)$
The network information signal $\tilde{\pi}_i(t)$
$\tilde{\pi}_i(t)$
$\zeta_i(t)$
$v_i(t)$
Switching term
Switching term
Local observer dynamics $\dot{w}_i(t)$
$w_i(t)$
The desired TVF shape $h_i(t)$
$h_i(t)$
Adaptive observer dynamics $\dot{v}_i(t)$
$\dot{v}_i(t)$
$\int dt$
+
−
+
Input $u_i(t)$
Vehicle-Agent $i$
$y_i(t)$
$h_i(t)$

Figure 4.5: Bloc diagram of the proposed L-F TVF tracking protocol (4.56).

**Theorem 4.27** *[83] The output-feedback TVFT problem can be solved using the fully distributed discontinuous protocol (4.56) under assumptions (4.12), (4.13), (4.17) and (4.25); where the parameters K, F, $\Gamma$ and $\mu_i$ are designed as in Theorem-4.22, with $\alpha \geq e_0$ and $Q > 0$, $S > 0$ are respectively solutions to the LMI (4.41) and the following LMI respectively,*

$$\overline{W} = S(A + BK) + (A + BK)^T S < 0 \tag{4.59}$$

*Furthermore, the coupling weights $c_i(t), i = [1, ..., N]$ ultimately converge to some finite values with $c_i(0) \geq 1$.*

***Proof of Theorem-4.27:*** Recall the protocol (4.56), we let $\mathcal{Y}_i = B^T S \zeta_i$ and $\tilde{\mathcal{Y}}_i = B^T P \varrho_i$ yields,

$$
\begin{aligned}
u_i &= K_h h_i + K v_i - \alpha \mathcal{X}\left(\mathcal{Y}_i\right) \\
\dot{v}_i &= \begin{cases} \left(A + BK\right) v_i + F\left(c_i + \mu_i\right) \tilde{\pi} \\ \quad -\alpha B \left[\mathcal{X}\left(\mathcal{Y}_i\right) - \mathcal{X}\left(\tilde{\mathcal{Y}}_i\right)\right] \end{cases} \\
\dot{c}_i &= \tilde{\pi}_i^T \Gamma \tilde{\pi}_i, i \in [1, N]
\end{aligned} \tag{4.60}
$$

From (4.57), we denote $\zeta = [\zeta_1^T, \ldots, \zeta_N^T]$, then we can write in compact form,

$$
\zeta = \left[\tilde{\mathcal{L}}_1 \otimes I_n\right] \tilde{x} \tag{4.61}
$$

The goal is to prove that $\zeta \rightarrow 0 \Rightarrow \tilde{x} \rightarrow 0$ as $t \rightarrow \infty$, since the matrix $\tilde{\mathcal{L}}_1$ is non-singular $M$-Matrix.

First, let rewrite in compact form the signals $\dot{\varrho}$ and $\dot{\zeta}$ by differentiating with respect to time, the signals $\varrho$ and $\zeta$ from (4.40) and (4.61) respectively as,

$$
\dot{\varrho} = \left[I_N \otimes A + \tilde{\mathcal{L}}_1 \left(\hat{c} + \hat{\mu}\right) \otimes FC\right] \varrho - \left(\tilde{\mathcal{L}}_1 \otimes B\right) \left(\alpha \mathcal{X}\left(\tilde{\mathcal{Y}}\right) - 1_N \otimes u_0\right) \tag{4.62}
$$

$$
\dot{\zeta} = \left[I_N \otimes \left(A + BK\right)\right] \zeta + \left(I_N \otimes BK\right) \varrho - \left(\tilde{\mathcal{L}}_1 \otimes B\right) \left[\alpha \mathcal{X}\left(\mathcal{Y}\right) + 1_N \otimes u_0\right] \tag{4.63}
$$

where, $\mathcal{Y} = \left[\mathcal{Y}_1^T, \ldots, \mathcal{Y}_N^T\right]^T$, and $\tilde{\mathcal{Y}} = \left[\tilde{\mathcal{Y}}_1^T, \ldots, \tilde{\mathcal{Y}}_N^T\right]^T$ and $\mathcal{X}\left(\mathcal{Y}\right) = \left[\mathcal{X}(\mathcal{Y}_1)^T, \ldots, \mathcal{X}(\mathcal{Y}_N)^T\right]^T$.
Let define the following candidate Lyapunov function,

$$
V_4(t) = \sum_{i=1}^{N} \zeta_i^T S \zeta_i + \gamma V_3(t) \tag{4.64}
$$

where $\gamma > 0$ is a positive constant to be determined later and $V_3(t)$ is defined in (4.45). It is straightforward to conclude that $V_4(t)$ is a positive definite with respect to $\hat{d}$, $\varrho$ and $\zeta$.

We want to prove that $\dot{V}_4\left(t\right) \leq 0$, $\forall t > 0$. Firstly, following the same steps as in the proof of Theorem-4.22 to compute the derivative of $V_3(t)$ yields,

$$
\begin{aligned}
\dot{V}_3\left(t\right) &\leq \varrho^T \left[G\left(\hat{c} + \hat{\mu}\right) \otimes W\right] \varrho + \Lambda. \\
\Lambda &= -2\varrho^T \left[G\left(\hat{c} + \hat{\mu}\right) \tilde{\mathcal{L}}_1 \otimes PB\right] \left[-1_N \otimes u_0 + \alpha \mathcal{X}\left(\tilde{\mathcal{Y}}\right)\right]
\end{aligned} \tag{4.65}
$$

Note that $\mathcal{W}$ is defined in (4.41). Now, let simplify $\Lambda$. Based on the Laplacian matrix properties $\widetilde{\mathcal{L}}_1 1_N = -\widetilde{\mathcal{L}}_2$ and $\widetilde{\mathcal{A}}_2 = -\widetilde{\mathcal{L}}_2$, yields $\widetilde{\mathcal{L}}_1 1_N = \widetilde{\mathcal{A}}_2$. By using the above result and relying on assumption-(4.25), we can write the first term of $\Lambda$ as,

$$\begin{aligned} & 2\varrho^T \left[ G\left(\hat{c}+\hat{\mu}\right) \widetilde{\mathcal{L}}_1 \otimes PB \right] \left(1_N \otimes u_0(t)\right) \\ & = \sum_{i=1}^{N} \left[ 2g_i\left(c_i+\mu_i\right) \varrho_i^T PB a_{i0} u_0(t) \right] \\ & \leq \sum_{i=1}^{N} 2g_i\left(c_i+\mu_i\right) \left\| B^T P\varrho_i \right\| a_{i0} e_0. \end{aligned} \tag{4.66}$$

Furthermore, using the switching function given in (4.58), useful properties can be found [51],

$$\begin{aligned} & \varrho_i^T PB\mathcal{X}\left(\tilde{\mathcal{Y}}_i\right) = \varrho_i^T PB \tfrac{B^T P\varrho_i}{\left\|B^T P\varrho_i\right\|} = \left\|B^T P\varrho_i\right\| \\ & \varrho_i^T PB\mathcal{X}\left(\tilde{\mathcal{Y}}_j\right) \leq \left\|\varrho_i^T PB\right\| \left\| \tfrac{B^T P\varrho_j}{\left\|B^T P\varrho_i\right\|} \right\| = \left\|B^T P\varrho_i\right\| \end{aligned}$$

Then, knowing that $a_{ij} \geq 0$, the second term of $\Lambda$ can be developed using the above properties as,

$$\begin{aligned} & -2\varrho^T \left[ G\left(\hat{c}+\hat{\mu}\right) L_1 \otimes PB \right] \alpha \mathcal{X}\left(\widetilde{\mathcal{Y}}\right) \\ & = -2\sum_{i=1}^{N} \left\{ g_i\left(c_i+\mu_i\right) \alpha \varrho_i^T PB \left[ a_{i0}\mathcal{X}(B^T P\varrho_i) + \sum_{j=1}^{N} a_{ij}\left(\mathcal{X}(B^T P\varrho_i) - \mathcal{X}(B^T P\varrho_j)\right)\right] \right\} \\ & \leq -2\sum_{i=1}^{N} g_i\left(c_i+\mu_i\right) \left\|B^T P\varrho_i\right\| a_{i0}\alpha \end{aligned} \tag{4.67}$$

Based on the obtained developments (4.66), (4.67) and the fact that $\alpha \geq e_0$ (see theorem-4.27), the term $\Lambda$ in (4.65) satisfies,

$$\Lambda \leq 0 \tag{4.68}$$

Consequently, because $\mathcal{W} < 0$ in (4.41), $\dot{V}_3\left(t\right)$ satisfies,

$$\dot{V}_3\left(t\right) \leq \varrho^T \left[ G\left(\hat{c}+\hat{\mu}\right) \otimes \mathcal{W} \right] \varrho \leq 0 \tag{4.69}$$

Similar to the proof of Theorem-4.22, $V_3\left(t\right)$, $\varrho_i$ and $\mu_i$ are bounded. Knowing that $\dot{c}_i(t) \geq 0$, so $c_i(t), i = [1, N]$ increase and converge to some finite values. We have $\dot{V}_3\left(t\right) \equiv 0$ implies that $\varrho = 0$. Therefore, by LaSalle Invariance Principle [74], $\lim\limits_{t\to\infty} \dot{V}_1\left(t\right) \rightarrow$

0, implies $\lim_{t\to\infty} \varrho \to 0$.

Now, recalling that the goal is to prove that $\zeta \to 0 \Rightarrow \tilde{x} \to 0$ as $t \to \infty$. Based on (4.57) and (4.63), let compute the derivative of $V_4(t)$ in (4.64),

$$\begin{aligned} \dot{V}_4(t) &= \begin{cases} \zeta^T(I_N \otimes \overline{\mathcal{W}})\zeta + 2\zeta^T(I_N \otimes SBK)\varrho \\ +\Lambda_2 + \gamma \dot{V}_3(t) \end{cases} \\ \Lambda_2 &= -2\zeta^T(\widetilde{\mathcal{L}}_1 \otimes SB)(\alpha\mathcal{X}(\mathcal{Y}) + 1_N \otimes u_0(t)) \end{aligned} \tag{4.70}$$

where $\overline{\mathcal{W}} < 0$ is defined in (4.59). Similar to the developments made with $\Lambda$ in (4.66) and (4.67), we can deduce that the two terms of $\Lambda_2$ satisfy,

$$\begin{aligned} -2\zeta^T(\widetilde{\mathcal{L}}_1 \otimes SB)(1_N \otimes u_0(t)) &\leq 2\sum_{i=1}^{N} a_{i0}\left\|B^T S\zeta_i\right\| e_0 \\ -2\zeta^T(\widetilde{\mathcal{L}}_1 \otimes SB)(\alpha\mathcal{X}(\mathcal{Y})) &\leq -2\sum_{i=1}^{N} a_{i0}\left\|B^T S\zeta_i\right\| \alpha \end{aligned}$$

Hence, $\Lambda_2$ satisfies,

$$\Lambda_2 \leq 0 \tag{4.71}$$

In the other side, the second term of $\dot{V}_4(t)$ can be developed using Lemma-4.3 as,

$$2\zeta^T(I_N \otimes SBK)\varrho \leq \begin{cases} \frac{1}{2}\zeta^T[I_N \otimes (-\overline{\mathcal{W}})]\zeta \\ +\frac{2\lambda_{\max}(K^T B^T SSBK)}{\lambda_{\min}(-\overline{\mathcal{W}})}\varrho^T\varrho \end{cases}$$

Now, we substitute (4.69), (4.71) and the above obtained inequality in (4.70) yields,

$$\dot{V}_4 \leq \begin{cases} \frac{1}{2}\zeta^T(I_N \otimes \overline{\mathcal{W}})\zeta + \frac{2\lambda_{\max}(K^T B^T SSBK)}{\lambda_{\min}(-\overline{\mathcal{W}})}\varrho^T\varrho \\ -\gamma\varrho^T\left[G\left(\hat{c} + \hat{\mu}\right) \otimes (-\mathcal{W})\right]\varrho \end{cases} \tag{4.72}$$

We know that $\mu_i(t) \geq 0, c_i(t) \geq 1, \forall t \geq 0$ (see theorem-4.27), thus, the following inequality holds $(\hat{c} + \hat{\mu}) \geq I_N$. Hence, we wan write,

$$\varrho^T\left[G\left(\hat{c} + \hat{\mu}\right) \otimes (-\mathcal{W})\right]\varrho \geq \lambda_{\min}(G)\lambda_{\min}(-\mathcal{W})\varrho^T\varrho,$$

We substitute the above inequality in (4.72) and let choose $\gamma \geq \frac{2\lambda_{\max}(K^T B^T SSBK)}{\lambda_{\min}(-\overline{\mathcal{W}})\lambda_{\min}(G)\lambda_{\min}(-\mathcal{W})} \geq 0$ yields,

$$\dot{V}_4(t) \leq \frac{1}{2}\zeta^T(I_N \otimes \overline{\mathcal{W}})\zeta \leq 0, \tag{4.73}$$

For brevity, the analysis of $\dot{V}_4(t)$ is similar to $\dot{V}_3(t)$ in (4.52) and (4.69). Thus, it is valid to write $\lim_{t\to\infty} \zeta(t) \to 0$, which implies that $\lim_{t\to\infty} \tilde{x}(t) \to 0$ from (4.61) as $\widetilde{\mathcal{L}}_1$ is a non-singular $M-$matrix, that is the proof is completed.

**Remark 4.28** *In protocol (4.34), the TVF tracking convergence rate is depending on the ones of the observers $v_i$ (4.34) and $w_i$ in (4.53); which in its turn depends on the choice of the matrices $K$ and $F$. It result that the TVFT convergence rate depends on the smallest real part of eigenvalues of $A + BK$ and $A + FC$.*
*In the other side, the examination of the derivative of the Lyapunov function (4.52), $\dot{V}_3\left(t\right) \leq \varrho^T \left[G\left(\hat{c} + \hat{\mu}\right) \otimes \mathcal{W}\right] \varrho \leq 0$, shows that the interaction topology among the agents has an effect on the convergence rate of the MASs towards the desired TVF, through the positive diagonal matrix $G = diag[g_1, ..., g_N]$ defined in lemma-4.2 by $[g_1, \ldots, g_N]^T = \widetilde{\mathcal{L}_1}^{-1} 1_N$. Thus, we know from (4.8) that $\widetilde{\mathcal{L}}_1$ depends on $\widetilde{\mathcal{A}}_1$ and $\widetilde{\mathcal{A}}_2$, which implies that the Lyapunov function decreasing speed $\dot{V}_3(t)$ depends on the Laplacian matrix among the MASs $\widetilde{\mathcal{L}}_1$ and the vector $\widetilde{\mathcal{A}}_2$ that defines the agents having access to leader's output. This is because $\mu_i = \varrho_i^T Q \varrho_i$, $c_i(t) = \varrho_i^T C^T \Gamma C \varrho_i$ and the matrix $\mathcal{W}$ in (4.41) is constant. The same conclusion is valid for the controller (4.56) as we have in (4.64), $V_4(t) = \sum\limits_{i=1}^{N} \zeta_i^T S \zeta_i + \gamma V_3(t)$. In other words, the larger are the components of $G$ (depending on the choice of $\widetilde{\mathcal{A}}_1$ and $\widetilde{\mathcal{A}}_2$), the faster is the convergence rate of $\varrho_i$ towards zero, implying the convergence rate of the MASs TVFT error to decrease faster towards zero relying on (4.52). Furthermore, the protocols (4.34) and (4.56) remain valid for a MASs to achieve TVF tracking under switching interaction topologies as long the assumption-4.17 remains satisfied.*

## 4.6 Simulations results

In this section, we will show three examples to demonstrate the conjecture of the theoretical results. In Example-, we show the effectiveness of the fully distributed adaptive TVF tracking controller given in (4.34) under leader-follower scheme, where the leader's input is zero. Then, in Example-2, we show the application of the fully distributed TVF tracking controller given in (4.56) for a group of networked quadrotors to achieve successfully the desired TVF while tracking a target of unknown control

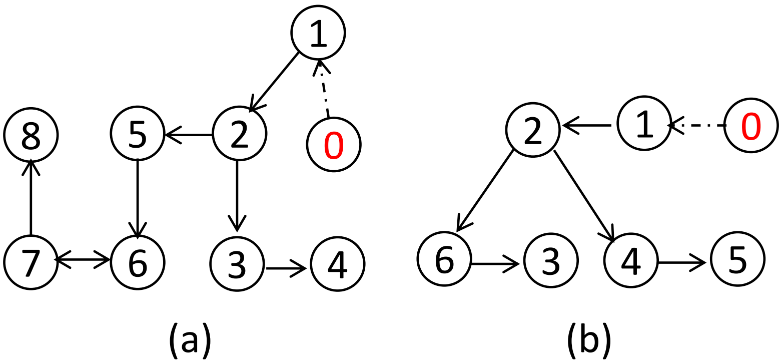


Figure 4.6: Directed interaction topology among agents, (a) example-1 and (b) example-2.

input (i.e., target enclosing behaviour). Finally, in Example-3, we evaluate the validity of the proposed MASs protocols under time-varying interaction topology, and further we analyse the effect of the control input delay and the output measurement delay on the MASs performance and stability.

### Example-1:

we consider a networked *MASs* consisting of nine $(09)$ agents with third order dynamics described as in (4.4),

$$A = \begin{bmatrix} 0 & -1 & 1 \\ 0 & 0 & 0.5 \\ -1 & -1 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

We let $C = [I_2, 0_{2\times 1}]$. It is clear that $(A, B, C)$ is stabilizable and detectable. We consider the agent labelled by 0 the leader and the agents labelled by 1 to 8 the followers. The interaction topology among agents is described by the graph given in Fig. 4.6 a. It can be verified that the interaction topology is not strongly connected. Furthermore, it contains a directed spanning tree, where the leader acts as the root node satisfying assumption-(4.17). Moreover, the leader's output is only accessible to agent 1. The desired TVF shape is given as $h(t) = \left[h_1^T(t), \ldots, h_8^T(t)\right]^T$, with

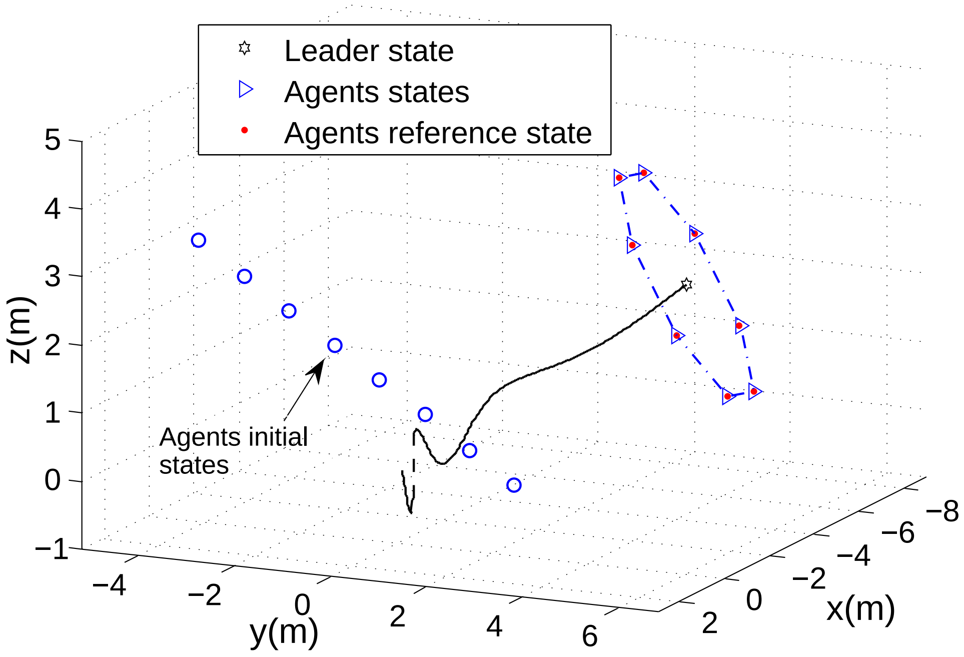


Figure 4.7: The MASs States in 3-D space.

$h_i\,(t)\,, i \in [1, 8]$ is chosen as,

$$h_i\,(t) = \begin{bmatrix} 2\sin\left(\bar{\omega}t + \frac{(i-1)\pi}{4}\right) - 2\cos\left(\bar{\omega}t + \frac{(i-1)\pi}{4}\right) \\ -2\cos\left(\bar{\omega}t + \frac{(i-1)\pi}{4}\right) \\ 2\sin\left(\bar{\omega}t + \frac{(i-1)\pi}{4}\right) \end{bmatrix}$$

It can be verified that $\sum\limits_{i=1}^{8} h_i\,(t) = 0$, which means that the leader will lie inside the desired formation. We set $\bar{\omega}\,(\mathrm{t}) = 1$ in $h(\mathrm{t})$, we can check that the formation constraint (4.5) is satisfied with $u_{h,i}\,(t) = K_h h_i(t)$ and the matrix $K_h$ is found to be $K_h = [1, 0, 0; 0, 0, 0.5]^T$. The formation reference trajectory comes from the leader's state. Since the leader's input $u_0(t)$ is zero in this example, means the leader state $x_0(t)$ is solution of $\dot{x}_0\,(t) = Ax_0(t)$. To get $(A + BK)$ Hurwitz, let assign the following eigenvalues, $[-1.5, -1.5, -2.5]$, thus the matrix $K$ can be obtained using pole placement technique. By solving the LMI (4.41), the matrices $Q$ and $F$ are found to

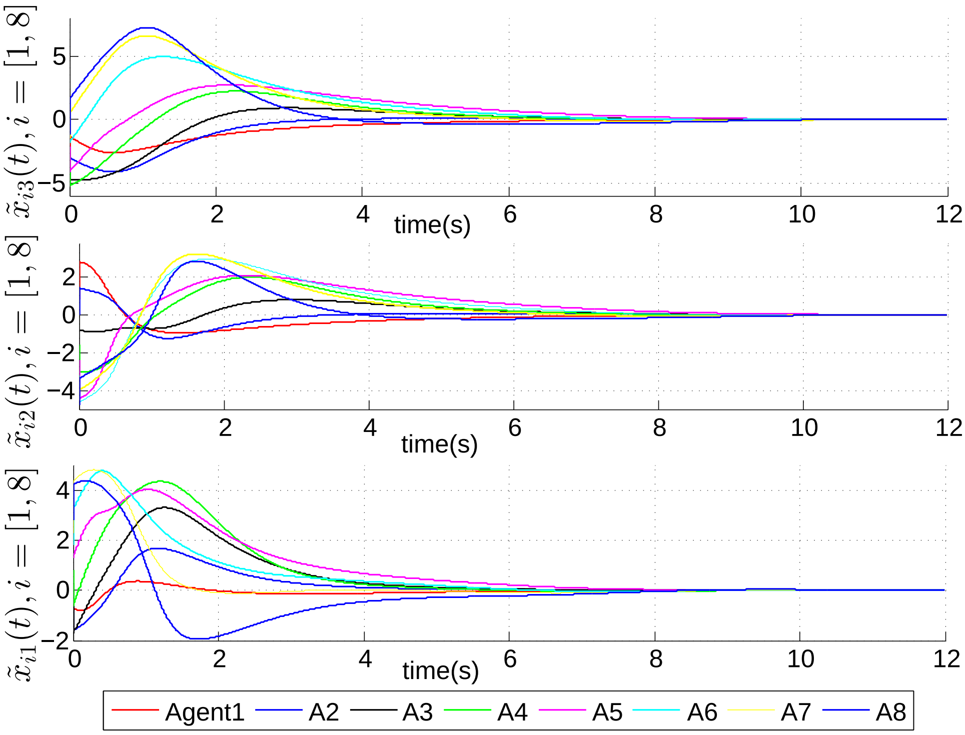


Figure 4.8: The TVF Tracking error $\tilde{x}_i(t) = (x_i - x_0 - h_i), i = [1, ..., 8]$.

be,

$$K = \begin{bmatrix} -0.8750 & 2.2500 & -2.7500 \\ 1.8750 & -2.7500 & 0.7500 \end{bmatrix}; Q = \begin{bmatrix} 1.3644 & 0.3548 & -0.3861 \\ 0.3548 & 1.9571 & -0.5141 \\ -0.3861 & -0.5141 & 1.6998 \end{bmatrix}$$

$$F = \begin{bmatrix} -0.8030 & 0.1061 \\ 0.1061 & -0.5691 \\ -0.1503 & -0.1480 \end{bmatrix}$$

The MASs initial states are set as, $x_i(0) = [-3 + 0.3(i - \eta), 2 - 0.8(i + \eta), -1 + 0.5(i + \eta)]^T$, $i \in [1, 8]$ with $\eta$ a random number between 0 and 1, and the leader's initial state is set $x_0(0) = [0.6, 0.4, 0.2]^T$. We set $c_i(0) = 0.5$, $i \in [1, 8]$

The state trajectories of the MASs in 3-D space are depicted in Fig. 4.7. In Fig. 4.8, the TVF tracking errors $\tilde{x}_i = x_i - x_0 - h_i$ of the followers are shown to converge to zero which implies that the output TVFT problem is solved. The Fig. 4.9 and Fig. 4.10 show the distributed observer state $v_i$ and the local observer tracking error $\tilde{w}_i$ respectively, where it can be seen that its converge to zero. In Fig. 4.11, we see

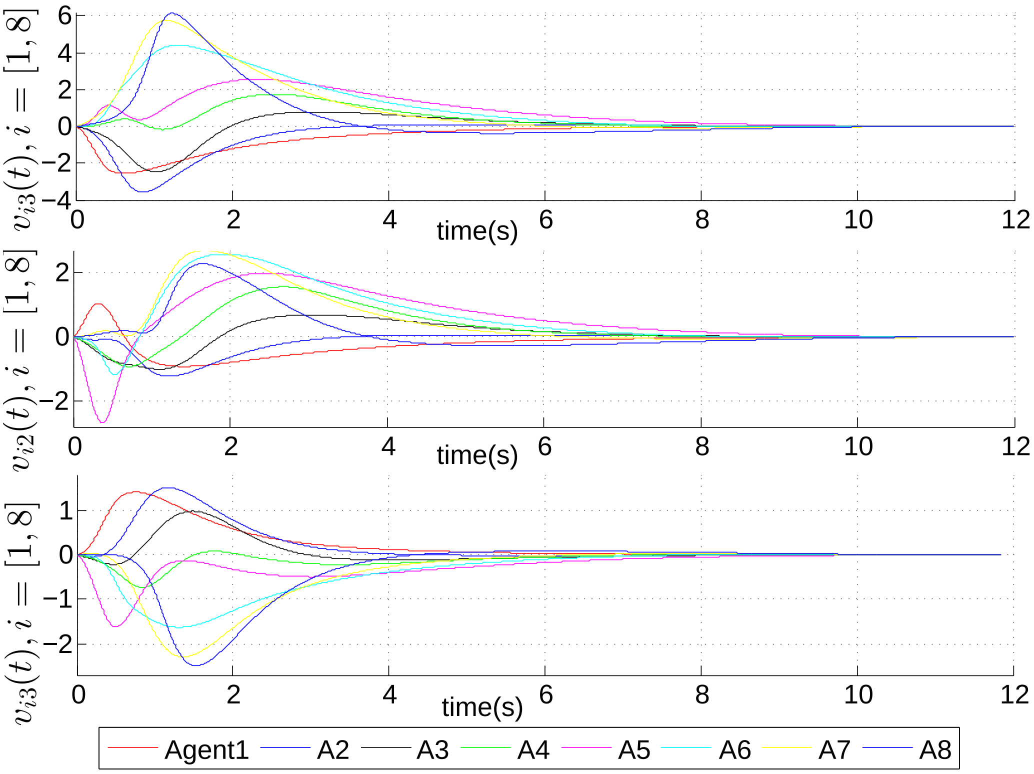


Figure 4.9: Distributed observer state $v_i(t), i = [1, ..., 8]$.

the convergence of the adaptive coupling weight to some final steady-value, which confirms the conjecture of the obtained theoretical results. Finally, in Fig. 4.12, we show the developed control inputs of the six agents.

## Example-2:

In this example, we consider a MASs of six (06) Unmanned Aerial Vehicles (quadrotors) and a target (i.e., quadrotor) labelled by zero. Each UAV's state vector consists of a position coordinates in global frame, attitude angles (roll, pitch, yaw) and their derivatives as $x_{q,i} = [X_i, Y_i, Z_i, \dot{X}_i, \dot{Y}_i, \dot{Z}_i]^T$, $\xi_{q,i} = \left[\varphi_i, \theta_i, \psi_i, \dot{\varphi}_i, \dot{\theta}_i, \dot{\psi}_i\right]^T$, $i \in [1, 6]$ respectively.

The dynamics of the quadrotor can be viewed as a double sub-dynamics in cascade, the rotational dynamics (inner-loop) which is fast and the translational dynamics (outer-loop) which is slow. The control strategy of the networked quadrotors system is divided into two parts. The first part involves the proposed TVF tracking protocol

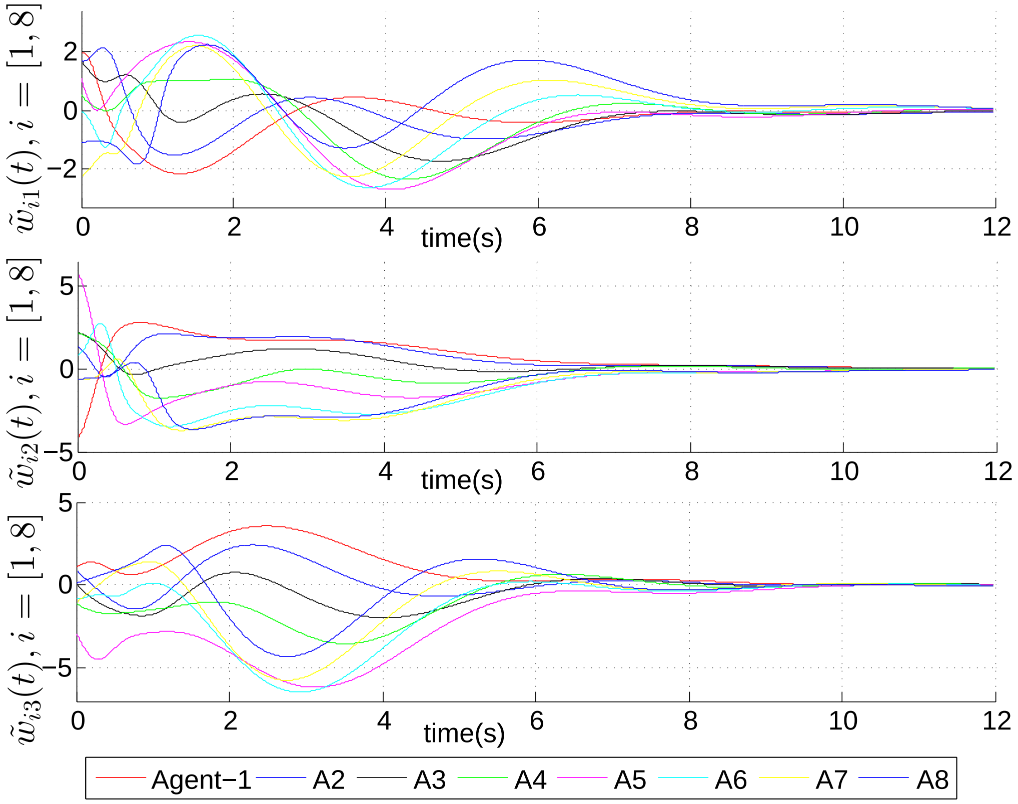


Figure 4.10: Local observer tracking errors $\tilde{w}_i(t) = w_i(t) - \varrho_i(t), i = [1, ..., 8]$.

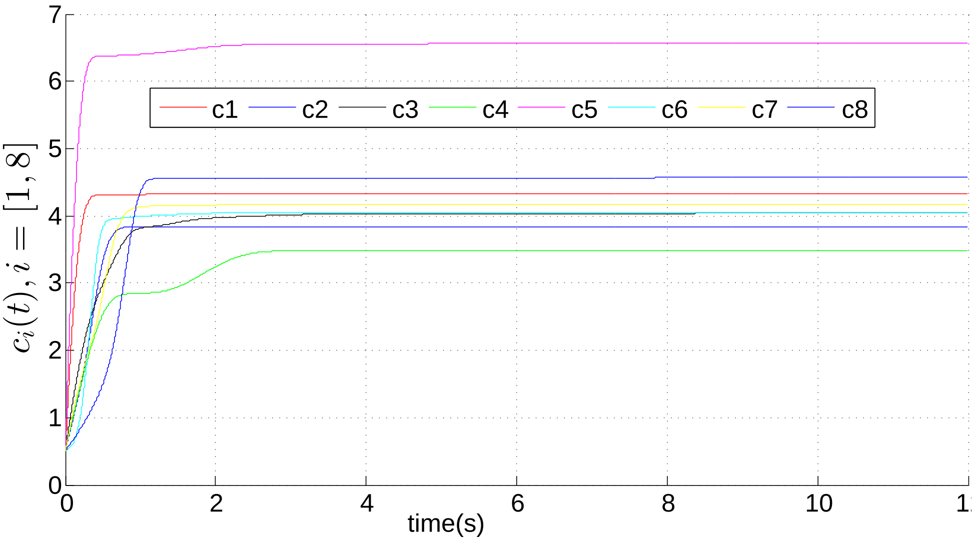


Figure 4.11: Adaptive Coupling weights $c_i(t), , i = [1, ..., 8]$.

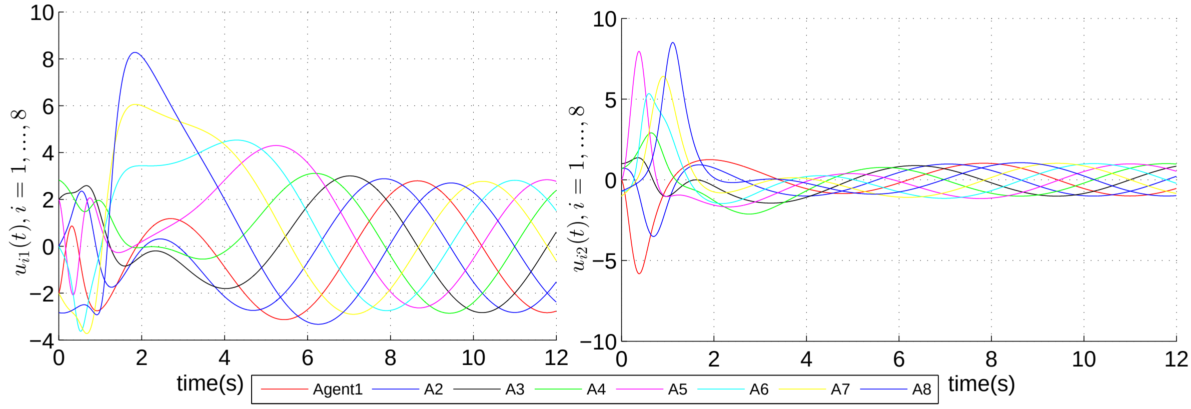


Figure 4.12: Control inputs $u_i(t), , i = [1, ..., 8]$.

(4.56) and will be applied to the translational dynamics to drive the quadrotors to a desired formation while tracking the target trajectory (viewed as a leader). The second control part concern the angles control, where the outputs of the first control-part will feed-forward the second control-part. The translational dynamics of a quadrotor is described by the non-linear equations given in (2.9), [84]. For reading convenience, we rewrite it here,

$$\begin{aligned}
\ddot{X}_i &= \left(\frac{-K_{fx}}{M_q}\right)\dot{X}_i + \frac{u_{x,i}}{M_q}F_i \\
\ddot{Y}_i &= \left(\frac{-K_{fy}}{M_q}\right)\dot{Y}_i + \frac{u_{y,i}}{M_q}F_i \\
\ddot{Z}_i &= \left(\frac{-K_{fz}}{M_q}\right)\dot{Z}_i - g + \frac{\cos\varphi_i \cos\theta_i}{M_q}F_i
\end{aligned} \tag{4.74}$$

where $M_q$ and $F_i$ represent the quadrotor mass and the total thrust respectively, the rest of the parameters are defined in Section-2.2.2. The terms $u_{x,i}$, $u_{y,i}$ are given as,

$$\begin{aligned}
u_{x,i} &= cos\phi_i cos\psi_i sin\theta_i + sin\phi_i sin\psi_i \\
u_{y,i} &= cos\phi_i sin\psi_i sin\theta_i - sin\phi_i sin\psi_i
\end{aligned}$$

Since the attitude angles are weakly varying around zero, we can approximate $\cos\phi_i \sin\theta_i \approx 1$ for the altitude dynamics, thus the control of $Z_i$ is trivial. It is feasible to assume that the attitude angles are fast stabilized due to their fast dynamics, thus, we can write $\phi_i \approx \phi_i^r, \theta_i \approx \theta_i^r, \psi_i \approx \psi_i^r$ [84]. Furthermore, in this work, the formation tracking begins after the altitude dynamics is stabilized, hence, it is possible to approximate $F_i \approx M_q g / cos\phi_i cos\theta_i$ during the formation. Let neglect the aerodynamic friction forces, and we choose the yaw angle to be constant ($\psi_i = 0$). Consequently, the simplified translational dynamics of the quadrotor could be rewritten as [84],

$$\begin{aligned} \ddot{X}_i &= (\tan\theta_i^r)g \\ \ddot{Y}_i &= \frac{-\tan\phi_i^r}{\cos\theta_i^r}g \\ \ddot{Z}_i &= \frac{F_i}{M_q} \end{aligned} \tag{4.75}$$

Now, we know that the quadrotor is driven along x-axis and y-axis by changing the pitch and roll angles. Thus, to transform the nonlinear model (4.74) to a double integrator model, we can choose,

$$\begin{aligned} \theta_i^r &= \tan^{-1}\left(\frac{u_{x,i}}{g}\right) \\ \phi_i^r &= \tan^{-1}\left(\frac{-u_{y,i}\cos\left(\tan^{-1}\frac{u_{x,i}}{g}\right)}{g}\right) \\ \psi_i^r &= 0 \end{aligned} \tag{4.76}$$

where $u_{x,i}$, $u_{y,i}$ are virtual auxiliary control inputs satisfying,

$$\ddot{X}_i = u_{x,i}, \ddot{Y}_i = u_{y,i} \tag{4.77}$$

In this work, the TVFT controller (4.56) will be applied to the $(X, Y)$ quadrotors coordinates and their derivatives $(\dot{X}, \dot{Y})$, while the $Z$ and $\dot{Z}$ states will be controlled apart using a $PID$ controller. Thus, we denote by $x_i = [X_i, Y_i, \dot{X}_i, \dot{Y}_i]^T$ and $u_i = \left[u_{x,i}, u_{y,i}\right]^T$ the state vector and the control input vector of the translation dynamics for each quadrotor along x-axis and y-axis, respectively. It is worth reminding that the translational TVF tracking control input $u_i \in \mathbb{R}^{2\times 1}$ is a feed-forward via (4.76) for the attitude controller, that is chosen to be a classical $PID$ controller. From (4.77), the $x - y$ linear translational state model of each quadrotor can be written as in (4.4) with,

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We choose $C = \begin{bmatrix} I_2 & 0_2 \end{bmatrix}$.

**Remark 4.29** *the followed strategy to linearize the nonlinear translational model is more appropriate and involves less linearization errors than the direct linearization approach based on first-order Taylor approximation.*

The interaction topology among the quadrotor is given in Fig. 4.6 (b), where the agent labelled by zero is the target (viewed as a leader) whose dynamics is supposed to be as in (4.4), and it's control input is unknown and bounded. We can verify that $(A, B, C)$ is stabilizable and detectable. We can check that the interaction topology satisfies assumption-(4.17).

The desired formation shape $h(t)$ will be constructed by three parameters that are: the formation shape $\bar{h}$, the formation scale $r(t)$ and the formation rotating frequency around its center $\bar{\omega}(t)$. For the quadrotor $i, i \in [1,6]$, the $\bar{h}_i = [\bar{h}_{i1}, ..., \bar{h}_{i6}]^T$ can be defined as,

$$\bar{h}_i = \begin{cases} r\sin(\bar{\omega}t + (i-1)\,\pi/3) - r\cos(\bar{\omega}t + (i-1)\,\pi/3) \\ 2r\sin(\bar{\omega}t + (i-1)\,\pi/3) \\ r\bar{\omega}[\cos(\bar{\omega}t + (i-1)\,\pi/3) + \sin(\bar{\omega}t + (i-1)\,\pi/3)] \\ 2r\bar{\omega}\cos(\bar{\omega}t + (i-1)\,\pi/3) \end{cases}$$

Now, different formation shapes (i.e. parallelogram, circle, triangle,...) could be designed as a combination of $\bar{h}_i$. Let $h(t)$ to be designed as,

$$h(t) = \begin{matrix} \left[\bar{h}_1^T, \frac{\bar{h}_1^T+\bar{h}_3^T}{2}, \bar{h}_3^T, \bar{h}_4^T, \frac{\bar{h}_4^T+\bar{h}_6^T}{2}, \bar{h}_6^T\right]^T, 0 \le t \le 15 \\ \left[\bar{h}_1^T, \frac{\bar{h}_1^T+\bar{h}_3^T}{2}, \bar{h}_3^T, \frac{\bar{h}_3^T+\bar{h}_5^T}{2}, \bar{h}_5^T, \frac{\bar{h}_1^T+\bar{h}_5^T}{2}\right]^T, 15 < t \le 30 \\ \left[\bar{h}_1^T, \bar{h}_2^T, \bar{h}_3^T, \bar{h}_4^T, \bar{h}_6^T, \bar{h}_6^T\right]^T, 30 < t \le 45 \end{matrix}$$

From the description of $h(t)$, the desired formation shape is a triangle as $t \in [0s, 15s]$, then a parallelogram shape as $t \in [15s, 30s]$ and as hexagons shape when $t \in [30s, 45s]$. We set $\bar{\omega}(t) = 0.15$, and we set $r(t)$ as,

$$r\,(t) = \begin{cases} 3, t \le 15 \\ 3 + 3e^{-0.18(t-15)}, t > 15 \end{cases}$$

It can be checked that the formation constraint (4.5) is satisfied with $u_{h,i}\,(t) = K_h h_i(t)$, thus the matrix $K_h$ is found to be,

$$K = \begin{bmatrix} -1.60 & 0 & -2.60 & 0 \\ 0 & -1.60 & 0 & -2.60 \end{bmatrix}$$

$$P = \begin{bmatrix} 0.6684 & 0 & -0.6684 & 1 \\ 0 & 0.6684 & 0 & -0.6684 \\ -0.6684 & 0 & 2.0051 & 0 \\ 0 & -0.6684 & 0 & 2.0051 \end{bmatrix}$$

$$S = \begin{bmatrix} 42.8006 & 0 & 13.0449 & 0 \\ 0 & 42.8006 & 0 & 13.0449 \\ 13.0449 & 0 & 16.5852 & 0 \\ 0 & 13.0449 & 0 & 16.5852 \end{bmatrix} ; F = \begin{bmatrix} -2.2443 & 0 \\ 0 & -2.2443 \\ -0.7481 & 0 \\ 0 & -0.7481 \end{bmatrix}$$

We set the initial state vectors of quadrotors as $x_{q,i}(0) = [-2\eta + 2i(-1)^i,\ -3\eta + 2i(-1)^i,\ 0.5 - \eta, 0, 0, 0]^T$, $\xi_{q,i}(0) = [0.2\eta, 0.1\eta, \eta - 0.5, 0, 0, 0]^T$ and $c_i(0) = 1 + 1.5\eta$, $i \in [1, 6]$ with $\eta$ being a random number with uniform distribution between 0 and 1. The target initial state is set to, $x_{q,0}(0) = [3, 4, 10, 0, 0, 0]^T$, and assume that the target's control input is defined as $u_0(t) = [\frac{2}{50}, \frac{3}{50} - \frac{1}{10}e^{-0.2t}, \frac{1}{100} - \frac{1}{100}\sin(\frac{2\pi t}{30})]^T$. Thus, we have $\|u_0(t)\| \leq e_0 = 0.075$. Then, we choose $\alpha$ in (4.56) as $\alpha = 0.08 \geq e_0$.

The used qudarotor's mechanical and electrical parameters in simulation were taken from [58]. It is worth to remind that the derived quadrotor's control inputs relying on (4.56) and the feedback-linearised model (4.77) will be applied to the nonlinear quadrotors model.

Let now analyse the obtained simulation results. In Fig. 4.13, we show the position snapshots of quadrotors during formation achieving at different time instants, where it can be seen that the quadrotros are tracking the target while maintaining time-varying formation with different shapes. Fig. 4.14 illustrates the quadrotors TVF Tracking errors along the x-axis and y-axis $\tilde{x}_i = x_i - x_0 - h_i$, and the tracking errors along the z-axis $\tilde{Z}_i = Z_i - Z_i^r$, while Fig. 4.16 shows the angles tracking errors $\tilde{\xi}_i = \xi_i - \xi_i^r$, $i \in [1, 6]$, where it can be seen that it tends to zero. In Fig. 4.15 and Fig. 4.17, the distributed observer states and the dynamic coupling weights are shown respectively, in which, it is obvious that the coupling weights converge to a finite values. Finally,

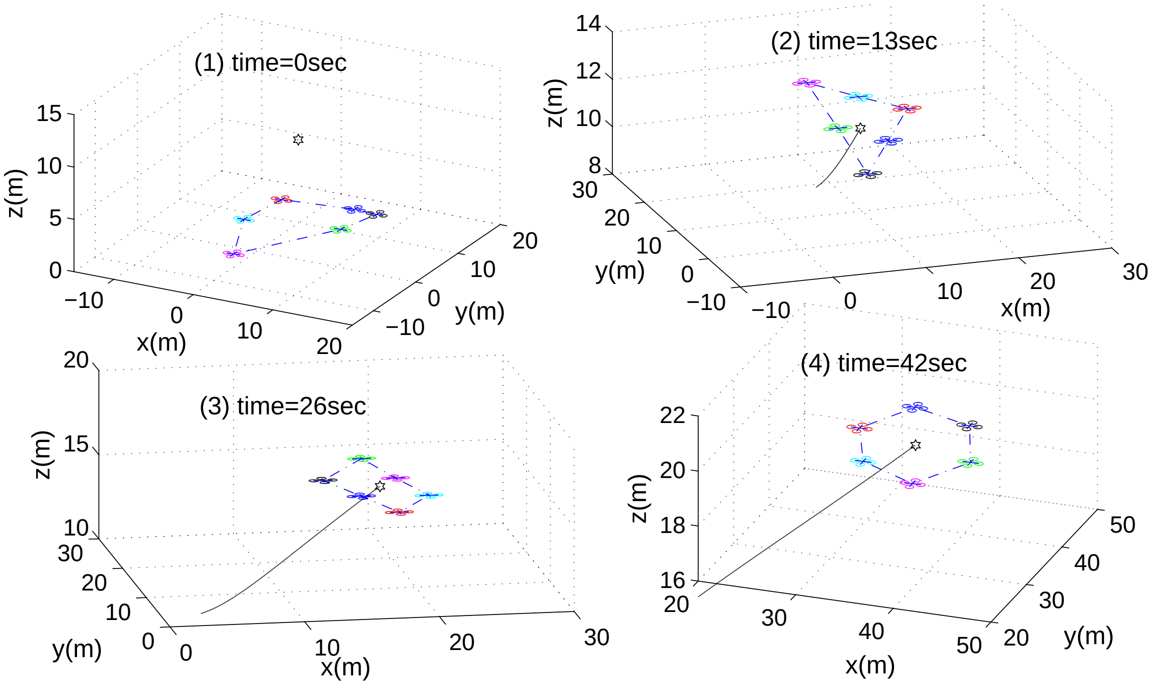


Figure 4.13: Position snapshots of the quadrotors achieving TVF and tracking the target trajectory (example-2).

we show in Fig. 4.18 and Fig. 4.19 the translational sub-system control inputs $u_i(t), i \in [1,6]$ resulted from the TVF Tracking controller (4.56) and the obtained quadrotors torques $\tau_{\phi,i}$, $\tau_{\theta,i}$, $\tau_{\psi,i}$, with the total thrust $F_i, i = [1,..,6]$ respectively.

It is important to notice that the TVF Tracking errors in Fig. 4.13 are bounded and lie in a neighbourhood of the origin. This is due to the fact that the target's control input is viewed as an external disturbance. Hence, the controller (4.56) is designed in sort to reduce its effect requiring the knowledge of the upper layer of the target's input magnitude $e_0$. This is considered as shortcoming that needs to be dealt with. Moreover, the chattering effect might be resulted due to the presence of discontinuous term in (4.56), thus, a widely common method to reduce this effect is by replacing the nonlinear function (4.58) with smooth approximated function [51]. However, the MASs stability needs to be restudied.

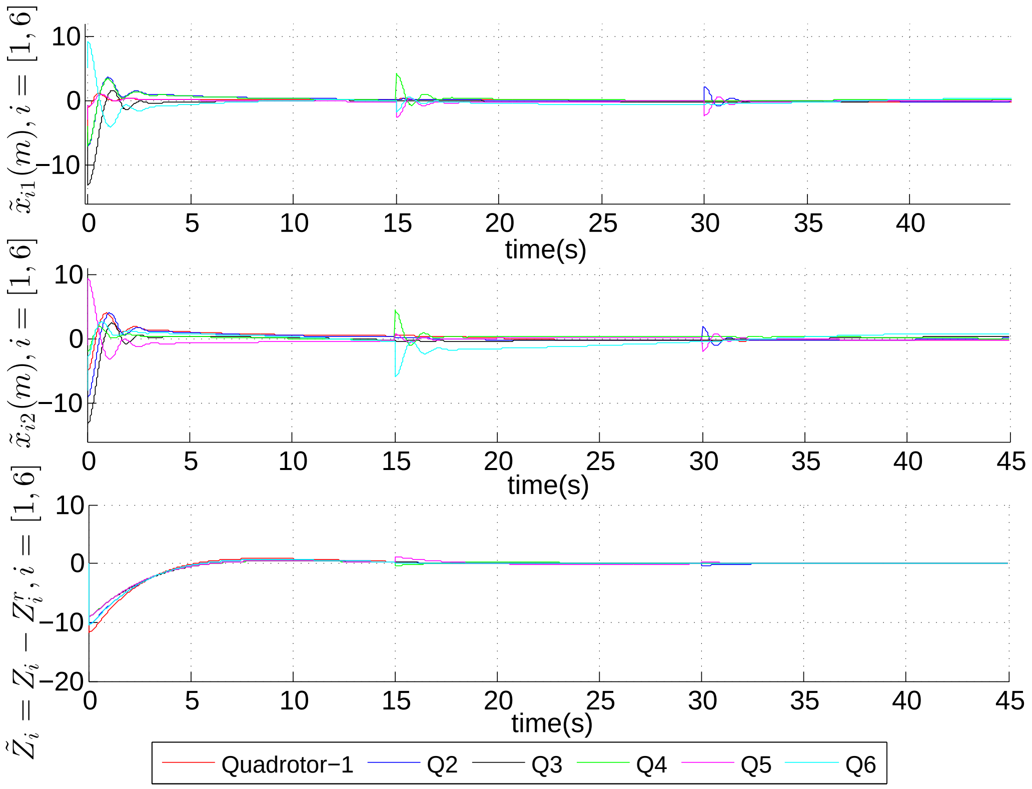


Figure 4.14: TVF tracking errors $\tilde{x}_i(m) = x_i - x_0 - h_i$, $i \in [1, ..., 6]$, (Example-2).

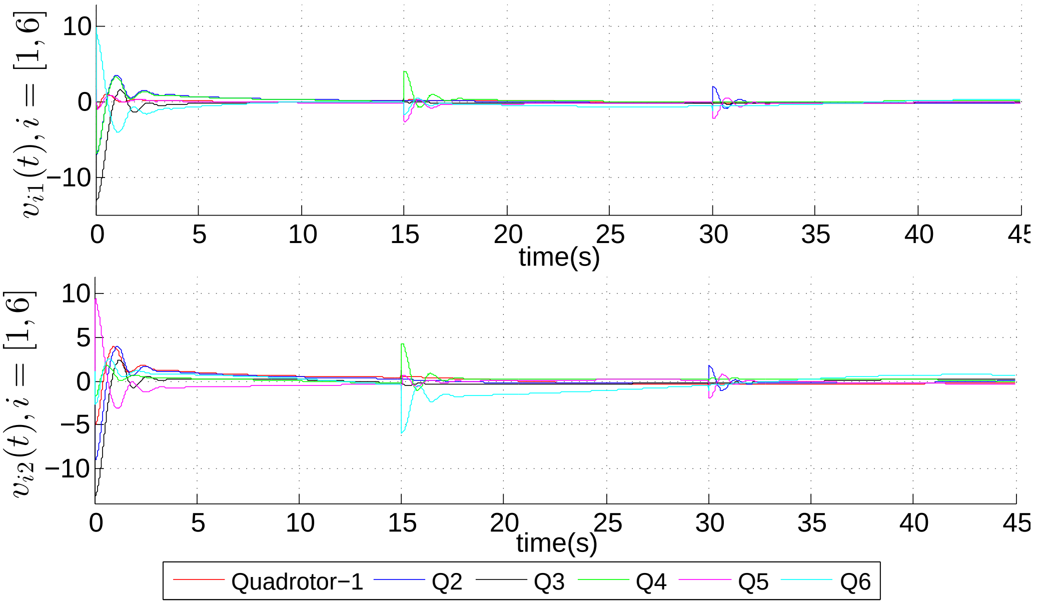


Figure 4.15: The distributed observer state $v_i(t)$, $i \in [1, ..., 6]$, (Example-2).

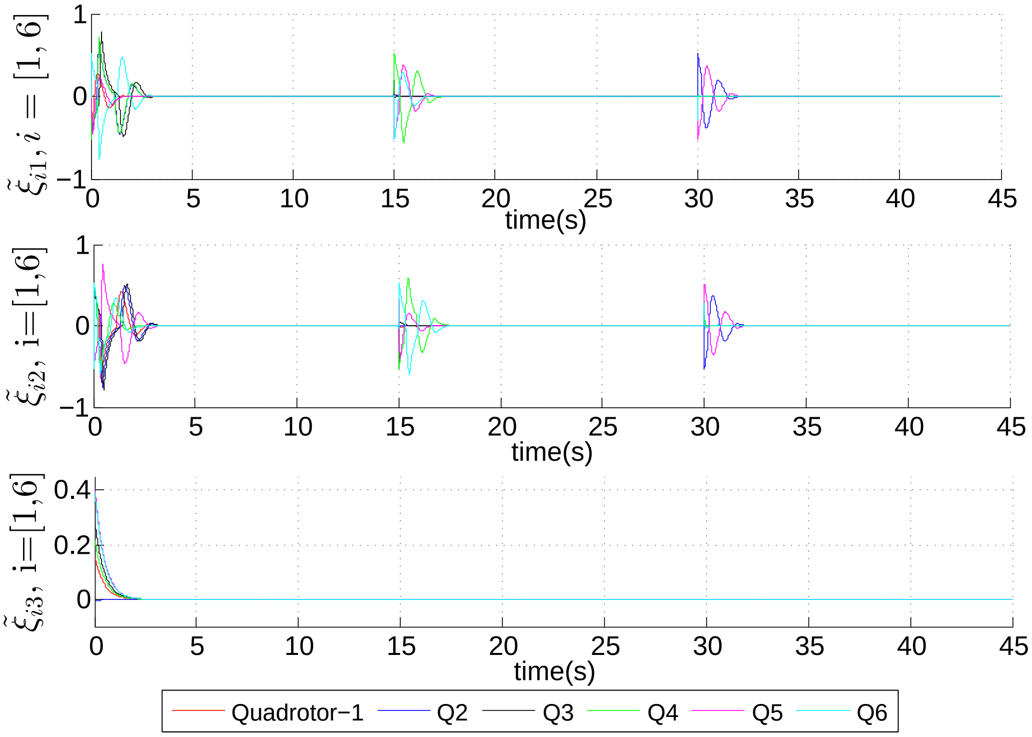


Figure 4.16: The quadrotors angles tracking errors $\tilde{\xi}_i(rad) = \xi_i - \xi_i^r$, $i \in [1, ..., 6]$, (Example-2).

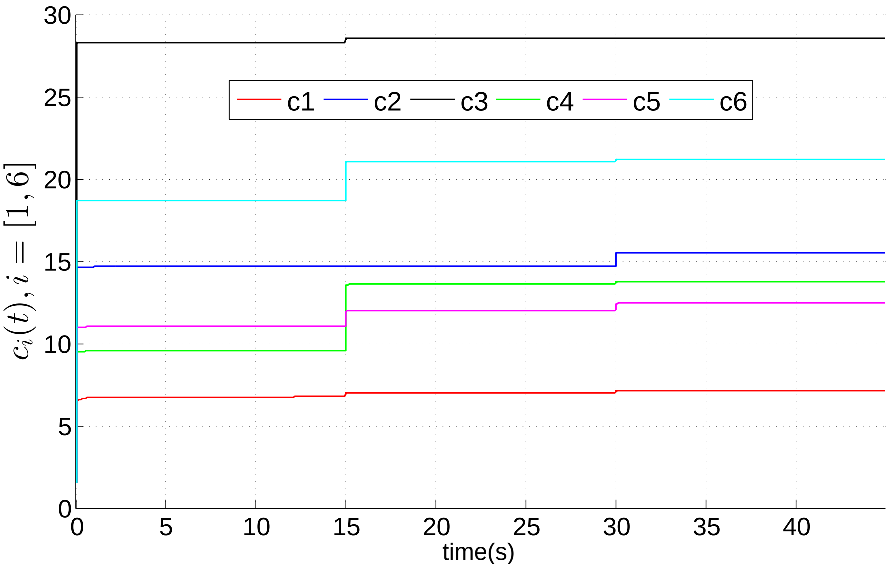


Figure 4.17: The adaptive coupling weight $c_i(t)$, $i \in [1, ..., 6]$, (Example-2).

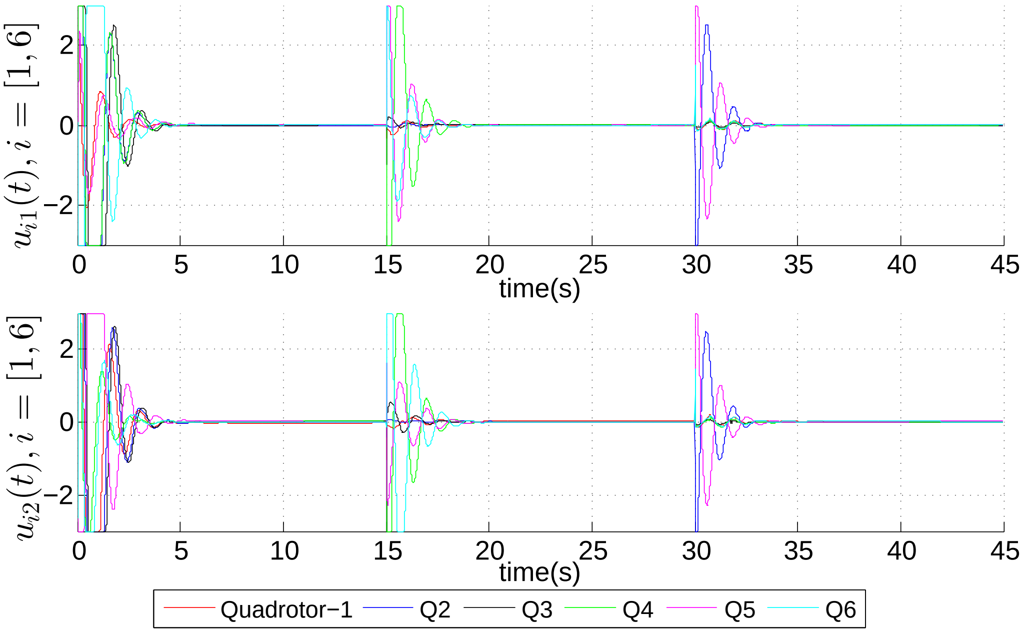


Figure 4.18: The MASs virtual control inputs (4.56), $u_i(t)$, $i \in [1, 6]$, (Example-2).

## Example-3:

In this example, we consider the same MASs of Example-2 that consists of six (06) quadrotors and a target (i.e., quadrotor) labelled by zero. We keep the same control approach opted in the previous example. The aim in this example is to examine the validity of the proposed TVF tracking controller (i.e., (4.56)) in the case of switching communication graphs (i.e., scenario-1) and further to analyse the effect of the output measurement delays and the control input delays on the MASs stability (i.e., scenario-2).

As stated in Example-2, the qudrotor control strategy is divided into two cascaded parts. In part-1, the TVF tracking control (4.56) is applied to the $x, y$ states dynamics, yielding the virtual control inputs $u_{x,i}$, $u_{y,i}$. These later signals will feed-forward the Euler angles control blocs (part-2).

In **scenario-1**, we suppose that the output delay and the control input delay are zero. In addition, we suppose that the interaction topology among the quadrotors agents is time-varying, given in Fig. 4.20, which means that some communication links among the agents may get broken and others get created over time, depending on the inter-agents distance. The desired formation shape $h(t)$ to be achieved

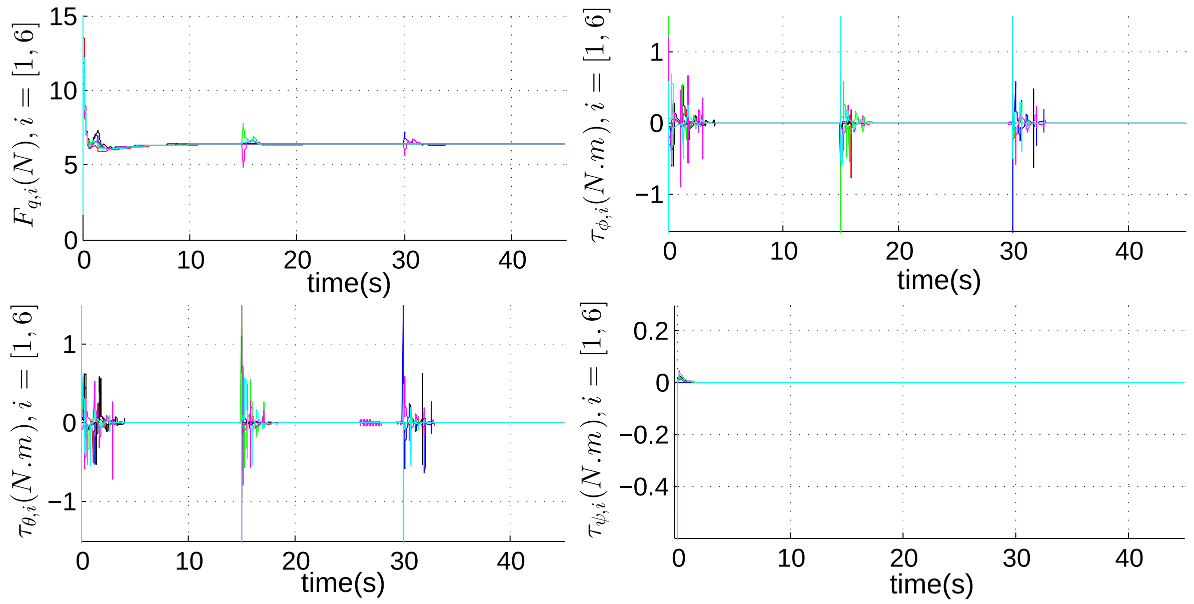
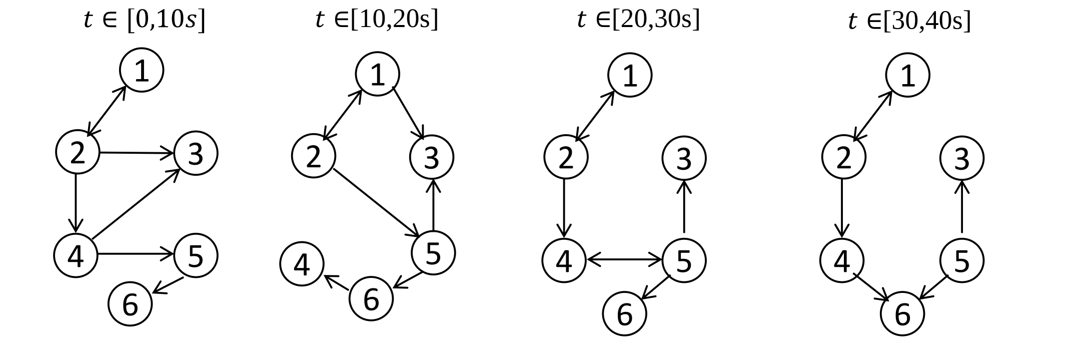


Figure 4.19: The quadrotors control inputs, $F_{q,i}$, $\tau_{\phi,i}$, $\tau_{\theta,i}$, $\tau_{\psi,i}$, $i \in [1, 6]$, (Example-2).

is the same as in Example-2. We keep the same control design parameters (i.e., $K$, $P$, $S$, $F$, $u_0(t)$, $\alpha$) as in Example-2. Refer to Fig. 4.20, note that interaction topology among the quadrotors is time-varying and contains a directed spanning tree rooted by agent 1, except during the time-interval $[30, 40s]$, where the graph doesn't satisfy the assumption-4.17. We can notice in Fig. 4.20-d, that the agents 3,5 do not receive information from the rest of the team.

In Fig. 4.21 and Fig. 4.22, we show the quadrotors tracking errors along the x-axis and y-axis $\tilde{x}_i = x_i - x_0 - h_i$, and the Euler angles tracking errors $\tilde{\xi}_i = \xi_i - \xi_i^r$,



(a) Directed graph (contains a spanning tree)

(b) Directed graph (contains a spanning tree)

(c) Directed graph (contains a spanning tree)

(d) Directed graph (Doesn't contain a spanning tree)

Figure 4.20: The sequence of the interaction topologies among the MASs (Example-3).

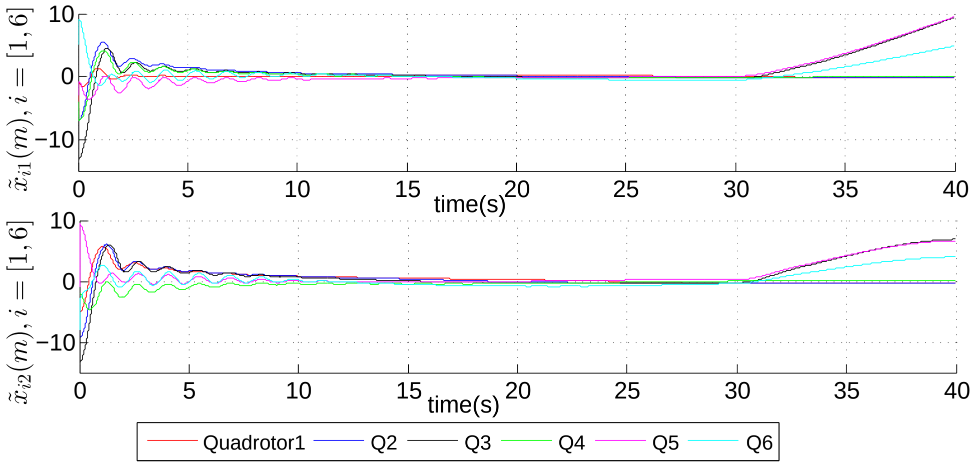


Figure 4.21: The quadrotors tracking errors $\tilde{x}_i(m) = x_i - x_0 - h_i$, $i \in [1, ..., 6]$ in Example-3 (scenario-1).

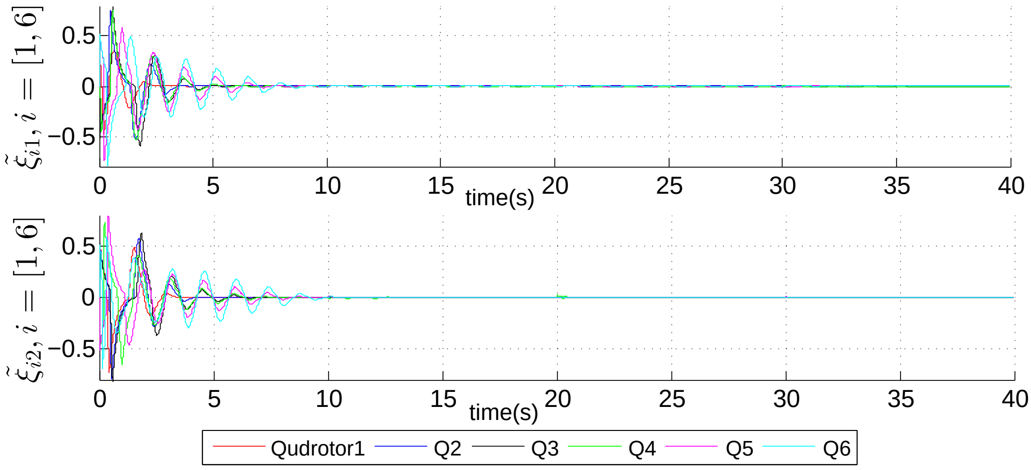


Figure 4.22: The Euler angles tracking errors $\tilde{\xi}_i(rad) = \xi_i - \xi_i^r$, $i \in [1, ..., 6]$ in Example-3 (scenario-1).

$i \in [1, 6]$ respectively. It can be seen that the quadrotors are tracking correctly the formation shape during the time-interval $[0, 30s]$ despite the fact that the communication graph is time-varying and satisfying assumption-4.17. However, starting from the time instant $t = 30s$, we observe that the $x$, $y$ tracking errors magnitude of agents 3,5,6 are not in the neighbourhood of the origin, which means that the agreement on the leader output $y_0(t)$ and the formation shape $h(t)$ is not achieved over time. This result promotes to the study and analysis of the MASs formation tracking in the case of switching graphs.

In **scenario-2**, we keep a fixed communication graph (Fig. 4.6-b), and we suppose that each agent $i$ is subjected to an output measurement delay $y_j(t - t_{net}), j \in [1, N]$ of

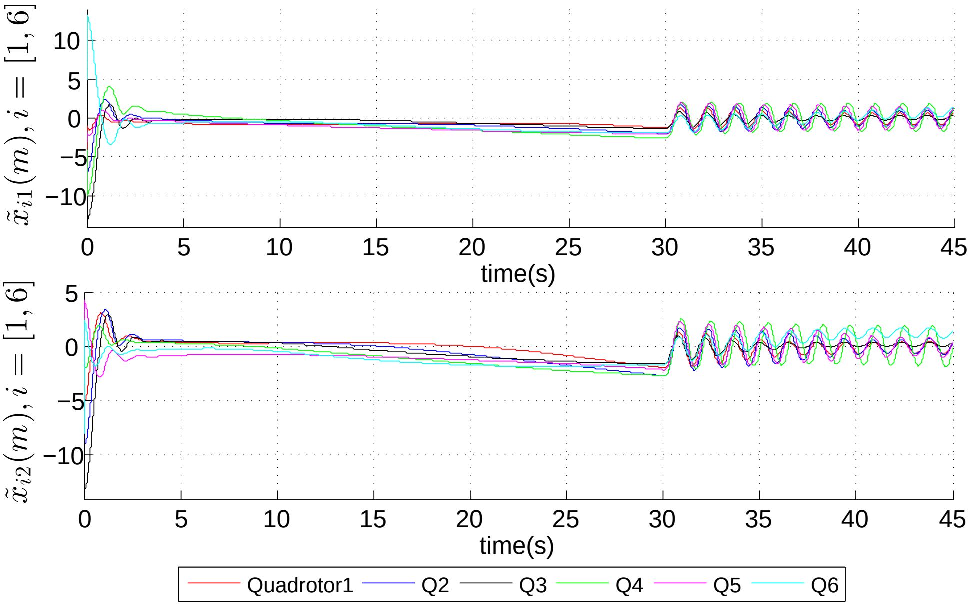


Figure 4.23: The quadrotors tracking errors $\tilde{x}_i(m) = x_i - x_0 - h_i$, $i \in [1, ..., 6]$ in Example-3 (scenario-2).

$t_{net} = 250ms$ during the time-interval$[0, 30s]$, and suppose further that the quadrotors are subjected to a control input delay $u_i(t - t_c), i \in [1, N]$ of $t_c = 90ms$ during the time-interval $[30, 45s]$. We keep the same control parameters as in Example-2.

In Fig. 4.23 and Fig. 4.24, we show the quadrotors tracking errors along the x-axis and y-axis $\tilde{x}_i = x_i - x_0 - h_i$ and the Euler angles tracking errors $\tilde{\xi}_i = \xi_i - \xi_i^r$, $i \in [1, 6]$ respectively. It is noticeable that the MASs formation is degraded during the time-interval $[0, 30s]$, and starting from $t = 30s$, oscillations are observed due to the effect of the input delay, which cause the MASs to get into the instability neighbourhood over time.

Fig. 4.25 and Fig. 4.26 show the virtual control inputs $u_{i,1}, u_{i,2}, i = [1, 6]$ and the quadrotors control inputs respectively. We can observe the chattering phenomena in $u_{i,1}, u_{i,2}$ which provokes the same result into the quadrotors control inputs $\tau_{\phi,i}, \tau_{\theta,i}$. The obtained result promote to the study and analysis of the MASs tracking control under the constraints of input and output delays.

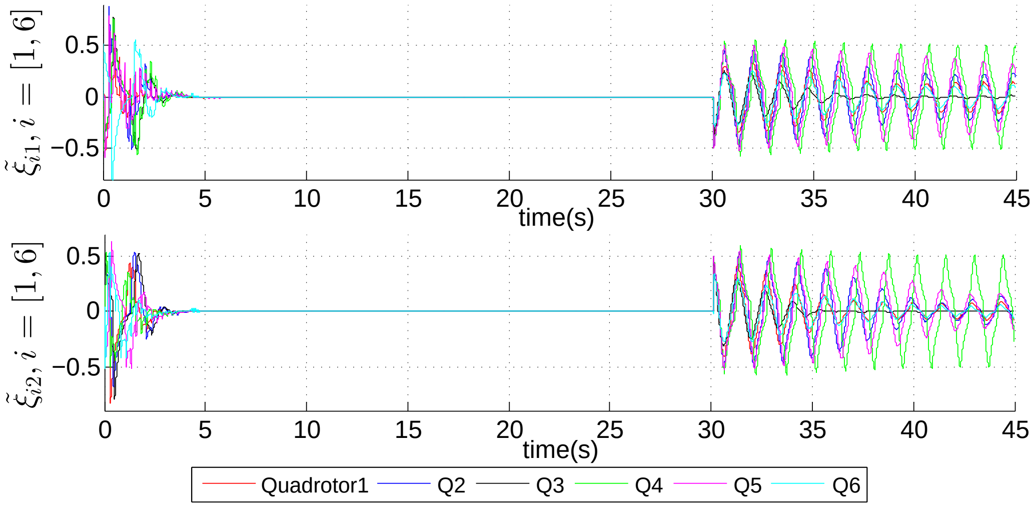


Figure 4.24: The Euler angles tracking errors $\tilde{\xi}_i(rad) = \xi_i - \xi_i^r$, $i \in [1, ..., 6]$ in Example-3 (scenario-2).

## 4.7 Conclusion

In this chapter, we addressed the fully distributed output TVF Tracking problem for MASs featured by general linear dynamics, under time-invariant interaction topology. The MASs is considered as Leader-Follower model. The study was conducted from undirected interaction topology to directed topology, and from a leader whose control input is zero $u_0(t) = 0$ to a leader with $u_0(t)$ different from 0, bounded and unknown to all the agents. The leader's output $y_0(t)$ is required to be known to only a subset of followers. In addition, the implementation of each agent local-controller requires receiving the relative measurable output information $(y_i - y_j)$ and the relative protocol observer-output $(Cv_i - Cv_j)$ instead of the full states vectors $(x_i - x_j)$, $(v_i - v_j)$. Furthermore, it can be seen that the proposed control design doesn't require exchanging the state of the local-observer $w(t)$, which yields a significant reduction of the network data-exchanging comparing with previous frameworks. According to the leader's input, two controllers were proposed, and accordingly two contributions were provided in this chapter on the TVF tracking framework corresponding to the case when the leader's input is zero and the case when the it is nonzero, bounded and unknown to all the followers agents,

- Firstly, a distributed algorithm is provided with clear steps to design the TVF Tracking controller satisfying reduced network information exchange among the

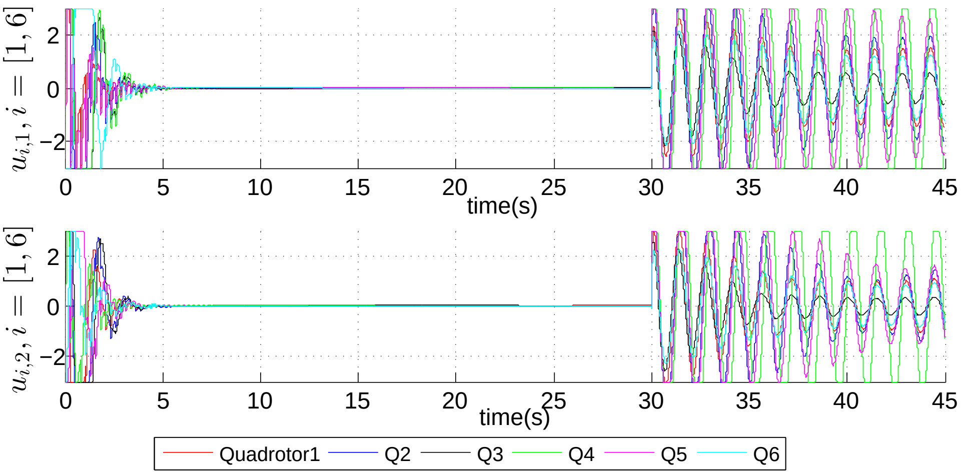


Figure 4.25: The quadrotors virtual control inputs (4.77), $u_{i1}(t)$, $u_{i2}(t)$, $i \in [1,6]$, in Example-3 (scenario-2).

agents. In which the interaction topology among the agents is directed containing a spanning tree rooted by the leader whose control input is zero.

- Secondly, we extended the above controller to the case where the leader's control input in nonzero, unknown and bounded. The controller design relies on adaptive observer-type and disturbance rejection technique.

- Thirdly, we analysed the effect of the interaction topology structure on the tracking-error convergence rate. The MASs stability is analysed based on Lyapunov theory.

- Finally, simulation results were provided to demonstrate the effectiveness of the proposed controllers on target enclosing scenario by a group of UAVs (Quadrotors).

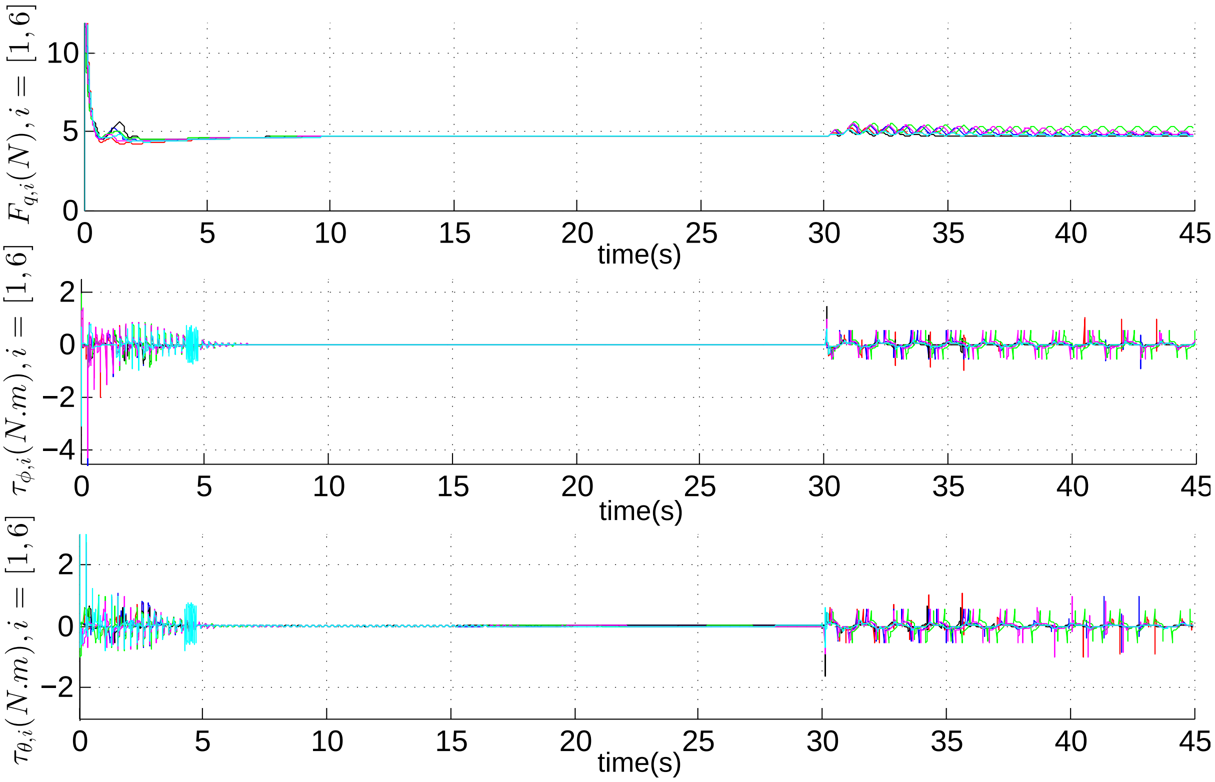


Figure 4.26: The quadrotors control inputs, $F_{q,i}$, $\tau_{\phi,i}$, $\tau_{\theta,i}$, $i \in [1, 6]$, Example-3 (scenario-2).

# GENERAL CONCLUSION

# GENERAL CONCLUSION

The main focus of this thesis was into developing and to designing cooperative control laws for MVSs, in particular in the branch of UAV-UVGs coordination control and the TVF tracking control of networked MASs. The objective is to improve the overall system performance in terms of robustness, distributed property, formation flexibility, network data exchanging-rate, and further to relax some requirements on the interaction topology among the networked MASs.

This thesis has mainly focuses on the following two aspects:

**1.** In the branch of UAV-UVGs formation and coordination control, the aim was into designing robust nonlinear control laws for the UGVs to realize two behaviours, namely, deployment and formation tracking. The UAV acts as leader whose role is to define the formation shape and the formation trajectory. The UGVs-formation shape was modelled using EFDs tool and its corresponding IPF. Thus, a novel deployment control was proposed, in which the estimated IPF is used as potential function to generate the UGVs reference trajectories. In the TVF tracking control, a combined robust velocity/torque controller based Backsteeping is proposed, for which the UGVs-formation flexibility and robustness have been improved face modeling errors and external disturbance. The key idea is to introduce a virtual control input through a nonlinear feedback, and further by adding a switching term function of the upper bound of the disturbance to guarantee the mobile robots velocity tracking. The simulation results revealed good performance in terms smooth convergence towards the desired TVF shape, accurate tracking, and further in terms of robustness against unknown disturbance. An experimental validation is further provided using three mobile robots type festo's Robotino$^{(\mathbb{R})}$.

**2.** In the branch of TVF tracking of networked MASs, the aim was into designing fully distributed cooperative control laws that provide several contributions such as, (i) relaxing the constraints on the interaction topology among the agents, where it is not required to be strongly connected, instead it needs just to have a directed spanning tree, which is a mild requirement in real applications, (ii) the proposed control laws are valid for the case of switching communication graphs as long as the assumption-4.17 still satisfied, (iii) the implementation of each agent local-controller, requires receiving the relative neighbouring outputs measurement and the relative protocol-observer outputs rather then the full relative states vectors, and doesn't require receiving the neighbouring local-observer state as in many frameworks, which yields a significant reduction of the network information exchanging. The opted MASs control structure is Leader-Followers. The proposed TVF tracking protocols are valid for a leader whose control input is zero and extended to the case of real leader whose control input is bounded and unknown. The MASs control design relies on adaptive observer-type and on disturbance rejection techniques to deal with the nonzero leader control input. The analysis in this chapter was conducted in order to derive a description as global as possible of the errors closed loop dynamics stability induced by the proposed protocols by using the Lyapunov theory. The analysis of the effect of the MASs communication-graph on the convergence rate is provided. Finally, a target enclosing and trajectory tracking by a group of quadrotors is illustrated as an application of the proposed MASs protocols.

However, there remain potentially new findings in the area of cooperative control of networked MASs that can still be explored. These can be summarized as follows:

- Extend the current study to the case of switching (i.e., time-varying) directed communication graph that does not satisfy in permanence the assumption-4.17 (i.e., the existence of a directed spanning tree). As such, the MASs communication graph can be weakly connected during certain time-intervals, and the designed MASs protocol would be more suitable and more applicable in real conditions.

- Pointing out attention most to the study of heterogeneous MASs rather then the homogeneous MASs, where the heterogeneousness may lie in the agents/vehicles dynamics or in its operating environment (UAV, UGV, UWV), which will yield an important functional complementarity in real applications. However the control design becomes difficult and more challenging, in particular when considering input/output delays constraints.

- Extend the current study to design distributed MASs controllers under the constraints of external unknown disturbance, constant or time-varying control input delay and output measurements delay. Thus, because we believe that their effect may have a huge impact on the system stability and the MASs performance.

- The Artificial Intelligence (AI) becomes every where in our life. We believe that combining the new findings of this research-area into the area of MASs cooperative control may provide potentials solutions to handle some issues and constraints such as, input/output delays. As an example, equipping each agent with the AI-ability (by using deep reinforcement learning method) may promote to design an intelligent TVF tracking protocols for a MASs.

- Implementing the results in this thesis to real experiments in laboratory is important. During this thesis researches, a laboratory-made MVSs platform, consisting of a quadrotor and two balancing/unicycle mobile robots was realized based on Arduino board units, where the communication task is carried out using ESP8266 WiFi-devices. As such, it is possible to measure the network data exchanging-rate, the input and output delays with high accuracy, as we have access to low-level control loops.

# BIBLIOGRAPHY

# BIBLIOGRAPHY

[1] HULYA Yalcin, UNEL, Mustafa, WOLOVICH, William. Implicitization of parametric curves by matrix annihilation. *International Journal of Computer Vision*, 2003, vol. 1, no 54, pp. 105–115.

[2] ALLAM, Ahmed, TADJINE, Mohamed, KOBZILI, Elhaouari, et al. 3D Robots Formation cooperative Control Using EFDs. In 2018 3rd IEEE International Conference on Pattern Analysis and Intelligent Systems (PAIS), 2018, pp. 1–6.

[3] OH, Kwang-Kyo, PARK, Myoung-Chul, and AHN, Hyo-Sung. A survey of multi-agent formation control. *Automatica*, vol. 53, pp. 424–440.

[4] CORTES, Jorge, and MAQNUS, Egerstedt. Coordinated control of multi-robot systems: A survey. *Journal of Control, Measurement, and System Integration*, 2017, vol. 10, no 6, pp. 1–9.

[5] PARKER, Lynne, DANIELA, Rus, and GAURAY, Sukhatme. Multiple mobile robot systems, Springer Handbook of Robotics, 2008, pp. 921–941.

[6] CAO, Yongcan, WENWU Yu, WEI Ren, and CHEN Guanrong. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transation Industriel infomatics*, 2012, vol. 9 ,no 1, pp. 427–438.

[7] ZHANG, Dan, XU, Zhenhua, REZA, Hamid, WANG, Qing-Guo. Distributed output-feedback control for consensus of heterogeneous linear multiagent systems with aperiodic sampled-data communications. *IEEE Transactions on Industrial Electronics*, 2018, vol. 65, no 5, pp. 4145–4155.

[8] DONG, Xiwang, and HU, Guoqiang. Time-varying formation control for general linear multi-agent systems with switching directed topologies. *Automatica*, 2016, vol. 1, no 73, pp. 47–55.

[9] ANTONELLI, Gianluca, FILIPPO, Arrichiello, FABRIZIO, Caccavale. Decentralized time-varying formation control for multi-robot systems. *International Journal of Robotics Research*, 2014, vol. 33, no 7, pp. 1029–1043.

[10] ZHAO, Qilun, XIWANG Dong, XUN Song, and ZHANG Ren. Cooperative time-varying formation guidance for leader-following missiles to intercept a maneuvering target with switching topologies. *Nonlinear Dynamics*, 2019, vol. 95, no 1, pp. 129–141.

[11] ZHANG, Mingfeng, and HUGH Liu. Cooperative tracking a moving target using multiple fixed-wing UAVs. *Journal of Intelligent and Robotic Systems*, 2016, vol. 81, no 3, pp. 505–529.

[12] LIN, Jiaying, JAN-JORAN Gehrt, RENE, Zweigel, and DIRK, Abel. Cooperative localization of networked multi-agent system. in: Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation, Miami, 16-19 sept 2019, pp. 1976–1990.

[13] SASKA, Martin, VOJTECH, Vonasek, JAN, Chudoba, et al. Swarm distribution and deployment for cooperative surveillance by micro-aerial vehicles. *Journal of Intelligent Robotic Systems*, 2016, vol. 84, no 1, pp. 469–492.

[14] BRANDAO, Alexandre, MARIO, Sarcinelli-Filho, et al. Decentralized control of leader–follower formations of mobile robots with obstacle avoidance, *in: IEEE International Conference on Mechatronics*, Spain, 2009, pp. 1–6.

[15] HOU, Zhicheng, and FANTONI, Isabelle, Distributed leader-follower formation control for multiple quadrotors with weighted topology, in: In SoSE 10th System of Systems Engineering Conference, IEEE, USA, 2015, pp. 256–261.

[16] YANG, Xi-Xin, GONG-YOU, Tang, YANG, Li, and WANG, Pei-Dong, Formation control for multiple autonomous agents based on virtual leader structure, in: IEEE 24th Chinese control and decision conference (CCDC), Taiyuan, China, 2012, pp. 2833–2837.

[17] REZAEE, Hamed, and FARZANEH, Abdollahi. Decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots. *IEEE Transactions on Industrial Electronics*, 2014, vol. 61, no 1, pp. 347–354.

[18] WEI, Ren. Consensus strategies for cooperative control of vehicle formations. *IET Control Theory and Applications*, 2007, vol. 1, no 2, pp. 505–512.

[19] KHAN, Muhammed, LI, Shuai, QIXIN, Wang, and ZILI, Shao. Formation control and tracking for cooperative robots with nonholonomic constraints. *Journal of Intelligent and Robotic Systems*, 2016, vol. 82, no 1, pp. 163–174.

[20] ZHONGKUI, Li, DUAN, Zhisheng, CHEN, Guanrong, and HUANG, Lin. Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2010, vol. 1, no 57, pp. 213–224.

[21] ZHONGKUI, Li, WEN, Guanghui, DUAN, Zhisheng, and WEI, Ren. Designing fully distributed consensus protocols for linear multi-agent systems with directed graphs, *IEEE Transactions on Automatic Control*, 2014, vol. 60, no 4, pp. 1152–1157.

[22] YU, Ziquan, YAOHONG, Qu, and YOUMIN, Zhang. Fault-tolerant containment control of multiple unmanned aerial vehicles based on distributed sliding-mode observer. *Journal of Intelligent and Robotic Systems*, 2019, vol. 93, no 1, pp. 163–177.

[23] WU, Qize, JUNTONG, Qi, CHONG, Wu, and WANG, Mingming, Design of UGV trajectory tracking controller in UGV-UAV cooperation, in: IEEE 39th Chinese Control Conference (CCC), China 2020, pp. 3689–3694.

[24] AGHAEEVAN, A., ABDOLLAHI, F., and TALEBI, H. Robust cooperative control in the presence of obstacles, In: 21st IEEE Iranian Conference on Electrical Engineering (ICEE), 2013, pp. 1–6.

[25] RAHIMI, Reihane, FARZANEH, Abdollahi, and KARO, Naqshi. Time-varying formation control of a collaborative heterogeneous Multi Agent System. *Robotics and autonomous systems*, 2014, vol. 62, no 12, pp. 1799–1805.

[26] MIGUEL, Aranda, MEZOUAR, Youcef, et al. Scale-free vision-based aerial control of a ground formation with hybrid topology. *IEEE Transactions on Control Systems Technology*, 2018, vol. 27, no 4, pp. 1703–1711.

[27] MATHEWS, Nithin, ANDERS, Christensen. Supervised morphogenesis: Exploiting morphological flexibility of self assembling multi-robot systems through cooperation with aerial robots. *Robotics and Autonomous Systems*, 2019, vol. 154, no 112, pp. 154–167.

[28] SANTANA, Lucas, S. BRANDAO, Alexandre, et al. Heterogeneous leader-follower formation based on kinematic models, in: International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, USA, 2016, pp. 342–346.

[29] RABELO, MARCOS, Felipe Santos, et al. Centralized control for an heterogeneous line formation using virtual structure approach, in: IEEE Procedings of Latin American Robotic Symposium, Brazil, 2018, pp. 135–140.

[30] YESIM, Esin, MUSTAFA, Unel, et al. Formation control of multiple robots using parametric and implicit representations, in: ICIC International Conference on Intelligent Computing, Berlin, 2008, pp. 558–565.

[31] SANEM, Evren, MUSTAFA, Unel. Planar formation control of swarm robots using dynamical elliptic fourier descriptors. *Transactions of the Institute of Measurement and Control*, 2015, vol. 5, no 37, pp. 661–671.

[32] REZA, Olfati-Saber, ALEX, Fax, et al. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 2007, vol. 95, no 1, pp. 215–233.

[33] WEI, Ren, ELLA, Atkins. Distributed multi-vehicle coordinated control via local information exchange. *International Journal of Robust and Nonlinear Control*, 2007, vol. 17, no 10, pp. 1002–1033.

[34] XIWANG, Dong, JIE, Xiang, LIANG, Han, et al. Distributed time-varying formation tracking analysis and design for second-order multi-agent systems. *Journal of Intelligent Robotic Systems*, 2017, vol.86, no 2, pp. 277–289.

[35] DONG, Xiwang, ZONGYING Shi, GENG Lu, et al. Time-varying formation control for high-order linear swarm systems with switching interaction topologies. *IET Control Theory and Applications*, 2014, vol. 8, no 18, pp. 2162–2170.

[36] RUI, Wang, XIWANG, Dong, QINGDONG, Li, et al. Distributed adaptive time-varying formation for multi-agent systems with general high-order linear time-invariant dynamics. *Journal of the Franklin Institute*, 2016, vol. 353, no 10, pp. 2290–2304.

[37] ZHANG, Mingfeng, and HUGH Liu. Cooperative tracking a moving target using multiple fixed-wing UAVs. *Journal of Intelligent and Robotic Systems*, 2016, vol. 81, no 3, pp. 505–529.

[38] BRINON-ARRANZ, Lara, SEURET, Alexandre. Cooperative control design for time-varying formations of Multi-Agent Systems. *IEEE Transactions on Automatic Control*, 2014, vol. 58, no 8, pp. 2283–2288.

[39] DONG, Xiwang, JIANXIANG, Xi, GENG, Lu, et al. Formation control for high-order linear time-invariant Multi-Agent Systems with time delays. *IEEE Transactions on Control of Network Systems*, 2014, vol. 1, no 3, pp. 232–240.

[40] WANG, Rui, XIWANG. Dong, QINGDONG, Li, et al. Distributed adaptive formation control for linear swarm systems with time-varying formation and switching topologies. *IEEE Access*, 2016, vol. 4, pp. 8995–9004.

[41] WANG, Rui, XIWANG. Dong, QINGDONG, Li, et al. Distributed adaptive control for time-varying formation of general linear multi-agent systems. *International Journal Systems Science*, 2017, vol.48, no 16, pp. 3491–3503.

[42] WANG, Rui, XIWANG. Dong, QINGDONG, Li, et al. Distributed time-varying output formation control for general linear multiagent systems with directed topology. *IEEE Transactions on Control of Network Systems*, 2018, vol. 6, no 2, pp. 609–620.

[43] WANG, Rui, XIWANG. Dong, QINGDONG, Li, et al. Distributed time-varying formation control for linear swarm systems with switching topologies using an adaptive output-feedback approach. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017, vol. 49, no 12, pp. 2664–2675.

[44] ZHAO, Yu, DUAN, Qixiu, WEN, Guanghui, et al. Time-varying formation for general linear multiagent systems over directed topologies: A fully distributed adaptive technique. *IEEE Transactions on Systems, Man, and Cybernetics: System*s, 2018, vol. 51, no 1, pp. 1-10.

[45] WANG, Rui, DONG, Xiwang, LI, Qingdong, et al. Distributed time-varying formation control for multiagent systems with directed topology using an adaptive output-feedback approach. *IEEE Transactions on Industrial Informatics*. 2019, vol. 15, no 8, p. 4676-4685.

[46] DONG, Xiwang, and GUOGIANG, Hu. Time varying formation tracking for linear Multi-Agent Systems with multiple leaders. *IEEE Transactions on Automatic Control*, 2017, vol. 62, no 7, pp. 3658–3664.

[47] HE, Lei, XIUXIA, Sun, and YAN, Lin. Distributed adaptive control for time-varying formation tracking of a class of networked nonlinear systems. *International Journal of Control*, 2017, vol. 90, no 7, pp. 1319–1326.

[48] WEI, Jiang, GUOGUANG, Wen, et al, Distributed adaptive time-varying formation tracking for linear multi-agent systems: A dynamic output approach, in: In 36th IEEE Chineese Control Conference (CCC), 2017, pp. 8571–8576.

[49] ALLAM, Ahmed, NEMRA, Abdelkrim, and TADJINE, Mohamed. Parametric and implicit features based UAV-UGVs time-varying formation tracking: Dynamic approach. *Journal of Unmanned Systems*, 2022, vol. 10, no 1, pp. 1–20, available in https://doi.org/10.1142/S2301385022500066.

[50] XIAOHUA, Ge, FUWEN, Yang, QINGLONG, Han. Distributed networked control systems: A brief overview. *Information Sciences*, 2017, vol. 380, pp. 117–131.

[51] LV, Yuezu, ZHONGKUI, Li, et al. Fully distributed adaptive output feedback protocols for linear Multi-Agent Systems with directed graphs: A sequential observer design approach, 2015. arXiv preprint, [visited 10 January 2019]. Available in: https://arxiv.org/abs/1511.01297.

[52] LV, Yuezu, ZHONGKUI, Li, and DUAN, Zhisheng. Distributed adaptive consensus protocols for linear Multi-Agent Systems over directed graphs with relative output information. *IET Control Theory and Applications*, 2018, vol. 12, no 5, pp. 613–620.

[53] LI, Man, QICHAO, MA, ZHOU, Chongjian, et al. Distributed time-varying group formation control for generic linear systems with observer-based protocols.*Neurocomputing*. 2020, vol. 397, pp. 244-252.

[54] KUHL, Frank, and GIARDINA, Charles. Elliptic fourier features of a closed contour. *Computer Graphics and Image Processing*, 1982, vol. 18, no 3, pp. 236–258.

[55] ESIN, Yeşim Humay, and UNEL, Mustafa. Formation control of non-holonomic mobile robots using implicit polynomials and Elliptic Fourier Descriptors. *Turkish Journal of Electrical Engineering and Computer Sciences*, 2010, vol. 18, no 5, pp. 765–778.

[56] MARK, Nixon, ALBERTO, Aguado, Feature extraction and image processing, Elsevier, London, 2008, [visited 25 march 2017]. available in: https://dl.acm.org/doi/10.5555/1571711.

[57] BAN, WANG, and ZHANG, Youmin. Adaptive sliding mode fault-tolerant control for an Unmanned Aerial Vehicle. *Unmanned Systems Journal*, 2017, vol. 5, no 4, pp. 209–221.

[58] BOUABDALLAH, Samir, *Design and control of quadrotors with application to autonomous fying*. PhD thesis: Automatique et informatique Industrielle. Lausanne: Ecole Polytechnique Federale, 2007, 155 p.

[59] RAFEAL, Fierro, and FRANK, Lewis. Control of a nonholomic mobile robot: Backstepping kinematics into dynamics. *Journal of robotic systems*, 1997, vol. 3, no 14, pp. 149–163.

[60] SAIDI, Yasmine, NEMRA, Abdelkrim, and TADJINE, Mohamed. Robust mobile robot navigation using fuzzy type 2 with wheel slip dynamic modeling and parameters uncertainties. *International Journal of Modeling and Simulation*, 2020, vol. 40, no 6, pp. 397–420.

[61] RACHED, Dhaouadi, and ABU-HATAB, Ahmed. Dynamic modelling of differential-drive mobile robots using lagrange and newton-euler methodologies a unified framework. *Advances in Robotics and Automation*, 2013, vol. 2, no 3, pp. 1–7. Rached Dhaouadi Ahmad Abu Hatab

[62] FRANK, Lewis. Neural network control of robot manipulators. *IEEE Expert*, 1996, vol. 11, no 3, pp. 64–75.

[63] HASSAN, Khalil. Nonlinear systems *Third Edition*. in: Englewood Cliffs, NJ: Prentice Hall, 2002.

[64] Website of the festo subsidiary. [visited 15 december 2019]. URL: https://www.festo-didactic.com.

[65] KELLALIB, Bilel, ACHOUR, Nouara, et al. Towards simultaneous localization and mapping tolerant to sensors and software faults: Application to omnidirectional mobile robot. *Procedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 2021, vol. 235, no 2, pp. 269–288.

[66] SAIF, Osamah, FANTONI, Isabelle, and ZAVALA, Arturo. Real-time flocking of multiple-quadrotor System of Systems. in: IEEE 10th System of Systems Engineering Conference (SoSE), USA, 2015.

[67] KAZUNORI, Sakurama. Multi-robot formation control over distance sensor network. *IFAC-Papers On Line*, 2016, vol. 49, no 22, pp. 198–203.

[68] TENGFEI, Liu, ZHON, Jiang. Distributed formation control of nonholonomic mobile robots without global position measurements. *Automatica*, 2013, vol. 49, no 2, pp. 592–600.

[69] EHSAN, Peymani, HAVARD, Grip, SABERI, Ali, et al. H$infini$ almost output synchronization for heterogeneous networks of introspective agents under external disturbances. *Automatica*, 2014, vol. 50, no 4, pp. 1026–1036.

[70] GODSIL, Chris, and GORDON, Royle. Algebraic graph theory. *Springer Graduate Texts in Mathematics*, 2001, vol. 207, 433 p.

[71] OLFATI-SABER, Reza, and RICHARD, Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 2004, vol. 49, no 9, pp. 1520–1533.

[72] QU, Zhihua, Cooperative control of dynamical systems: Applications to autonomous vehicles, in: Springer Science and Business Media, 2009.

[73] DENNIS, Bernstein, Matrix mathematics: Theory, facts, and formulas (Second Edition), Princeton: Princeton University Press, 2009, [visited 25 march 2019], available in: https://doi.org/10.1515/9781400833344.

[74] KRSTIC, Miroslav, KOKOTOVIC, Petar, IOANNIS, Kanellakopoulos, Nonlinear and adaptive control design, in: John Wiley and Sons (1st Edition), 1995, ISBN: 978-0-471-12732-1, 576 p.

[75] JADABAIE, Ali, JIE, Lin, STEPHAN, A. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 2003, vol. 48, no 6, pp. 988–1001.

[76] TAMAS, Vicsek, ANDRAS, Czirok, et al. Novel type of phase transitions in a system of self-driven particles. *Physical Review Letters*, 1995, vol. 75, no 6, pp. 1226–1229.

[77] WEI, Ren, KEVIN, Moore, and CHEN, Yangquan. High-order and model reference consensus algorithms in cooperative control of multi-vehicle systems. *Journal of dynamical systems, measurement and control*, 2007, vol. 129, no 5, pp. 678–688.

[78] ZHONGKUI, Li, REN, Wei, XIANGDONG, Liu, et al. Consensus of Multi-Agent Systems with general linear and lipschitz nonlinear dynamics using distributed adaptive protocols. *IEEE Transactions on Automatic Control*, 2012, vol. 58, no 7, pp. 1786–1791.

[79] ZHONGKUI, Li, REN, Wei, XIANGDONG, Liu, et al. Distributed consensus of linear multi-agent systems with adaptive dynamic protocols. *Automatica*, vol. 9, no 7, pp. 1986–1995.

[80] WEI, Jiang. A unified framework of fully distributed time-varying formation control for large-scale multi-agent systems: An observer viewpoint, in: 18th IEEE European Control Conference (ECC), Italy, 25-29 June 2019, pp. 2338–2343.

[81] WEI, Jiang, ZHAXIA, Peng, et al. A unified framework of fully distributed output Time-Varying Formation Control for linear Multi-Agent Systems: an observer viewpoint. 2018, in arXiv preprint. Available in: https://arxiv.org/abs/1803.08989.

[82] XIWANG, Dong, JIANXIANG, Xi, GENG, Lu, et al. Formation control for high-order on control of network systems. *IEEE Transactions on Control of Network Systems*, 2014, vol. 1, no 3, pp. 232–240.

[83] ALLAM, Ahmed, NEMRA, Abdelkrim, TADJINE, Mohamed. Distributed time-varying formation tracking of MASs with reduced network information exchange and nonzero leader’s input, Unmanned Systems Journal (Under review December 2021).

[84] ZHICHENG, Hou, FANTONI, Isabelle. Leader-Follower formation saturated control for multiple quadrotors with switching topology, In: IEEE Workshop on Research, Education and Development of Unmanned Aerial Systems, 2015, pp. 8–14.