HAL
open science

Fault tolerant methods design for a fleet of autonomous vehicles against faults/failures based on multi-agent systems

Juan Antonio Vazquez Trejo

To cite this version:

Juan Antonio Vazquez Trejo. Fault tolerant methods design for a fleet of autonomous vehicles against faults/failures based on multi-agent systems. Automatic. Université de Lorraine; Centro Nacional de Investigación y Desarrollo Tecnológico (Cuernavaca, Mor., México), 2021. English. NNT : 2021LORR0122 . tel-03342043

HAL Id: tel-03342043

https://hal.univ-lorraine.fr/tel-03342043v1

Submitted on 13 Sep 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE LORRAINE

AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4
Code de la Propriété Intellectuelle. articles L 335.2- L 335.10
http://www.cfcopies.com/V2/leg/leg_droi.php
http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

École doctorale IAEM Lorraine

Fault Tolerant Methods Design for a Fleet of Autonomous Vehicles Against Faults/Failures Based on Multi-agent Systems

THESIS

présentée et soutenue publiquement le 13 juillet 2021

pour l'obtention du

Doctorat de l'Université de Lorraine

(mention automatique)

par

Juan Antonio VAZQUEZ TREJO

Composition du jury

Rapporteurs :	Isabelle FANTONI	Senior Researcher Laboratoire des Sciences du Numérique de Nantes (LS2N) UMR CNRS 6004 Nantes Cedex 3, France.
	Mohammed CHADLI	Professeur, Université Paris-Saclay Laboratoire IBISC - Univ Evry, France.
Examinateurs :	Cristina MANIU	Professeur,CentraleSupélec/Laboratoire des Signaux et Systèmes Gif sur Yvette, France.
	Andrea MONTERIU	Associate Professeur, Department of Information Engineering Università Politecnica delle Marche, Ancona, Italy.
Directeur :	Manuel ADAM MEDINA	Professeur, Cenidet/Tecnm, Mexique.
	Didier THEILLIOL	Professeur, Université de Lorraine, France.

Centre de Recherche en Automatique de Nancy CRAN – CNRS UMR 7039

Acknowledgments

I want to thank CONACYT (Consejo Nacional de Ciencia y Tecnología) for giving me the opportunity to obtain a doctoral scholarship and a scholarship to stay abroad; to CENIDET (Centron Nacional de Investigación y Desarrollo Tecnológico) for accepting me in their program specifically in the department of electronic engineering (DIE), to CRAN (Centre de Recherche en Automatique de Nancy) and l'Université de Lorraine for accepting the work in co-tutelle of this thesis.

I am grateful with affection and gratitude to Lic. Lorena Ruiz Ramirez; to my thesis directors Dr. Manual Adam Medina with whom I have been working for 6 years with great pleasure; to Dr. Carlos Daniel García Beltrán and Prof. Didier Theilliol who have supported me in many ways during the development of this thesis; to my jury in France Dr. Isabelle Fantoni, Prof. Mohammed Chadli, Prof. Cristina Maniu, Prof. Andrea Monteriu; and my jury in Mexico Dr. Victor Manuel Alvarado Martínez, Dr. Gerardo Vicente Guerrero Ramírez, and Dr. Dante Mújica Vargas; especially, to Adrien Guenard, Laurent Ciarletta, and Jean-Christophe Ponsart for helping me doing the implementation of the algorithms in Loria.

I am thankful to Damiano Rotondo for giving me his time and explaining what is necessary to understand the tools used in this thesis, in addition to the many collaborations together; to César Martínez and Violeta Cervantes for helping me during my stay in France; to Abderrahmane Jarrou for being *my brother from another mother*.

I am indebted to Alfonso de Jesús Morales because He has helped me and has been a great friend and partner. I am contented to my friends in Cenidet Alexis Alonso, Carlos Ríos, Isaura Hernández, Daniel Salazar ("El friky"), Alberto Arturo Flores ("El turi"), Gerardo Ortíz, César Peregrino, Jashiel Pérez, and Jesús Reyes ("El chuy").

I am appreciative to my family, my little brother, parents, and grandparents; to my best friends, Héctor Yéred Ríos, Tania Lucero Hernández, Stefanny Palacios Casarrubias, Zaidy Mayte Rodríguez, Jessica Huerta, Alejandra Moreno, Raúl Pérez, Diego Ramírez, Abraham Avelar, Katia Cortés; to my new friends Marion Ott, Krishnan Srinivasarengan, Layla Mohajer, Rama Alhasan, Mohammed Elismaili, Saliou Diop, Sena Jung, Karol Barragán, Víctor Rodríguez, Bernadeta Jasiok, Thomas Pierrejean, and Racha Djebbar.

And a special appointment to the Antonio Lopez family.

Dedicated to my family
and friends.

Résumé

La demande de développement de systèmes dynamiques sûrs et fiables est devenue de plus en plus importante ces dernières années. Les processus de contrôle/commande deviennent de plus en plus complexes et sophistiqués, en conséquence, les questions de disponibilité, de fiabilité, de sécurité de fonctionnement et de protection de l'environnement sont d'une importance majeure. La synthèse de loi de commande tolérante aux défauts/défaillances est devenue un sujet d'études important aussi bien sur les aspects théoriques que pratiques.

En raison de la complexité accrue et du nombre croissant de composants, les systèmes multi-agents ont vu leur essor d'un point de vue méthodologique. Ces systèmes sont particulièrement sensibles défauts/défaillances, se produisant avec une probabilité plus élevée et entraînant une dégradation des performances ou une panne de tous les agents.

Ce mémoire de thèse présente la conception de méthodes tolérantes aux fautes dévolues aux systèmes multi-agents ou également dénommée résilience de systèmes multi-agents. Au cœur de ce travail, le problème du suivi du leader est considéré dans le but que tous les agents suivent la trajectoire d'un unique agent en dépit des défauts/défaillances occurrent au sein des autres agents.

Les principales contributions de cette thèse se concentrent sur les stratégies de contrôle/commande en présence de différentes entrées inconnues externes considérées comme des défauts et/ou des perturbations :

1) La conception d'un mécanisme déclenché par un événement pour résoudre le problème de contrôle/commande du suivi du leader en réduisant l'échange d'informations entre les agents et le taux de mise à jour du contrôle;

2) La conception d'une commande tolérante aux pannes basée sur des actionneurs virtuels dans des systèmes multi-agents soumis à des défauts d'actionneurs;

3) La conception d'un contrôle/commande de formation déclenché par événement pour les systèmes multi-agents avec des défauts de communication;

4) La conception d'une commande leader-suiveur robuste à des perturbations bornées dans les systèmes multi-agents.

La performance et l'efficacité des stratégies développées dans ce mémoire sont présentées à l'aide d'exemples numériques mais également d'une mise en œuvre d' une flotte expérimentale de véhicules aériens sans pilote (drônes).

Mots-clés: Contrôle tolérant aux pannes, leader-suiveur, défauts de communication, consensus de délimitation quadratique, actionneurs virtuels, contrôle de formation, véhicules aériens sans pilote.

Abstract

Increasing demand for safe and reliable dynamic systems has been becoming an important subject. Modern control systems are becoming more complex and sophisticated, in consequence, the issues of availability, cost efficiency, reliability, operating safety, and environmental protection are of major importance. Fault-tolerant control has become an important subject in modern control theory and practice.

Due to the higher complexity and the increasing number of components, multi-agent systems are particularly sensitive to faults, which can happen with a higher probability and result in performance degradation or breakdown of all the agents.

This thesis presents the design of different fault-tolerant methods based on multi-agent system theoretical framework. The leader-following problem is considered with the aim that all agents follow the trajectory of a leader agent in spite of faults.

The main contributions of this thesis are focused on control strategies for different external unknown inputs considered as faults and/or disturbances:

1) The design of an event-triggered mechanism to solve the leader-following control problem reducing the exchange of information between agents and the control update rate;

2) The design of a fault-tolerant control based on virtual actuators in multi-agent systems subject to actuators faults;

3) The design of an event-triggered formation control for multi-agent systems with communication faults;

4) The design of a quadratic boundedness leader-following control in multi-agent systems subject to bounded disturbances.

The performance and effectiveness of the proposed strategies are shown through numerical examples and implementation in an experimental fleet of unmanned aerial vehicles.

Keywords: Fault-tolerant control, leader-following consensus, communication faults, quadratic boundedness consensus, virtual actuators, formation control, UAVs.

Contents

Chapter 1 General introduction **1**

1.1 Introduction 1

1.1.1 Objectives and scientific contributions 2

1.1.2 Extra scientific contributions 4

1.1.3 Experimental results 4

1.2 Outline of the document 4

Chapter 2
State-of-the-art of multi-agent systems structure and control design

2.1 Introduction 7

2.2 Multi-agent systems 7

2.2.1 Cooperative control tasks in homogeneous mobile multi-agent systems . 12

2.3 Control approaches in multi-agent systems 15

2.3.1 Consensus and leader-following consensus 16

2.3.2 Robust multi-agent systems with disturbances and nonlinearities 17

2.3.3 Event-triggered control 18

2.3.4 Fault-tolerant control 19

2.4 Conclusions 24

Chapter 3
Event-Triggered and Fault-Tolerant Leader-Following Control for Multi-agent Systems

3.1 Introduction 27

3.2 Observer-based event-triggered leader-following control for multi-agent systems 30

3.2.1 Observer-based leader-following control design 31

3.2.2 Event-triggered control mechanism 34

3.2.3 Event-triggered leader-following consensus in second-order multi-agent systems 37

3.3 Observer-based fault-tolerant leader-following control for multi-agent systems . 41

3.3.1 Leader-following consensus for multi-agent systems with actuator faults 41

3.3.2 Observer-based fault-tolerant leader-following control design 43
3.3.3 Effectiveness of the fault-tolerant control strategy through numerical examples 45
3.4 Conclusions 48

Chapter 4
Time- and Event-triggered Formation Control for Multi-agent Systems Under Communication Faults

4.1 Introduction 49
4.2 Time-triggered leader-following formation control design 51
4.3 Event-triggered leader-following formation control design 55
4.4 Comparison of the proposed approach and the classical formation control 55
4.5 Fleet of unmanned aerial vehicles under communication faults 61
4.5.1 Experimental platform description 61
4.5.2 Experimental results 62
4.6 Conclusions 67

Chapter 5
Quadratic boundedness leader-following control for multi-agent systems: Passive FTC approach

5.1 Introduction 69
5.2 Quadratic boundedness consensus in multi-agent systems 71
5.2.1 Controller and observer gains synthesis 74
5.3 Formation control in a fleet of UAVs subject to wind turbulence 75
5.4 Fourth-order lateral F-8 aircraft under bounded faults 81
5.5 Conclusions 83

General conclusions and perspectives

1 Conclusions 85
2 Perspectives 86

Appendixs

1 Lemmas 87
2 Notation and graph theory 87
3 Unmanned aerial vehicle general model 88
4 Controllability, observability, stabilizability, and detectability 89
4.1 Controllability 89
4.2 Observability 89

4.3 Stabilizability 90
4.4 Detectability 90
5 Separation principle 90

References

List of Figures

1.1 Scheme of the outline of this thesis. 5

2.1 Example of a multi-agent scheme. 8
2.2 Motions of groups of animals in nature. 8
2.3 Different architectures in multi-agent systems. 9
2.4 Example of a directed graph. 9
2.5 Different types of communication topologies [1]. 10
2.6 Different types of communication topologies with a leader. 11
2.7 Spacial rendezvous. 13
2.8 Birds flying in formation. 13
2.9 Starling flock. 15
2.10 Event-based control loop [101]. 18
2.11 Distinction between actuator faults, plant faults, and sensor faults. 20

3.1 Event-triggered control scheme in multi-agent systems. 29
3.2 Virtual actuators scheme in multi-agent systems. 29
3.3 Communication topology for simulations in this subsection. 37
3.4 Comparison of the trajectories of all agents between time- and event-triggered control. 38
3.5 Comparison of the velocities of all agents between time- and event-triggered control. 38
3.6 Comparison of the control profile $\Delta u_{i_x}(t)$ between time- and event-triggered control. 39
3.7 Comparison of the control profile $\Delta u_{i_y}(t)$ between time- and event-triggered control. 39
3.8 Comparison of the control profile $\Delta u_{i_z}(t)$ between time- and event-triggered control. 40
3.9 Events of all the agents. 41
3.10 Communication topology for simulations in this subsection. 46
3.11 Free fault case performance of the synchronization error between the leader and the agents. 46
3.12 Comparison of the control performances with and without reconfiguration. . . . 47
3.13 Control law without reconfiguration. 48

4.1 Robust formation control strategy scheme for multi-agent systems under communication faults. 50
4.2 Event-triggered formation control strategy scheme in a fleet of UAVs under communication faults. 50

4.3 Topology of the communication used in these simulations. 56
4.4 Comparison of the classical formation control, time-, and event-triggered strategies. 57
4.5 Comparison of the velocities in the classical formation control, time-, and event-triggered strategies. 58
4.6 Comparison of the desired formation performance. 59
4.7 Comparison of the agents' performance following the leader agent. 60
4.8 Bebop 2 parrot. 61
4.9 Comparison of the trajectories' performance between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. 63
4.10 Comparison of the velocities' performance between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. 64
4.11 Comparison of the consensus' performance between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. 65
4.12 Comparison of consensus control law between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. 66
4.13 Event-triggered profile of the agents. 67
4.14 Number of events in each UAV. 67

5.1 Control scheme for an agent subject to unknown bounded inputs. 71
5.2 Communication topology between agents in these examples. 78
5.3 Profile of the wind turbulence affecting the fleet of UAVs. 79
5.4 Profile of the UAV trajectories. 79
5.5 Profile of the UAV velocities subject to wind turbulence. 80
5.6 Comparison of the desired formation performance. 80
5.7 Profile of the bounded faults. 82
5.8 Comparison of the consensus' performance. 83

1 Quadrotor aircraft conceptual scheme. 88

Chapter 1

General introduction

1.1 Introduction

Multi-agent systems have been of interest due to their potential in accomplishing missions that a single agent cannot perform [1]. Several applications have been developed using multi-agent system approaches including, among others, sensor networks [2], electric power systems [3], robotics [4], vehicle formation [5], and spacecraft flying [6]. Research works have increased the focus on the consensus problem of multi-agent systems considering different dynamics such as: single integrators [7], double integrators [8], high-order integrators [9], linear time invariant [10], or nonlinear dynamics [11]. The objective of synchronization in cooperative multi-agent systems, which is known as consensus, is often to reach a common value in relation to the states of the agents.

First-order and second-order multi-agent systems are used to illustrate mobile multi-agent systems in the context of rendezvous [12], formation control [13], or flocking [14] which are the three main tasks to tackle in cooperative applications. These types of multi-agent systems can exemplify the behavior of autonomous mobile robots [4], unmanned aerial vehicles (UAVs) [5], autonomous underwater vehicles [15], among others, in an Euclidean space. Rendezvous in mobile multi-agent systems is a typical example of coordination which consists of designing distributed controllers driving all the agents towards a common point in the space [12]. Formation control consists to reach desired shapes [16]. Flocking approach refers to the animal movement behavior in a group of agents without collisions inspired mainly by animal movement such as bird flocks, insect swarms, or school fish [17]. This cooperative tasks in mobile multi-agent systems can be achieved by using consensus algorithms.

Modern control systems are becoming more complex with control algorithms more sophisticated, therefore, cost efficiency, reliability, operating safety, and environmental protection have become of major importance. The consequences of faults might be extremely serious in terms of human mortality, environmental impact, and economic loss [18]. There is a need for on-line supervision, fault diagnosis, and its respective fault tolerant to increase the reliability of safety-critical systems [18]. The need to develop more autonomous and intelligent safety systems motives to study the problem of automated fault diagnosis and accommodation [19]. A fault can be defined as an unexpected change of system function although it may not represent physical failure or breakdown. A fault-tolerant control system is designed to maintain some portion of its control integrity in the case of a specified set of possible faults

or large changes in the system operating conditions [20]. This can be done if the control system has built-in an element of automatic reconfiguration, once a fault has been detected and isolated [18].

Due to the higher complexity and the increasing number of components, multi-agent systems are particularly sensible to faults, which can happen with a higher probability and result in performance degradation or breakdown of all the agents [21]. Not only do faults faults affect the performance of the consensus in real applications but also disturbances, uncertainties, or unknown inputs. Rejection of these is one of the objectives in controller design [22]. Multi-agent systems are also subject to these type of undesired inputs which disturb the performance of the consensus [23].

On the other hand, as a result of the rapid development of computer technologies, discrete-time systems are becoming increasingly important due to embedded microprocessors. The system computes every sampling time the control law in the classical time-driven control. In many cases, the time-triggered approaches make changes in the control law provoking unnecessary energy consumption. In this spirit, event-based control approaches have been of interest in order to handle these disadvantages [24]. The controller is not executed unless it is required in event-based control [25]. Event-triggered approaches have been employed in multi-agent systems in order to reduce the broadcasting in the communication and the update control rate [26]–[28]. Consequently, the energy consumption in each agent is reduced, and the bandwidth limitation is well managed. However, event-triggered control strategies and fault-tolerant control for multi-agent systems under actuator, sensor, or communication faults have been recently explored.

The present thesis addresses the problem of multi-agent systems subject to communication and actuator faults. Then, fault-tolerant control strategies and robust control based on linear matrix inequalities conditions (LMIs) and the Lyapunov stability analysis are designed. The main contributions of this thesis are: 1) the design of an event-triggered leader-following control in multi-agent systems in order to reduce the information exchange and the control update rate; 2) the design of distributed virtual actuators subject to actuator faults; 3) the design of an time- and event-triggered formation control in a fleet of UAVs under communication faults; and 4) the design of a passive fault-tolerant control based on quadratic boundedness consensus considering bounded faults. In this light, the following objectives of this thesis are listed.

1.1.1 Objectives and scientific contributions

The main objective of this research is to develop fault-tolerant methods for multi-agent systems under communication/actuator faults. Unlike the classical fault-tolerant problem, there is no centralized controller handling global information. The agents have only local knowledge of its states and information of their neighboring agents' states. The following research objectives will be achieved:

- Development of an event-triggered control strategy for multi-agent systems subject to communication faults.
- Development of a fault-tolerant control strategy for multi-agent systems under multiplicative/additive actuator faults.

The scientific contributions of this thesis are listed by Chapter as follows:

Chapter 3: This chapter presents the following problems separated in two parts: the first problem is to reduce the information exchange between agents and the control update rate using an event-triggered mechanism; the second problem is to design an effective fault-tolerant control for multi-agent systems under actuator faults. The main contributions are the design of an event-triggered mechanism for leader-following multi-agent systems and the design of distributed virtual actuators for multi-agent systems subject to actuator faults. These results have been presented in the following publications:

- J. A. Vazquez Trejo, D. Rotondo, M. Adam Medina, and D. Theilliol, "Observer-based Event-triggered Model Reference Control for Multi-agent Systems," 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, June 2020, pp. 421-428.

- J. A. Vazquez Trejo, D. Rotondo, M. Adam Medina, and D. Theilliol, "Observer-based fault-tolerant leader-following control for multi-agent systems," 2021 European Control Conference (ECC), 2021, Accepted.

Chapter 4: This chapter addresses the design of a time- and event-triggered formation control for multi-agent systems under communication faults which are modeled as smooth-varying delays dependent on the distance between agents. The main contributions are the design of two controllers. One of the control strategy synthesizes a robust control gain which guarantees stability of the synchronization error in spite of faults in the information exchange. The second control strategy extends the first mentioned strategy adding an event-triggered mechanism based on the closed-loop system in order to reduce the information exchange and the control update rate. Both strategies have been implemented in an experimental platform of a fleet of UAVs. These results have been presented in the following publications:

- J.A. Vazquez Trejo, D. Theilliol , M. Adam Medina, C.D. García Beltrán, and M. Witczak (2021) Leader-Following Formation Control for Networked Multi-agent Systems Under Communication Faults/Failures. In: Korbicz J., Patan K., Luzar M. (eds) Advances in Diagnostics of Processes and Systems. Studies in Systems, Decision and Control, vol 313. Springer, Cham.

- J. A. Vazquez Trejo, A. Guenard, M. Adam Medina, J.-C. Ponsart, L. Ciarletta, D. Rotondo, and D. Theilliol, "Event-triggered Leader-following Formation Control for Multi-agent Systems Under Communication Faults: Application to a Fleet of Unmanned Aerial Vehicles," Journal of Systems Engineering and Electronics, Accepted.

Chapter 5: The problem under consideration in this chapter is to reach a quadratic boundedness leader-following consensus when the agents are affected by external disturbances/faults. The main contribution lies in developing a quadratic boundedness leader-following consensus protocol for multi-agent systems subject to exogenous bounded disturbances/faults. This result has been presented in the following publication:

- J. A. Vazquez Trejo, D. Rotondo, D. Theilliol, and M. Adam Medina, "Observer-based quadratic boundedness leader-following control for multi-agent systems". International Journal of Control, submitted.

1.1.2 Extra scientific contributions

A work that has not been reported in any chapter of this document is the design of a robust observer-based leader-following consensus control for a class of nonlinear multi-agent systems. The problem under consideration is to design and synthesize an effective controller such that the synchronization and estimated errors are stable for multi-agent systems which have an additive nonlinearity dependent on the neighboring and local states of the agents. The main contributions are the design of a robust leader-following control for nonlinear multi-agent systems and linear matrix inequality (LMI)-based sufficient conditions to compute the controller and observer gains. This result has been presented in the following publication:

- J. A. Vazquez Trejo, D. Rotondo, M. Adam Medina, and D. Theilliol, "Robust observer-based leader-following consensus for a class of nonlinear multi-agent systems: application to UAV formation control," 2021 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 2021, accepted.

1.1.3 Experimental results

Experimental results have also been carried out in this thesis at the Crativ'lab in Loria (`https://creativlab.loria.fr/en/`). The algorithms presented in Chapter 4 have been tested on a fleet of unmanned aerial vehicles (UAVs) subject to communication faults. The implementations results can be found in the following video `https://youtu.be/Lo_kuGY9Wq4`.

1.2 Outline of the document

In order to understand how to read the thesis document, Fig. 1.1 shows a flowchart about the order that must be followed to understand each chapter. The thesis document is organized as follows. Chapter 2 introduces a state-of-the-art in multi-agent systems, a basic graph theory, tasks of mobile multi-agent systems, and different problems found in the literature. Chapter 3 presents the design of an event-triggered and a fault-tolerant leader-following control for multi-agent systems. Chapter 4 addresses the design of a time- and event-triggered formation control for multi-agent systems under communication faults. Chapter 5 develops a quadratic boundedness leader-following consensus protocol for multi-agent systems subject to exogenous bounded disturbances/faults. Finally, the main conclusions of this thesis and perspectives are drawn.

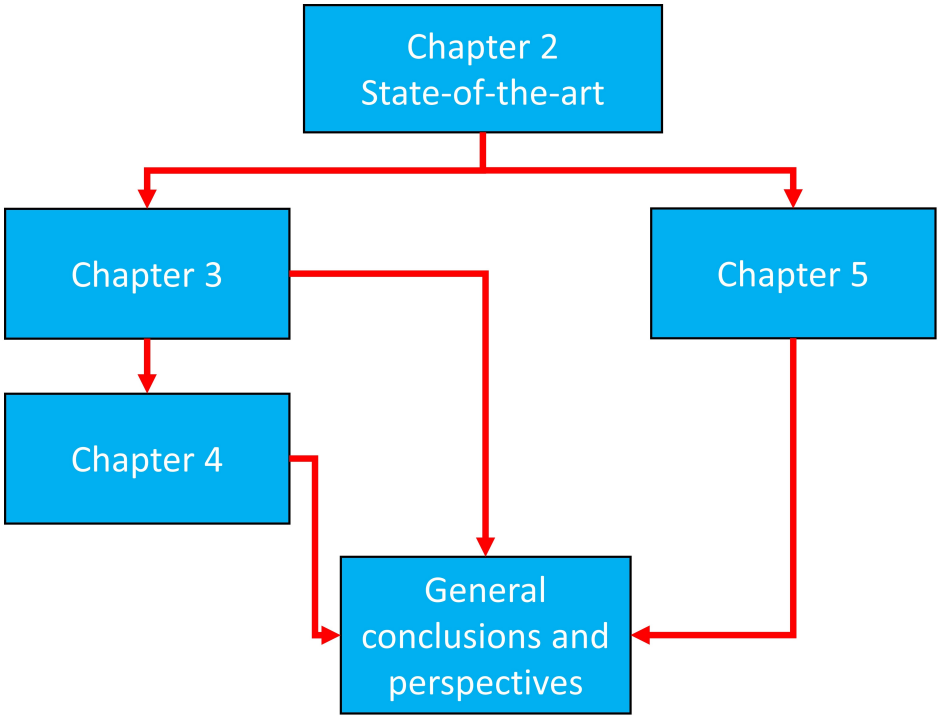


Fig. 1.1: Scheme of the outline of this thesis.

Chapter 2

State-of-the-art of multi-agent systems structure and control design

2.1 Introduction

This chapter introduces multi-agent systems, a basic graph theory, tasks of mobile multi-agent systems, and different approaches found in the literature of main problems in multi-agent systems. A definition of multi-agent systems is given in order to understand the parts that make them up. It is worth mentioning that only homogeneous multi-agent systems are considered in this thesis. Due to the characteristic of exchanging information through a communication topology, basic concepts of graph theory are presented in order to model the interaction between the agents. As a result of the communication topology, certain Laplacian matrix properties are analyzed. Multi-agent systems can be classified as mobile or non-mobile systems. In both type of systems, the control objective is to achieve a common value, however, in non-mobile systems is not possible to imagine the behavior of the synchronization. For this reason, tasks of mobile multi-agent systems using consensus algorithm are given. Finally, a state-of-the-art of different control approaches in multi-agent systems subject to unknown inputs such as nonlinearities or disturbances, communication, actuators, sensors, or parameter faults, with and without leader agent, is considered.

2.2 Multi-agent systems

Multi-agent systems are composed by individual dynamic systems that can communicate each other through networks. An individual agent has actuators, sensors, and a computer for relating to its own environment. Each agent has a communication protocol which allows the exchange of information with neighboring agents [1]. The exchange of information between agents through networks is the main characteristic of multi-agent systems [29]. Fig. 2.1 represents a scheme of multi-agent systems sharing their information through a communication topology. Nevertheless, bandwidth limitations, delays, or packet losses are challenges in real engineering applications [30].

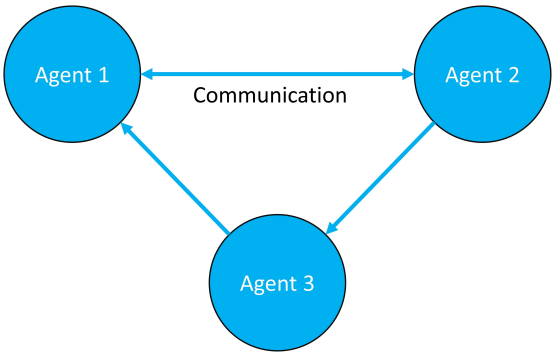


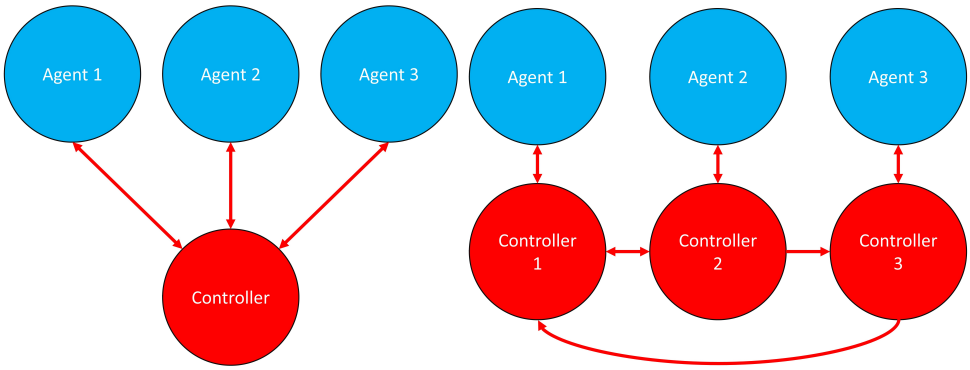
Fig. 2.1: Example of a multi-agent scheme.

Multi-agent systems have been inspired by the behavior of certain animals such as flocks of birds, swarms of insects, or school of fishes. The first reported computer simulations of these animal's behaviors are presented in [17] and [31] moving particles in two dimensional space (see Fig. 2.2). In nature, collective motions allow the group to achieve a common objective. The benefits of motion include defense from predators or forage for food. For example, motions of groups of lions, wolves, and jackals, allow a more efficient pursuit of prey [32].

(a) Birds flying in V (b) Flock of birds

Fig. 2.2: Motions of groups of animals in nature.

For achieving a desired coordination objective between agents, centralized and distributed architectures have been adopted to design appropriated controllers [1]. The centralized architectures assumes that at least a central hub collects information and send control signals to all the agents. In the distributed architectures, each agent exchanges information with its neighbors for computing the control law (see Fig. 2.3) [33]. In the centralized architectures, at least one controller is used to coordinate all the agents. In the distributed architectures, there is a flow of information through communication topologies which allows using more controllers. "*Shared information is a necessary condition for cooperation*" [33].



(a) Centralized architecture (b) Distributed architecture

Fig. 2.3: Different architectures in multi-agent systems.

In order to model the communication relationship between agents in distributed architectures, the graph theory is taken. A graph is a representation of a group of nodes, where each node is the graphical representation of an agent, and some of them are connected. A graph $\mathcal{G}$ is an ordered pair $(\mathcal{V}, \mathcal{E})$ consisting of a non-empty finite node set $\mathcal{V}$ and a set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ of links. A link is an ordered pair of two nodes. A directed link (i, j) is an incoming link to node j and an outgoing link node i. It is said that node i is a neighbor of node j if the link (i, j) exists in the graph $\mathcal{G}$. $\mathcal{N}_i = \{j : (i, j) \in \mathcal{E}\}$ denotes the set of neighbors of node i [34]. Each node represents each agent, and each link denotes the flow of information between agents (see Fig. 2.4). After this explanation each node will be called an agent.

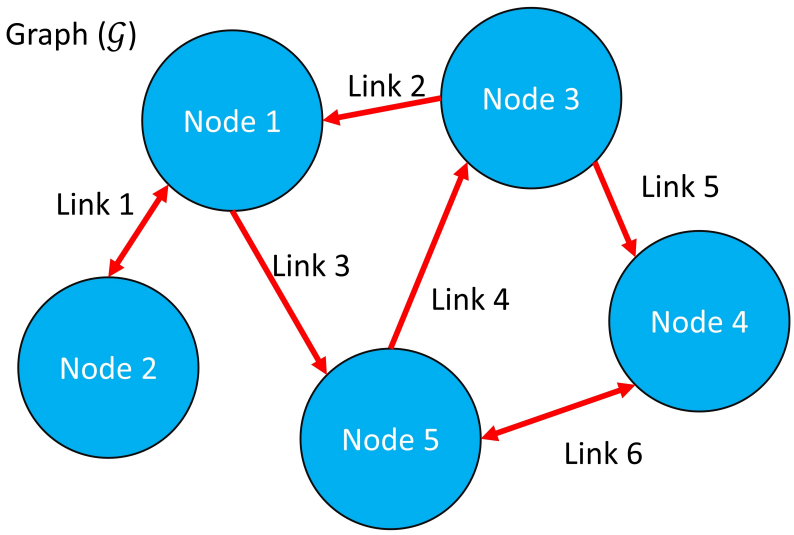
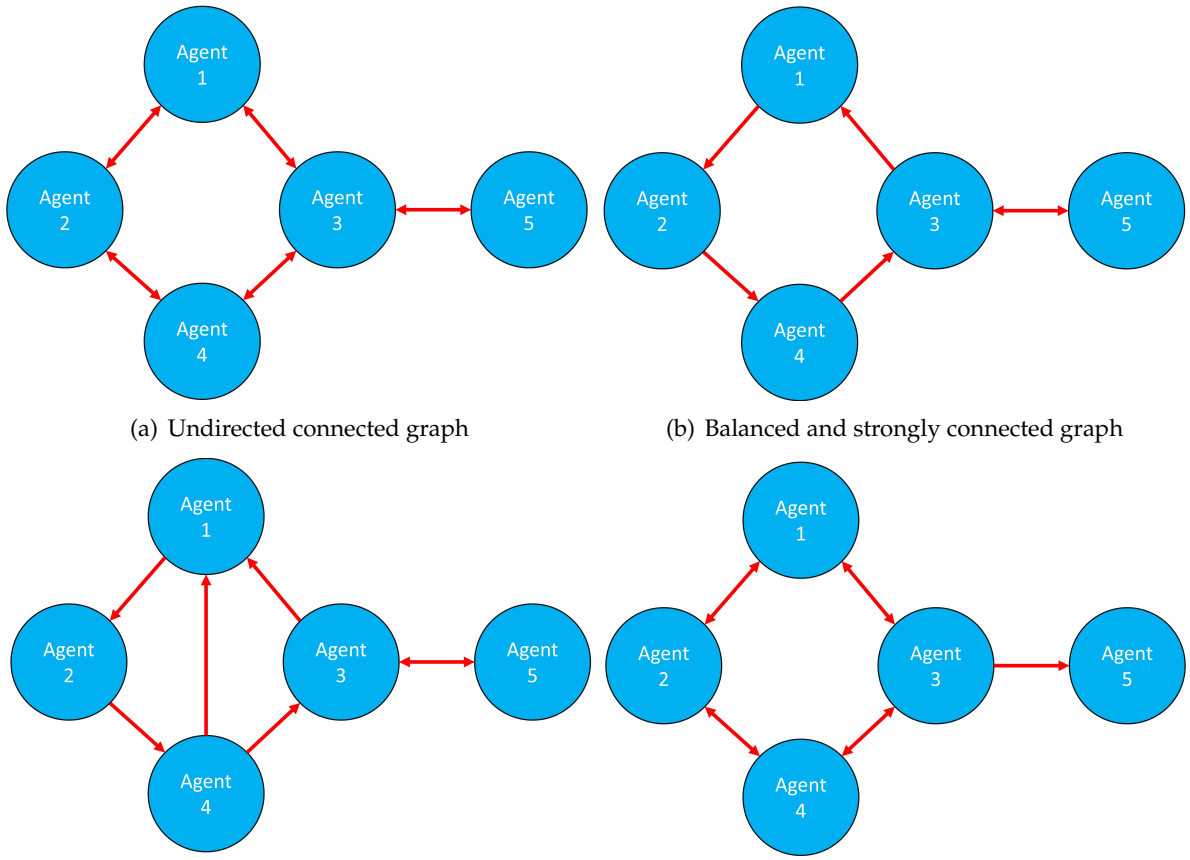


Fig. 2.4: Example of a directed graph.

There are several types of network topologies depending on the way the agents are relating to each other. Fig. 2.5 shows different examples of communication topologies. A directed link is denoted by a line with a directional arrow while an undirected link is denoted by a bidirectional arrow.



(a) Undirected connected graph

(b) Balanced and strongly connected graph

(c) Strongly connected graph

(d) Weakly connected graph

Fig. 2.5: Different types of communication topologies [1].

An undirected tree is an undirected graph (see Fig. 2.5(a)) where all the agents can be connected by the way of a single undirected path. A directed graph is strongly connected (see Fig. 2.5(b,c)) if there is a directed path from every agent to every other agent. A directed graph is complete if there is an link from every agent to every other agent. A directed tree is defined as spanning when it connects all the agents in the graph. A graph contains a directed spanning tree if a subset of the links forms a directed spanning tree which is equivalent saying that the graph has at least one agent with directed paths to all other agents. If there is a spanning tree in a undirected graph then it is equivalent to being connected. However, in directed graphs, the existence of a directed spanning tree is a weaker condition (see Fig. 2.5(d)) than being strongly connected. A strongly connected graph contains at least one directed spanning tree [1].

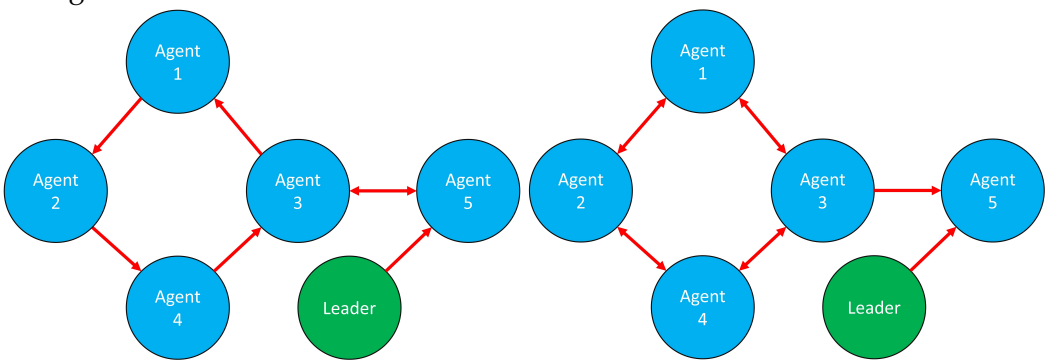
The adjacency matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N\times N}$ associated with the graph $\mathcal{G}$ is defined such that $a_{ii} = 0$, $a_{ij} > 0$ if and only if $(i,j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. The Laplacian matrix $\mathcal{L} = [\mathcal{L}_{ij}] \in \mathbb{R}^{N\times N}$ of the graph $\mathcal{G}$ is defined as $\mathcal{L}_{ii} = \sum_{j\neq i} a_{ij}$ and $\mathcal{L}_{ij} = -a_{ij}, i \neq j$. For example from Fig. 2.5 let us compute the corresponding Laplacian matrices as follows

$$\mathcal{L}_a = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 \\ -1 & 2 & 0 & -1 & 0 \\ -1 & 0 & 3 & -1 & -1 \\ 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}, \mathcal{L}_b = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & -1 & -1 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix},$$

$$\mathcal{L}_c = \begin{bmatrix} 2 & 0 & -1 & -1 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & -1 & -1 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}, \mathcal{L}_d = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 \\ -1 & 2 & 0 & -1 & 0 \\ -1 & 0 & 2 & -1 & 0 \\ 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}.$$

Topologies allow to model the information flow between the agents in order to solve synchronization problems. A spanning tree topology is a weak topology which considers a agent root $\mathcal{L}_d$. If there is a spanning tree in the graph, the Laplacian matrix is semi-definite positive with one eigenvalue zero. A strongly connected topology have a directed path from every agent to every other agent ($\mathcal{L}_b$ and $\mathcal{L}_c$). An undirected topology is a particular case where the communication is bidirectional ($\mathcal{L}_a$). The Laplacian matrix of a undirected graph, besides having the properties of a spanning tree graph, is symmetric. Undirected and strongly connected topologies give a strong relationship between the agents.

In the literature, research activities can be found where all agents follow a common leader. Then, the aforementioned graphs are modified to consider the leader in the communication topology. Within the topology where a leader is considered, one or more follower agents receive the information of the leader's states. This leader does not receive information from any of the followers, so to guarantee the convergence of all the followers to the leader (called leader-following consensus), the leader must be considered as a root in the communication topology. Because of this, spanning tree topologies must ensure that even a leader is considered the topology is hold if not the consensus cannot be reached as Fig. 2.6(b) shows due to the leader is not a root in the topology. In strongly connected and undirected graphs, the followers can follow the leader because there is a spanning tree if and only if there is a directed path from one agent to another one as Fig. 2.6(a) shows due to the leader can transmit its information in any of the agents.



(a) Example of strongly connected graph with a leader where the consensus can be achieved.

(b) Example of weakly connected graph with a leader where the consensus cannot be achieved because the leader is not a root.

Fig. 2.6: Different types of communication topologies with a leader.

Within the possible multi-agent classifications, mobile or non-mobile, homogeneous or heterogeneous, are the most studied in the literature. Mobile agents are those systems that move in a space of two or three dimensions. Non-mobile agents are those systems that only synchronize their states in a common value. The term homogeneous means that the dynamics of all agents are assumed identical for all of them. The term heterogeneous indicates that the dynamics of all agents are assumed differently [35]. In the following section, some tasks in homogeneous mobile multi-agent systems are presented which perform the objective of rendezvous, formation control, and flocking with and without a leader.

2.2.1 Cooperative control tasks in homogeneous mobile multi-agent systems

In many cooperative control applications, it is almost impossible to visualize the performance of the agents due to the nature of the system. The objective of synchronization of cooperative multi-agent systems is often to reach a common value in relation to the states of the agents. As described above, an agent is a dynamic system able to communicate its states with other agents. Most of the multi-agent systems are smart grids [3], drones [5], and sensor networks [2], among others. Then, multi-agent systems can be classified as mobile or non-mobile. This is why first-order or second-order multi-agent systems are used to illustrate mobile multi-agent systems in the context of rendezvous, formation control, or flocking which are the three main tasks to tackle in cooperative applications. These types of multi-agent systems can exemplify the behavior of autonomous mobile robots, unmanned aerial vehicles (UAVs), autonomous underwater vehicles, among others, in an Euclidean space. In the following subsections, a brief of each approach is addressed.

a) Rendezvous

Rendezvous in mobile multi-agent systems is a typical example of coordination which consists of designing distributed controllers driving all the agents towards a common point in the space [12]. In [36], an event-triggered rendezvous problem of multi-agent systems with limited communication range the is studied. In [37], a multi-agent Q-learning based on rendezvous problem of cognitive radios quickly finding each other is addressed to establish communication in a multi-channel dynamic spectrum access environment. In [38], a finite-time rendezvous control is investigated for first order multi-agent systems with preserving topology connectivity and constrained energy. In [7], a distributed leader-following rendezvous with connectivity preservation for a linear multi-agent system where the leader system is a linear autonomous system and the follower system is a multiple single-integrator system is developed. In [39], discrete-time model predictive control algorithms to achieve stable consensus or rendezvous are addressed for distributed double-integrator multi-agent systems with directed switching topologies and input constraints. In Fig. 2.7, a pair of satellites is illustrated making a spacial rendezvous.

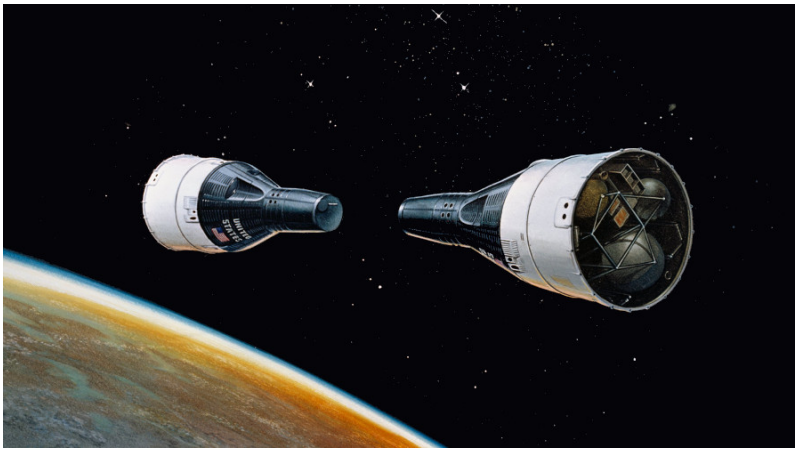


Fig. 2.7: Spacial rendezvous.

The objective of the agents is to achieve a common point in the space, unlike formation control, where it is necessary to achieve a desired formation.

b) Formation control

Formation control in multi-agent systems consists to reach desired shapes [16]. This approach has been inspired by the behavior of some animals as seen in Fig. 2.8 which consists of moving from one place to another using a defined formation[40].

Fig. 2.8: Birds flying in formation.

Formation control is categorized with the existing results based on position, displacement, and distance according to types of sensed and controlled variables. Table 2.1 presents the different types to solve the formation control. Based on position, agents sense their own positions with respect to a global coordinate system [41], [42]. They manage their own positions to achieve the desired formation described by the desired positions with respect to the global coordinate system. Based on displacement, agents control displacements of their neighboring agents to achieve the desired formation specified by the desired displacements with respect to a global coordinate system under the assumption that each agent is able to sense relative positions of its neighboring agents with respect to the global coordinate system. Based on distance, the agent distances are controlled to achieve the desired formation given by the desired distances. Individual agents sense relative positions of their neighboring agents with respect to their local position [13].

Table 2.1: Distinctions of formation control based on position, displacement, and distance [13].

	Position	Displacement	Distance
Sensed variables	Agent positions	Relative neighboring agent positions	Relative neighboring agent positions
Controlled variables	Agent positions	Relative neighboring agent positions	Inter-agent distances
Coordinate systems	Global	Local	Local
Interaction topology	Usually not required	Connectedness or existence of a spanning tree	Rigidity or persistence

In [43], formation control and obstacle-collision-avoidance problem using distributed model predictive control is used for each agent to track its reference trajectory is presented of two multi-unmanned aerial systems. In [44], a tracking control with switching formation in nonomniscient constrained space for multi-agent systems is presented. In [45], an optimal formation control for multi-agents systems with external disturbances is considered.

On the other hand, consensus algorithms have been widely used in formation control such as in [46]–[49], which consider a topological communication between the agents to define a control law. In [46], a leader-follower consensus formation control is proposed in five wheeled mobile robots. In [47], the cooperation problem in a group of vehicles performing a shared task using inter-vehicle communication to coordinate their actions is considered. In [48], extensions of a consensus algorithm are introduced for systems modeled by second-order dynamics to solve formation control problems by appropriately choosing information states. In [49], a distributed observer-based consensus control for multi-agent systems with a time-invariant communication topology consisting of general linear agent dynamics is addressed. In [50], a rigid formation control problem with switching topology is studied. Works [16], [40]–[42] have studied that consensus algorithms can be extended to solve the formation control problem for mobile multi-agent systems by correctly satisfying $Ah_i = 0$, $\forall i = 1, 2 \ldots, N$, where A is the system dynamics' matrix, h_i is the desired configuration of the final formation for agent i.

Formation control achieves a rigid desired formation, unlike flocking, which is more flexible in the final formation.

c) Flocking

Inspired mainly by animal movement such as bird flocks, insect swarms, or school fish; flocking approach refers to the movement animal behavior in a group of agents without collisions (see Fig. 2.9). Reynolds in 1987 [17] defined three heuristic rules to simulate the flocking behavior for first time in a computer which consist of the following 3 main elements:

Separation: avoiding collisions between agents.
Alignment: seeking a common agent velocity.
Cohesion: staying close between agents.

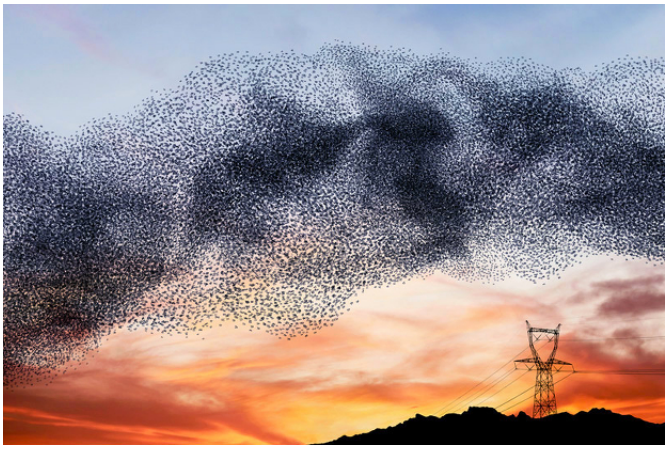


Fig. 2.9: Starling flock.

Based on the basic rules proposed by Reynolds, several related works have been generated. The paper reported in [31] focuses mainly on the alignment of particles by averaging the direction of their velocities. Together with Vicseck, [51]–[53] make up the first groups working in flocking. In [51], a basic theory of flocking is presented. A mathematical model of collective motion for the flocking behavior is studied in [52]. A discrete model of the motion of particles is investigated in [53]. In [54], a biologically-inspired intelligent controller based on a computational model of emotional learning in mammal's brain is employed for flocking control of multi-agent systems. In [55], a flocking control of a group of mobile agents having leader-follower configuration is presented with an artificial potential function to design the distributed control law of the agents. In [56], [57], a model predictive flocking control scheme for the Cucker-Smale multi-agent model with and without input constraints are developed. In [58], a class of generalized flocking algorithms based on the Olfati-Saber's results for self-driving multi-agent networks. In [59], a second-order multi-agent system with communication network determined by a metric rule based on a random interaction range is studied to determine a bound on the probability that the agents asymptotically agree on a common velocity. In [14], a theoretical framework for design and analysis of distributed flocking algorithms is presented. A flocking implementation in a fleet of UAVs is presented in [60].

Rendezvous, formation control, and flocking can be solved using consensus algorithms or not. Then, the consensus algorithms are used to exemplify the performance of the proposed control strategies in formation control.

2.3 Control approaches in multi-agent systems

The control objective in multi-agent systems are consensus and leader-following consensus. In both control objectives, the agents must achieve a common value in relation to their states. However, consensus is an agreement in relation to the states of the other agents and in leader-following consensus, the agents must follow the trajectories of a leader agent. In the following subsections, the main approaches of consensus and leader-following consensus for multi-agent systems focused on control strategies for different external unknown inputs considered as faults and/or disturbances are presented.

2.3.1 Consensus and leader-following consensus

Consensus is an important problem in the context of multi-agent systems where all the agents reach an agreement in relation to the states of the other agents [61]. A consensus algorithm or consensus protocol is an interaction rule that specifies the exchange of information between an agent and its neighbors on the network [62]. Pioneering works on the consensus for distributed making and parallel computing can be found in [63], [64]. The interest in consensus control is mainly stimulated by [62] and [65].

The multi-agent systems are said to achieve consensus if $\lim_{t\to\infty} \left\| x_i(t) - x_j(t) \right\| = 0$, $\forall i \leq j = 1, 2, \ldots, N$, where $x_i(t)$ and $x_j(t)$ are the states of all the agents, and N is the total number of agents [66]. Several research works have focused on the consensus problem such that distributed optimal with completely unknown dynamics [67], distributed quantized $\mathscr{H}_\infty$ consensus under switching weighted undirected or balanced directed topologies [68], or discrete- and continuous-time multi-agent systems with distance-dependent communication networks [69], just to mention some works.

On the other hand, leader-following consensus is a particular case obtained when all the trajectories of the agents must converge to the trajectory of a physical or virtual leader [70]. In the last decades, the leader-following consensus problem has been studied extensively considering time delays [71], switching topologies [72], [73], irregular discrete sampling times [74], additive and multiplicative noises [75], measurement noises [76], particle swarm optimization [77], multiple uncertain Euler-Lagrange systems [78], nonlinear multi-agent systems [79]–[81], high-order multi-agent systems [82], , and event-triggered mechanisms [27], [83], [84], among others. The leader-following control problem is solved if the multi-agent system satisfies $\lim_{t\to\infty} \|x_i(t) - x_r(t)\| = 0$, $\forall i = 1, 2, \ldots, N$, where $x_r(t)$ is the leader's states. In the last decades, the leader-following consensus has been employed in fleets of UAVs when the control objective of the collective group is to follow the trajectories and velocities described by a physical or virtual agent such as reported in [27], [77] considering second-order multi-agent systems, or multi-UAV-systems [43], [85].

The most studied problems in the context of multi-agent systems are consensus and leader-following consensus [61], [86]. Consensus and leader-following consensus can be compared with the regulation and tracking problem in conventional single systems. Consensus is to reach an agreement between agents in relation to the states of the other agents as regulation in a single system without reference. Leader-following consensus is to follow a leader agent which can be seen as tracking a reference. The leader-following consensus allows to synchronize a team of agents through a virtual/physical leader agent. For the reason, leader-following consensus is taken in order to manipulate the multi-agent system through a leader.

Nevertheless, it is well known that in real implementations there are several challenges to consider, such as faults in actuators, sensors, parameters or in communication; disturbances, parameter uncertainties, or measurement noise, which affect the performance of the consensus. These challenges in homogeneous multi-agent systems are presented in the following subsections.

2.3.2 Robust multi-agent systems with disturbances and nonlinearities

The performance of the consensus is often affected by external disturbances [23]. In this sense, $\mathscr{H}_\infty$ optimization problem has been studied for high-order multi-agent systems with Lipschitz nonlinearities and switching topologies [87], Lipschitz nonlinear multi-agent systems with uncertain dynamics[88], event-based strategy in second-order multi-agent systems [89], discrete-time nonlinear multi-agent systems with missing measurements [90], delays and parameter uncertainties [91]. Another control technique used for multi-agent systems with external disturbances is adaptive control. Disturbance rejection in multi-agent with actuator failures is investigated in [92]; disturbance-compensation based on learning control is proposed in [93]; nonlinear multi-agent systems with state constraints are investigated in [94], including deterministic disturbances [95], or bounded external disturbances [96], among others. Generally, those strategies can be divided into adaptive or $\mathscr{H}_\infty$ optimization problem. An alternative found in the control theory to solve the exogenous disturbances problem is the quadratic boundedness in systems with bounded disturbances. It is said that a system is quadratically bounded if all its solutions are bounded and this behavior can be guaranteed with a quadratic Lyapunov function [97]. Quadratic boundedness approaches have been used in fault estimation [98], optimal estimation of unmanned aerial vehicles [99], or feedback stabilization of Takagi-Sugeno systems [100] to compensate bounded external disturbances. Nevertheless, the quadratic boundedness consensus has not been found in the literature in order to solve such problem in multi-agent systems. The quadratic boundedness consensus can be applied to design a robust controller and observer in multi-agent systems affected by bounded disturbances so that both the synchronization and the estimation error are ultimately inside an invariant ellipsoid.

Another problem is examined when this input is nonlinear. A centralized and a decentralized event-triggered leader-following consensus for a class of switched nonlinear multi-agent are investigated in [83]. The distributed leader-following consensus problem for a class of first-order and second-order Lipschitz nonlinear multi-agent systems with unknown control directions and unknown bounded external disturbances is addressed in [80]. The distributed event-triggered consensus problem for a class of nonlinear multi-agent systems subject to actuator saturation considering an uncertain nonlinear function is presented in [81]. A sampled-data leader-following consensus of nonlinear multi-agent systems with random switching network topologies and communication delay is studied in [73]. In [83] and [80], heterogeneous nonlinear multi-agent systems have been contemplated where, in [83], the nonlinearity is different for each agent and [80] considers only different values in the agents' parameters. [81] and [73] present high-order nonlinear homogeneous multi-agent systems with a linear and an additive nonlinear part. However, in [73], [80], [81], [83], the additive nonlinearity is Lipschitz and only dependent on the local states of each agent. Until now, a nonlinearity dependent on states of neighboring agents has not been considered.

An alternative for reducing the exchange of information between agents and the update control rate is the event-triggered control. In the following subsection, works in the context of event-triggered control for multi-agent systems are presented.

2.3.3 Event-triggered control

As a result of the rapid development of computer technologies, discrete-time systems are becoming increasingly important due to embedded microprocessors. It is well known that discrete-time systems only sample signals at discrete-time. One crucial issue is how to select an appropriate sample time. The system computes every sampling time the control law in the classical time-driven control. In many cases, the time-triggered approaches make changes in the control law provoking unnecessary energy consumption. In this spirit, event-based control approaches have been of interest in order to handle these disadvantages [24]. In event-based control, the controller is not executed unless it is required [25]. Fig. 2.10 shows the main components of an event-based control loop which consists of a plant, the controller (continuous-time or discrete time), and an event generator. The communication links are only used after the occurrence of an event at time t_k and the current output $y(t_k)$ (or state $x(t_k)$) is transmitted to the controller. The controller determines the new input u_k, which is conditioned by the control input generator to determine the continuous input $u(t)$ in the time interval $[t_k, t_{k+1})$ until the next event occurs at time t_{k+1} [101].

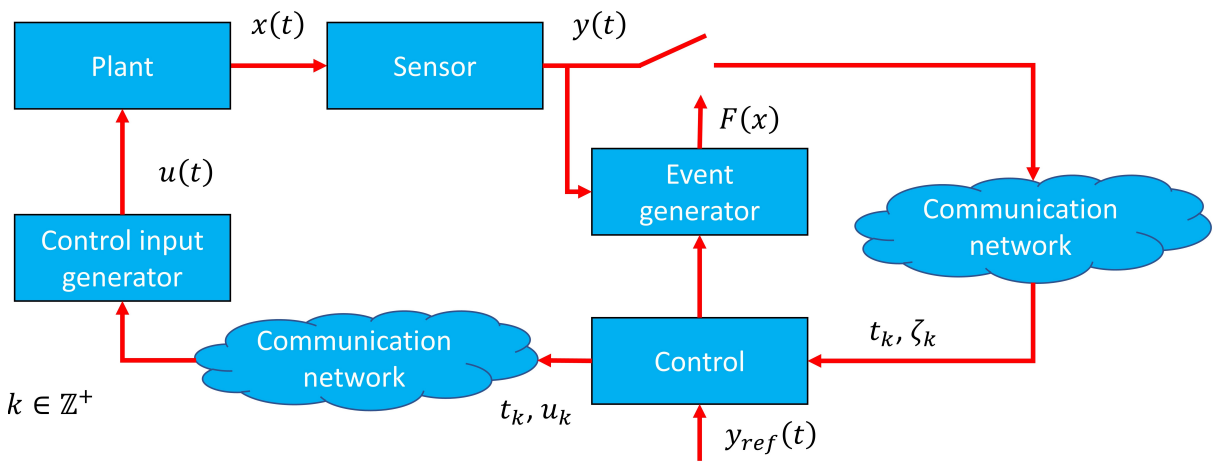


Fig. 2.10: Event-based control loop [101].

The event generator determines the time instants t_k, $k \in \mathbb{Z}^+$, at which the next communication between the event generator, the controller, and the control input generator is applied, and the information that is communicated from the sensor to the controller. The control input generator establishes the signal $u(t)$ for the time interval $t \in [t_k, t_{k+1})$ in dependence on the information obtained at time t_k.

A critical issue to be avoided in event-triggered control is Zeno behavior. Zeno behavior makes to the occurrence of an infinite number of discrete transitions (events) within a finite time interval. Hence, Zeno behavior is extremely undesired in event-triggered control [102], since it causes invalidation of theoretical analysis due to the nonexistence of solutions after some finite instant, and generates to an excessive waste of communication and computational resources [103].

Event-based control is a control methodology often used for reducing the communication load of a digital network. An overview of event-triggered consensus in multi-agent systems has been presented in [104]. Fixed-time event-triggered consensus problems are studied in [105], [106]. In [107], a dynamic event-triggered leader-following control under a directed

communication topology is designed. The broadcasting between each agent is reduced and is not considered continuous in time. The event-triggered tracking control problem of non-linear second-order multi-agent systems is investigated in [108] using the distributed sliding-mode control approach to decrease the controller sampling and save the network communication. In [109], a distributed robust event-triggered tracking methodology for synchronization of multiple uncertain under-actuated surface vessels under limited communication range is proposed. In [110], the event-triggered leader-follower consensus control problem for linear continuous-time multi-agent systems is investigated. A consensus protocol and an event-triggered communication strategy based on a closed-loop estimator are designed and applied to the problem of vehicle platooning. In [111], a secondary voltage control with distributed event-triggered sliding mode control for DC microgrid is presented which synchronizes the voltage magnitude of each follower. In [112], the synchronization of multiple memristive neural networks under cyber-physical attacks through distributed event-triggered control is investigated. A Lyapunov function and M-matrix properties are used to get the criteria for synchronization under cyber-attacks. An event-triggered formation tracking problem of nonholonomic mobile robots is studied in [113] introducing a super-twisting sliding mode differentiator. An event-triggered leader-following consensus for general linear fractional-order multi-agent systems subject to input delays is investigated in [114]. A mean square leader-following consensus for stochastic multi-agent systems using a distributed event-triggered mechanism with time-varying delay is addressed in [115]. An adaptive leader-following consensus using a centralized event-triggered function for a class of heterogeneous first-order multi-agent systems is studied in [116]. A reduced-order observer-based consensus for multi-agent systems with time delay an event-triggered strategy is investigated in [117]. An event-triggered containment control for first-order and second-order multi-agent systems with constant delays is presented in [118]. The importance of event-triggered control lies in providing an effective means to avoid continuous communication when the limited resources are considered (embedded processors or restricted bandwidth) [107], [108].

Event-triggered approaches have been employed in multi-agent systems in order to reduce the broadcasting in the communication and the update control rate [26]–[28]. Consequently, the energy consumption in each agent is reduced, and the bandwidth limitation is well managed. It should be noticed that event-triggered control strategies and fault-tolerant control for multi-agent systems under actuator, sensor, or communication faults have been recently explored. The next subsection is totally focused on fault-tolerant control for multi-agent systems.

2.3.4 Fault-tolerant control

An increasing demand for safe and reliable dynamic systems has been becoming an important subject. Modern control systems are becoming more complex and sophisticated, in consequence, the issues of availability, cost efficiency, reliability, operating safety, and environmental protection are major importance. The consequences of faults can be extremely serious in terms of human mortality, environmental impact, and economic loss. Therefore, there is a growing need for on-line control. A fault can be defined as an unexpected change of system function although it may not represent physical failure or breakdown. In order to avoid production deterioration or machine damage faults must be found to stop the propagation [20]. A fault-tolerant control system is designed to maintain some portion of

its control integrity in the case of a specified set of possible faults or large changes in the system operating conditions. This can be done if the control system has built-in an element of automatic reconfiguration, once a fault has been detected and isolated [18]. Fault-tolerant control has becoming an important subject in modern control theory and practice. A fault-tolerant control may not offer an optimal performance in a strict sense for normal system operation, but generally the fault-tolerant control can mitigate effects of faulty system component without completely compromising the mission or putting the users at risks. It is important to emphasize that when a fault occurs in a system, either in sensors or actuators, the characteristics of the entire system can submit to significant changes, i.e., degradation [119].

In fault-tolerant control systems, one of the important issues is to maintain the system performance close to the desirable one and to preserve stability conditions after occurrence of a fault [120]. Fault-tolerant control systems can be classified into two types: passive and active. Passive controllers are fixed and designed to be robust against faults, and they are very restrictive because all the expected faults cannot be known *a priori* [119]. Active controllers react actively to the faults by reconfiguring the controller so that the stability and an acceptable performance of the system can be ensured [120]. These controllers consist of adjusting the controllers on-line according to the fault magnitude in order to retain the closed-loop system performance [119].

Disturbances and model uncertainties change the plant behavior like faults. From the fault-tolerant control point of view, when faults are represented as additional external signals they are called *additive faults*. When the faults are modeled as parameter deviations they are called *multiplicative faults* because the system parameter depending on the fault size are multiplied with the malfunction. At the beginning, disturbances and model uncertainties have similar effects on the system. In contrast, faults are unknown elements which must be detected and removed. Disturbances and model uncertainties are known and they are handled by appropriate filtering or robust control design [20].

The faults can be classified as plant faults which change the dynamical properties of the system, sensor faults where the measuring devices of the system have errors, and actuator faults which influence the plant controller by interruptions or modifications (see Fig. 2.11).

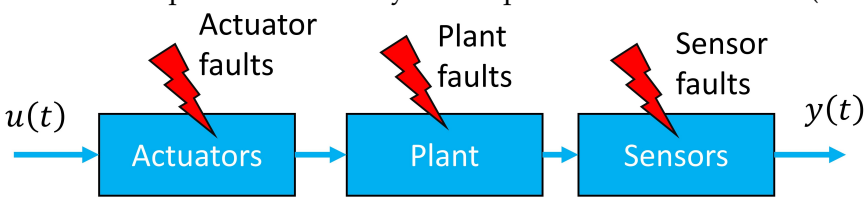


Fig. 2.11: Distinction between actuator faults, plant faults, and sensor faults.

Due to the higher complexity and the increasing number of components, multi-agent systems are particularly sensible to faults, which can happen with a higher probability and result in performance degradation or breakdown of all the agents [21]. Fault-tolerant control of multi-agent systems has attracted the researchers' interest in recent years. A synchronization resilience with unknown faults on communication links is addressed in [21]; the unknown faults are considered deterministic bounded functions. A back-stepping fault-tolerant sliding-mode control is designed for second-order nonlinear multi-agent systems subject to actuator faults and time-varying weighted topologies in [121]. An adaptive $\mathscr{H}_\infty$

control is proposed for linear time-invariant multi-agent systems subject to actuator faults, saturation, and $\mathscr{L}_2$ disturbance in [122]. A mitigation approach of cyber-physical attacks on sensors and actuators of discrete-time multi-agent systems is proposed in [123]. An output feedback fault-tolerant control scheme for heterogeneous multi-agent systems subject to loss of effectiveness, time-varying additive actuator faults, and external disturbance is investigated in [35]. A nonlinear and discontinuous adaptive observer-based event-triggered fault-tolerant control of multi-agent systems is investigated in [124]. An adaptive fault-tolerant control for multi-agent systems with deception attacks and actuator faults is studied in [125] where false data is injected in the communication channels as deception attacks. In [126], an adaptive consensus tracking for a class of high-order nonlinear time-varying multi-agent systems with output constraints, actuator faults, and external disturbance is studied. In [127], a distributed fault-tolerant tracking control based on adaptive neural networks is designed for a multi-agent systems under uncertain dynamics, unknown dead zones, and actuator failures. In [112], the synchronization of multiple memristive neural networks under cyber-physical attacks through distributed event-triggered control is investigated; Lyapunov analysis and M-matrix properties are used to get the criteria for synchronization under cyber-attacks. Several fault accommodation control schemes for a class of multi-agent systems are presented; a passive and active fault accommodation control schemes are presented based on an actuator fault detection in [128]. A resilient state feedback based leader-following tracking strategy is proposed for multi-agent systems under faults on sensors and actuators using an adaptive compensation and an $\mathcal{H}_\infty$ control in [129]. An adaptive fault-tolerant consensus control based on neural networks for nonlinear multi-agent systems with actuator faults is investigated in [130]. A distributed resilient control method for the consensus problem under denial-of-service attacks and communication delays is proposed in [131]. A fuzzy fault-tolerant control of a class of nonlinear multi-agent systems with actuator/sensor faults under directed and switching topology is addressed in [132]. A compensation based on fuzzy logic approach for actuator/sensor faults and unknown nonlinearities in a class of discrete time-delay systems under cyber-attacks is developed in [133]. Secure control problem of unknown cyber-physical systems with sparse actuator denial-of-service and deception attacks is investigated in [134]. A active fault-tolerant control in a second-order multi-agent systems is developed under partial loss of effectiveness actuator faults in [135]. A robust observer-based consensus reliable control for linear parameter-varying multi-agent systems against actuator faults is addressed using $\mathscr{H}_\infty$ in [136]. An adaptive observer-based fault-tolerant distributed containment control of multi-agent systems with actuator bias faults is studied in [137]. A fault-tolerant time-varying formation control problems for second-order multi-agent systems with directed topologies in [9], [138]; both bias and loss of effectiveness fault modes are considered. A robust adaptive fault tolerant protocol is presented to solve the flocking approach with uncertainties and actuator faults in [139]. A discrete fault-tolerant control based on $\mathscr{H}_\infty$ is presented with simultaneous faults in [140]. An adaptive fault tolerant cooperative control is presented in [141] considering unknown inputs. An adaptive fault-tolerant control problem for a class of multi-agent systems with matched unknown nonlinear functions and actuator bias faults is addressed in [142]. An adaptive fault-tolerant control based on neural network for nonlinear multi-agent systems with actuator faults is presented in [143]. A fully distributed adaptive fault-tolerant consensus protocol is proposed for leader-follower multi-agent systems with actuator loss of effectiveness and addictive faults in [144]. A fault-tolerant control in multi-agent systems subject to actuator faults is addressed based on an online estimator recursive least square and model predictive control in [145]. A fault-tolerant cooperative control problem of agent groups in the context of multi-agent flocking tasks with second order

linear dynamics is studied in [146]. An adaptive control based on fuzzy logic is presented in [147] using sliding mode observers with actuator faults under disturbances and uncertainties in the model of high-order multi-agent systems. A fault-tolerant leader-following control is presented in [148] with variant faults in the actuators. An adaptive fault-tolerant consensus in multi-agent system with an unknown function considered as actuator fault is investigated in [149]. A consensus tracking problem in multi-agent systems with both outage and partial loss of effectiveness types of actuator faults is investigated in [150]. A fault-tolerant control for a class of nonlinear multi-agent systems with variable communication topology is presented in [151].

Table 2.2 presents a classification of the papers previously described. The type of faults, cyber-attacks, the control strategy used, and the multi-agent system dynamics are analyzed. It is important to underline that the great majority of those research works study actuator fault problems using adaptive control strategies in linear-time invariant (LTI) systems.

Table 2.2: Classification of the state-of-the-art in fault-tolerant control for multi-agent systems.

Reference	Actuator	Sensor faults	Communication faults	Cyber-attacks	Control strategy	Dynamic
[21]			×		Adaptive	LTI
[121]	×				Back stepping sliding-mode	Second-order stochastic nonlinear
[122]	×				Adaptive/$\mathscr{H}_\infty$	LTI
[123]				Sensors and actuators	$\mathscr{H}_\infty$	Discrete-time
[35]	×				Adaptive/Virtual actuators	LTI heterogeneous
[124]	×				Adaptive event-triggered	LTI
[125]	×			Deception attacks	Adaptive	LTI
[126]	×				Adaptive	Nonlinear time-varying
[127]	×				Adaptive neural networks	Second-order
[112]				Cyber-attacks	M-matrix	Multiple memristive neural networks
[128]	×				Accommodation	LTI
[129]	×	×			Adaptive/$\mathscr{H}_\infty$	LTI
[130]	×				Adaptive neural networks	Nonlinear
[131]	×			Denial of service attacks	Distributed controller	Switching LTI
[132]	×	×			Fuzzy logic	Nonlinear
[133]	×	×		Cyber-attacks	Fuzzy logic	Discrete-time
[134]				Actuator attacks	LQ	LTI
[135]	×				Active FTC	Second-order
[136]	×				$\mathscr{H}_\infty$	LPV
[137]	×				Adaptive	LTI
[9], [138]	×				Formation control	LTI
[139]	×				Adaptive	Second-order
[140]	×	×			$\mathscr{H}_\infty$	LTI
[142]	×				Adaptive	Class of nonlinear
[143]	×				Adaptive neural networks	Nonlinear
[144]	×				Adaptive	LTI
[145]	×				MPC	LTI
[146]	×				Flocking	Second-order
[147]	×				Adaptive fuzzy logic	High-order
[149]	×				Adaptive	LTI
[150]	×				Virtual actuators	LTI

The following section yields an analysis and conclusion of the main areas of opportunity

for research work considering faults in multi-agent systems.

2.4 Conclusions

In the last decades, the most studied problems in the context of multi-agent systems are consensus and leader-following consensus [61], [86]. The difference between consensus and leader-following consensus can be compared as regulation and tracking. Regulation (consensus) consists to stabilize a system in an equilibrium point. Tracking (leader-following consensus) lies to follow a desired trajectory. As previously mentioned, the leader-following consensus allows to synchronize a team of agents through a virtual/physical leader agent. For this reason, leader-following consensus is taken in order to manipulate the multi-agent system through a leader.

A critical problem for cooperative control is to determine algorithms so that the multi-agent system can reach the consensus in the presence of imperfect sensors, communication dropout, spare communication topologies, and noisy communication links [33]. The consensus performance can be affected due to bandwidth limitations [152], packet losses [153], external disturbances [45], unknown inputs [154], sensor [129], actuator [121], or communication faults [21], as well as cyber-attacks [133]. As previously mentioned, several control approaches have been addressed those problems using event-triggered [155], adaptive [32], $\mathscr{H}_\infty$ optimization [70], fault-tolerant control [121], or resilient control [123], however, new methods could be developed as mentioned in the following paragraphs.

In fault-tolerant control systems, one of the important issues is to maintain the system performance close to the desirable one and to preserve stability conditions after occurrence of a fault [120]. As seen in Table 2.2, most of the researcher works have considered actuator faults using adaptive control strategies [35], [124]–[127]. Nevertheless, virtual actuators have been proposed as a fault accommodation approach. The faulty system is modified adding a virtual actuator block which allows to reconfigure the controller [20]. The main advantage of using virtual actuators is that the nominal control could be used without re-tuning it [156]. Loss of effectiveness, time-varying additive actuator faults, and external disturbance are investigated in [35] for heterogeneous multi-agent systems using adaptive virtual actuators. In [157], recursive least-square parameter estimation is used to estimate multiplicative actuator faults, in contrast with [150] where an adaptive observer is used to estimate the faults; both papers consider a nominal consensus control and then based on the fault estimation the virtual actuator gain is calculated independently for each agent using the pole placement approach. Neither, event-triggered techniques and virtual actuators or considering communication faults have been considered. The design of distributed virtual actuators in multi-agent systems subject to actuator faults is provided in Chapter 3.

Event-triggered approaches have been used in the literature for reducing the exchange of the information and the control update rate in linear agents [27], nonlinear agents [105], a multi-layer Kuramoto-oscillator network [106], general linear fractional order [114], single and double integrators [26], or a group of vertical take-off and landing unmanned aerial vehicles (VTOL-UAVs) [28]. The importance of event-triggered control lies in providing an effective means to avoid continuous communication when the limited resources are considered (embedded processors or restricted bandwidth) [107], [108]. Event-based strategies

can be found in second-order multi-agent systems subject to external disturbances [89], under actuator faults [124], cyber-attacks [112], or formation control [28]; however, communication faults has not been considered yet in event-triggered techniques. An event-triggered strategy in multi-agent systems is studied and applied in a fleet of UAVs under communication faults in Chapter 4.

Quadratic boundedness is an approach found in the control theory of linear time-invariant systems to solve the exogenous disturbances problem [97]. Quadratic boundedness approaches have been used in fault estimation [98], optimal estimation of unmanned aerial vehicles (UAVs) [99], or feedback stabilization of Takagi-Sugeno systems [100] to compensate bounded external disturbances. However, the quadratic boundedness consensus has not been found in the literature in order to solve such problem in multi-agent systems subject to disturbance/actuator faults. For that reason, a passive fault-tolerant control based on quadratic boundedness consensus considering bounded faults is designed in Chapter 5.

Chapter 3

Event-Triggered and Fault-Tolerant Leader-Following Control for Multi-agent Systems

3.1 Introduction

Proper resource management is a challenge to be considered when implementing control systems in real applications. The vast majority of autonomous systems have independent battery-based power sources, which in some cases are insufficient to carry out long operations over time. These autonomous systems have embedded microprocessors that are used to manage components such as sensors and actuators. As a result of the rapid development of computer technologies, discrete-time systems are becoming increasingly important. One crucial issue is how to select an appropriate sample time. The microprocessor samples every discrete-time the sensors and compute the control law in order to manipulate the system. The control which is periodically sampled each time is called time-triggered control. In time-triggered architectures, activities are triggered by the progression of global time [158]. Event-based control is a control methodology often used for reducing the communication load of a digital network. The sampling instants are determined by an event generator [101].

There are several reasons for using event-based feedback. First, the information exchange is reduced to ensure a required system performance. The physical structure requires that measurements or control actions have to be taken at time instants prescribed by the dynamics of the plant. Asynchronous communication protocols and real-time software do not allow to transfer and process information at specific clock times [101]. As mentioned in subsection 2.3.3, the event generator determines the time instants t_k, $k \in \mathbb{Z}^+$, at which the next communication between the event generator, the controller, and the control input generator is applied, and the information that is communicated from the sensor to the controller. The control input generator establishes the signal $u(t)$ for the time interval $t \in [t_k, t_{k+1})$ in dependence on the information obtained at time t_k.

Event-triggered approaches have been employed in multi-agent systems in order to reduce the broadcasting in the communication and the update control rate [26]–[28]. Consequently, the energy consumption in each agent is reduced, and the bandwidth limitation is well managed.

Just as proper resource management is a challenge in control systems, fault tolerance is an even greater challenge that must be considered. As presented in subsection 2.3.4, modern control systems are becoming more complex and sophisticated, in consequence, the issues of availability, cost efficiency, reliability, operating safety, and environmental protection are major importance [20]. In fault-tolerant control systems, one of the important issues is to maintain the system performance close to the desirable one and to preserve stability conditions after occurrence of a fault [120]. Due to the higher complexity and the increasing number of components, multi-agent systems are particularly sensible to faults, which can happen with a higher probability and result in performance degradation or breakdown of all the agents [21].

Virtual actuators have been proposed as a fault accommodation approach. The faulty system is modified adding a virtual actuator block which allows to reconfigure the controller [20]. The main advantage of using virtual actuators is that the nominal control could be used without re-tuning it [156]. Few works can be found in the literature concerning virtual actuators to reconfigure the cooperative control law in multi-agent systems. Loss of effectiveness, time-varying additive actuator faults, and external disturbance are investigated in [35] for heterogeneous multi-agent systems using adaptive virtual actuators. In [157], recursive least-square parameter estimation is used to estimate multiplicative actuator faults, in contrast with [150] where an adaptive observer is used to estimate the faults; both papers consider a nominal consensus control and then, based on the fault estimation, the virtual actuator gain is calculated independently for each agent using the pole placement approach.

This chapter presents the design of an event-triggered and a fault-tolerant leader-following control for multi-agent systems. The design of both strategies are divided in two sections. The first section is dedicated to design an leader-following control. Then main contribution consists in the design of an observer-based leader-following consensus which guarantee that all the agents follow the trajectories of a leader agent and then, based on the eigenvalues of the closed-loop multi-agent system, an event-triggered mechanism is added with a dynamic-threshold reducing the information exchange between agents and the update rate of the control law.

Fig. 3.1 shows the proposed strategy scheme where each agent has an observer, a consensus protocol, and an event-triggered mechanism.

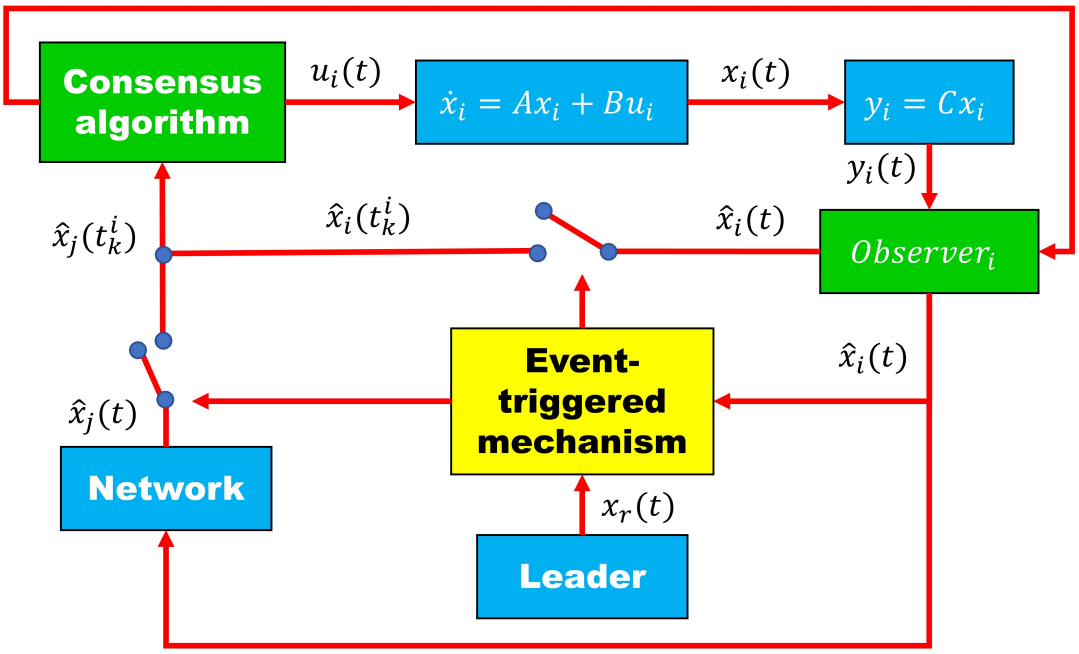


Fig. 3.1: Event-triggered control scheme in multi-agent systems.

The second contribution consists in the design of an observer-based fault-tolerant control in multi-agent systems under actuator faults. The proposed strategy reconfigures the observer-based leader-following nominal control preserving an acceptable consensus performance using distributed virtual actuators for each agent. Sufficient LMI-based conditions have been obtained to compute the controller, observer, and virtual actuator gains in order to guarantee the stability of the consensus, the estimated error, and the virtual actuators. Fig. 3.2 illustrates the scheme of the proposed strategy for each agent.

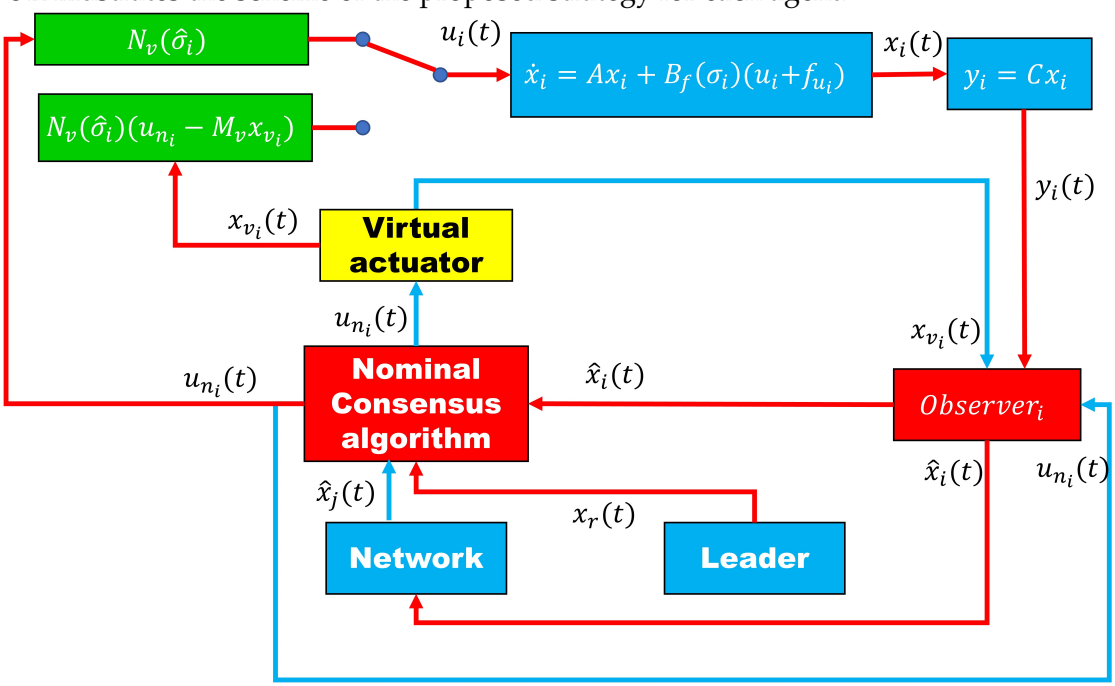


Fig. 3.2: Virtual actuators scheme in multi-agent systems.

Linear matrix inequalities (LMIs)-based sufficient conditions are obtained to compute the controller, observer, and virtual actuator gains which guarantee stability of the synchronization error, the estimation error, and the virtual actuator. The effectiveness of both strategies are illustrated through numerical examples. The focus in the following section is to design a leader-following control based on event-triggered approach using the estimated state vector of the local and neighboring agents such that all agents follow the leader's trajectories in order to reduce the update control rate and the exchange of information.

This chapter is organized as follows. The observer-based leader-following control is designed in subsection 3.2.1. The event-triggered mechanism is provided in subsection 3.2.2. A comparison of the proposed approach and the classical formation control is illustrated in Section 3.2.3. The fault-tolerant leader-following control is designed in subsection 3.3.1. The synthesis of the controller, observer, and virtual actuator gains are given in subsection 3.3.2. Numerical examples are used to illustrate the effectiveness of the proposed strategy in subsection 3.3.3. Finally, the conclusions of this chapter are drawn in Section 3.4.

3.2 Observer-based event-triggered leader-following control for multi-agent systems

Consider the following system:

$$\begin{aligned} \dot{x}_i(t) &= Ax_i(t) + Bu_i(t), \\ y_i(t) &= Cx_i(t), \end{aligned} \tag{3.1}$$

where $x_i(t) \in \mathbb{R}^n$ is the vector state, $u_i(t) \in \mathbb{R}^m$ is the input vector, $y_i(t) \in \mathbb{R}^p$ is the measurement output vector, i represents the *ith* agent ($\forall i = 1, 2, \ldots, N$) with matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$. The following assumptions hold in this section.

Assumption 1. *The pair (A, B) is stabilizable (see Appendix 4).*

Assumption 2. *The pair (A, C) is detectable (see Appendix 4).*

Assumption 3. *The graph $\mathcal{G}$ is undirected.*

Assumption 4. *All the agents have information about the virtual leader.*

The objective is to design a leader-following control for the multi-agent system (3.1) such that all the agents follow the leader's reference trajectories in a consensus. The dynamics of the leader are considered as follows:

$$\dot{x}_r(t) = Ax_r(t) + Bu_r(t), \tag{3.2}$$

where $x_r(t) \in \mathbb{R}^n$ is the leader's state vector and $u_r(t) \in \mathbb{R}^m$ is the leader's input vector. The trajectories to be followed are given by the leader-following control. Let us define the synchronization error between the leader and each agent i as follows:

$$\delta_i(t) = x_i(t) - x_r(t), \tag{3.3}$$

then, the synchronization error dynamics are obtained:

$$\begin{aligned} \dot{\delta}_i(t) &= \dot{x}_i(t) - \dot{x}_r(t), \\ &= Ax_i(t) + Bu_i(t) - Ax_r(t) - Bu_r(t), \\ &= A\delta_i(t) + B\Delta u_i(t) \end{aligned} \tag{3.4}$$

with $\Delta u_i(t) := u_i(t) - u_r(t)$.

Definition 1 ([86]). *The leader-following control problem is achieved if the multi-agent system (3.1) satisfies:*

$$\lim_{t\to\infty} \|x_i(t) - x_r(t)\| = 0, \ \forall i = 1, 2, \ldots, N. \tag{3.5}$$

for any initial condition.

According to [86], the classical observer-based leader-following control is given by the following equation:

$$\Delta u_i(t) = K\left[\sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{x}_i(t) - \hat{x}_j(t)\right) + \alpha\left(\hat{x}_i(t) - x_r(t)\right)\right] \tag{3.6}$$

where $K \in \mathbb{R}^{m\times n}$ is the control gain to be designed, $\alpha > 0$ is a positive constant, $\hat{x}_i(t)$ and $\hat{x}_j(t)$ are the estimated state vector of the agent i and j, respectively.

3.2.1 Observer-based leader-following control design

The full state vector is not always available, thus, the leader-following control uses the estimation provided by the following observer:

$$\begin{aligned} \dot{\hat{x}}_i(t) &= A\hat{x}_i(t) + Bu_i(t) + L_o\left(y_i(t) - \hat{y}_i(t)\right), \\ \hat{y}_i(t) &= C\hat{x}_i(t), \end{aligned} \tag{3.7}$$

where $\hat{x}_i(t) \in \mathbb{R}^n$ is the estimated state vector, $\hat{y}_i(t) \in \mathbb{R}^p$ is the estimated output vector, and the matrix $L_o \in \mathbb{R}^{n\times p}$ is the observer gain to be designed. The estimation error of each agent is defined as follows:

$$e_i(t) = x_i(t) - \hat{x}_i(t). \tag{3.8}$$

Thus, the dynamic estimation error is defined as follows:

$$\dot{e}_i(t) = \dot{x}_i(t) - \dot{\hat{x}}_i(t) = (A - L_oC)e_i(t). \tag{3.9}$$

Let $e(t) = \left[e_1(t)^T, e_2(t)^T, \ldots, e_N(t)^T\right]^T$, then, (3.9) can be rewritten in the compact form:

$$\dot{e}(t) = \left(I_N \otimes (A - L_oC)\right)e(t). \tag{3.10}$$

Note that:

$$\begin{aligned} &BK\sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{x}_i(t) - \hat{x}_j(t)\right) + \alpha BK\left(\hat{x}_i(t) - x_r(t)\right) \\ &= BK\sum_{j\in\mathcal{N}_i} a_{ij}\left(\left(\delta_i - \delta_j\right) - \left(e_i - e_j\right)\right) + \alpha BK\left(\delta_i - e_i\right). \end{aligned} \tag{3.11}$$

Based on (3.11), (3.4) becomes:

$$\dot{\delta}_i(t) = A\delta_i(t) + BK\sum_{j\in\mathcal{N}_i} a_{ij}\left(\left(\delta_i - \delta_j\right) - \left(e_i - e_j\right)\right) + \alpha BK\left(\delta_i - e_i\right). \tag{3.12}$$

Let $\delta(t) = \left[\delta_1(t)^T, \delta_2(t)^T, \ldots, \delta_N(t)^T\right]^T$. Then, the synchronization error model is written using the Kronecker product as follows:

$$\dot{\delta}(t) = \left(I_N \otimes A\right)\delta(t) + \left(\left(\alpha I_N + \mathcal{L}\right) \otimes BK\right)\left(\delta(t) - e(t)\right). \tag{3.13}$$

Let $z(t) = \left[\delta(t)^T, e(t)^T\right]^T$, combining (3.10) and (3.13), it is obtained

$$\dot{z}(t) = \tilde{A}z(t), \tag{3.14}$$

where

$$\tilde{A} = \begin{bmatrix} I_N \otimes A + (\alpha I_N + \mathcal{L}) \otimes BK & -(\alpha I_N + \mathcal{L}) \otimes BK \\ 0_{nN\times nN} & I_N \otimes (A - L_o C) \end{bmatrix}. \tag{3.15}$$

In order to analyze the stability of the synchronization error and the estimation error, let us choice a candidate Lyapunov function for the closed-loop system in (3.14) as follows $V_1(z) = z^T diag\left(I_N \otimes P_1, I_N \otimes P_2\right) z$ where $P_1 = P_1^T > 0$ and $P_2 = P_2^T > 0$. The derivative of $V_1(z)$ along the trajectories of (3.14) is given by:

$$\begin{aligned} \dot{V}_1 =& 2z^T diag\left(I_N \otimes P_1, I_N \otimes P_2\right) \dot{z}, \\ =& 2\delta^T \left(I_N \otimes P_1\right) \dot{\delta} + 2e^T \left(I_N \otimes P_2\right) \dot{e}, \\ =& 2\delta^T \left[\left(I_N \otimes P_1 A\right) \delta + \left(\left(\alpha I_N + \mathcal{L}\right) \otimes P_1 BK\right) \delta - \left(\left(\alpha I_N + \mathcal{L}\right) \otimes P_1 BK\right) e \right] \\ & + 2e^T \left[I_N \otimes P_2 \left(A - L_o C\right)\right] e. \end{aligned} \tag{3.16}$$

Let us perform a spectral decomposition of the Laplacian matrix $\mathcal{L}$, such that $\mathcal{L} = TJT^T$ with a matrix $T \in \mathbb{R}^{N\times N}$ and a diagonal matrix $J = diag\left(\lambda_1(\mathcal{L}), \lambda_2(\mathcal{L}), \ldots, \lambda_N(\mathcal{L})\right) \in \mathbb{R}^{N\times N}$ with $\lambda_1(\mathcal{L}) = 0$ where $\lambda_i(\mathcal{L})$ represents *ith*-eigenvalues of $\mathcal{L}$. By Lemma 2, the eigenvalues of $\mathcal{L}$ form a base of eigenvectors which are used to construct the invertible matrix T. Let us define a change of coordinates as follows:

$$\begin{aligned} \phi &= \left(T^T \otimes I_N\right) \delta, \\ \theta &= \left(T^T \otimes I_N\right) e. \end{aligned} \tag{3.17}$$

Replacing (3.17) in (3.16) leads to:

$$\begin{aligned} \dot{V}_1 =& 2\phi^T \left[\left(I_N \otimes P_1 A\right) + \left(\left(\alpha I_N + J\right) \otimes P_1 BK\right)\right] \phi - 2\phi^T \left(\left(\alpha I_N + J\right) \otimes P_1 BK\right) \theta \\ & + 2\theta^T \left(I_N \otimes P_2 \left(A - L_o C\right)\right) \theta. \end{aligned} \tag{3.18}$$

Because J has the eigenvalues of $\mathcal{L}$ on its diagonal, (3.18) can be rewritten as follows:

$$\dot{V}_1 = \sum_{i=1}^{N} \left[\phi_i^T \mathrm{He}\left\{P_1 \left(A + \left(\alpha + \lambda_i\right) BK\right)\right\} \phi_i - 2\phi_i^T \left(\alpha + \lambda_i\right) P_1 BK\theta_i + \theta_i^T \mathrm{He}\left\{P_2 \left(A - L_o C\right)\right\} \theta_i\right]. \tag{3.19}$$

Let us define $\varphi_i = \left[\phi_i^T, \theta_i^T\right]^T$ so that (3.19) is rewritten as:

$$\begin{aligned} \dot{V}_1 &= \sum_{i=1}^{N} \varphi_i^T \Theta_{1_i} \varphi_i, \\ \Theta_{1_i} &= \begin{bmatrix} \mathrm{He}\left\{P_1 \left(A + \left(\alpha + \lambda_i\right) BK\right)\right\} & -\left(\alpha + \lambda_i\right) P_1 BK \\ * & \mathrm{He}\left\{P_2 \left(A - L_o C\right)\right\} \end{bmatrix}. \end{aligned} \tag{3.20}$$

If $\Theta_{1_i} < 0$, $\forall i = 1, 2, \ldots, N$, then $\dot{V}_1 < 0$. Thus, the synchronization and the estimation error are stable. However, the matrices cannot be computed using conventional tools due to the products of decision variables, such as $\left(\alpha + \lambda_i\right) P_1 BK$. The following theorem provides LMIs-based conditions in order to compute the controller and observer gains.

Theorem 1. *Considering the closed-loop system in (3.14), given eigenvalues $\lambda_i(\mathcal{L})$ of the Laplacian matrix $(i = 1, 2, \ldots, N)$, the scalar $\alpha > 0$, and scalars $\mu_1 > 0$ and $\mu_2 > 0$, if there exist symmetric matrices $P_1 > 0$, $P_2 > 0$, and matrices K, M such that the following inequality*

$$\begin{bmatrix} \mathrm{He}\{P_1 A\} + I_n - \frac{2P_1}{\mu_1} & 0_{n\times n} & (\alpha+\lambda_i) P_1 B & \frac{P_1}{\mu_1} + (\alpha+\lambda_i)\mu_1 BK \\ * & \mathrm{He}\{P_2 A - MC\} & -\mu_2 K^T & 0_{n\times n} \\ * & * & -2\mu_2 I_m & 0_{m\times n} \\ * & * & * & -I_n \end{bmatrix} < 0 \tag{3.21}$$

holds, then the synchronization and the estimation error are stable, thus, the leader-following control problem for the multi-agent system in (3.1) is solved under the control defined as $u_i(t) = \Delta u_i(t) + u_r(t)$ and observer gain $L_o = P_2^{-1} M$.

Proof. Using Schur complement in (3.21), it is obtained:

$$\begin{bmatrix} Q_1 & 0_{n\times n} & (\alpha+\lambda_i) P_1 B \\ * & \mathrm{He}\{P_2 A - MC\} & -\mu_2 K^T \\ * & * & -2\mu_2 I_m \end{bmatrix} < 0, \tag{3.22}$$

where $Q_1 = \mathrm{He}\{P_1 A\} + I_n - \frac{2P_1}{\mu_1} + \left(\frac{P_1}{\mu_1} + (\alpha+\lambda_i)\mu_1 BK\right)^T \left(\frac{P_1}{\mu_1} + (\alpha+\lambda_i)\mu_1 BK\right)$. Note that,

$$\begin{aligned} &\mathrm{He}\{P_1 (A + (\alpha+\lambda_i) BK)\} \\ &\leq \mathrm{He}\{P_1 (A + (\alpha+\lambda_i) BK)\} + \mu_1^2 (\alpha+\lambda_i)^2 (BK)^T (BK) \\ &= \mathrm{He}\{P_1 A\} - \frac{P_1^2}{\mu_1^2} \\ &+ \left(\frac{P_1}{\mu_1} + (\alpha+\lambda_i)\mu_1 BK\right)^T \left(\frac{P_1}{\mu_1} + (\alpha+\lambda_i)\mu_1 BK\right), \end{aligned} \tag{3.23}$$

where $\mu_1 > 0$ guarantees $(\alpha+\lambda_i) BK$ would not be too big. Then, the following inequality is introduced

$$\begin{aligned} \left(I_n - \frac{P_1}{\mu_1}\right)\left(I_n - \frac{P_1}{\mu_1}\right) &\geq 0 \\ I_n - \frac{2P_1}{\mu_1} &\geq -\frac{P_1^2}{\mu_1^2}. \end{aligned} \tag{3.24}$$

Combining (3.23) and (3.24), it is obtained

$$\begin{aligned} &\mathrm{He}\{P_1 (A + (\alpha+\lambda_i) BK)\} \\ \leq &\mathrm{He}\{P_1 A\} + I_n - \frac{2P_1}{\mu_1} \\ &+ \left(\frac{P_1}{\mu_1} + (\alpha+\lambda_i)\mu_1 BK\right)^T \left(\frac{P_1}{\mu_1} + (\alpha+\lambda_i)\mu_1 BK\right). \end{aligned} \tag{3.25}$$

Multiplying (3.22) the left and the right sides by

$$\begin{bmatrix} I_n & 0_{n\times m} & 0_{n\times m} \\ 0_{n\times n} & I_m & -K^T \end{bmatrix}$$

and its transpose, it is obtained:

$$\begin{bmatrix} Q_1 & -(\alpha+\lambda_i)\,P_1BK \\ * & \mathrm{He}\left\{P_2A-MC\right\} \end{bmatrix} < 0. \tag{3.26}$$

Combining (3.25), (3.26), and selecting $M = P_2L_o$, it recovers:

$$\begin{bmatrix} \mathrm{He}\left\{P_1\left(A+(\alpha+\lambda_i)\,BK\right)\right\} & -(\alpha+\lambda_i)\,P_1BK \\ * & \mathrm{He}\left\{P_2\left(A-L_oC\right)\right\} \end{bmatrix} < 0. \tag{3.27}$$

Condition (3.27) is equal to (3.20), thus, the condition (3.21) satisfies the condition (3.20) which completes the proof. □

In the following subsection, the leader-following control is extended using an event-triggered mechanism in order to reduce the exchange of information and the control update rate.

3.2.2 Event-triggered control mechanism

In an event-triggered approach, the update of the control action depends on the synchronization error. When the magnitude of this error exceeds a threshold, then the value of the control law is updated, otherwise, the control law keeps the last value. In order to design an event-triggered leader-following control, (3.6) is modified as follows:

$$\Delta u_i(t) = K\left[\sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{x}_i(t_k^i)-\hat{x}_j(t_k^i)\right)+\alpha\left(\hat{x}_i(t_k^i)-x_r(t_k^i)\right)\right] \tag{3.28}$$

where $\hat{x}_i(t_k^i)$ and $\hat{x}_j(t_k^i)$ are the last value of the observer state vector from the agent i and j respectively, $x_r(t_k^i)$ is the last value of the leader's state vector, and t_k^i is the last event-time of the agent i. The sequence of event-times $0 \le t_0^i \le t_1^i \le \ldots$ for agent i is defined as $t_{k+1}^i = inf\{t : t > t_k^i, g_i\left(\xi_i(t)\right) > 0\}$ where $g\left(\xi_i(t)\right)$ is an appropriately chosen event function (see Theorem 2 in the next page). It should be clarified, agent i requests the information of the estimated state of j at the event t_{k+1}^i in order to update the control law, otherwise, the control law keeps the last value computed. That is, the control law and the neighboring estimated states are updated at event t_{k+1}^i. Note that from equation (3.12), the following is obtained:

$$\dot{\delta}_i(t) = A\delta_i(t)+BK\sum_{j\in\mathcal{N}_i} a_{ij}\left[\left(\delta_i(t_k^i)-\delta_j(t_k^i)\right)-\left(e_i(t_k^i)-e_j(t_k^i)\right)\right]+\alpha BK\left(\delta_i(t_k^i)-e_i(t_k^i)\right), \tag{3.29}$$

where $\delta_i(t_k^i)$ and $\delta_j(t_k^i)$ are the last value of the synchronization error of agent i and j, $e_i(t_k^i)$ and $e_j(t_k^i)$ are the last value of the observer error of agent i and j. Let us define the event error, which determines by a threshold when trigger an event, as follows:

$$\begin{aligned} \xi_i(t) =& \delta_i(t_k^i)-\delta_i(t), \\ \zeta_i(t) =& e_i(t_k^i)-e_i(t). \end{aligned} \tag{3.30}$$

Then, using (3.30) in (3.29), it is obtained:

$$\begin{aligned} \dot{\delta}_i(t) = A\delta_i(t)+BK\sum_{j\in\mathcal{N}_i} a_{ij}\left(\delta_i(t)+\xi_i(t)-\delta_j(t)-\xi_j(t)\right)-BK\sum_{j\in\mathcal{N}_i} a_{ij}\left(\zeta_i(t)+e_i(t)-\zeta_j(t)-e_j(t)\right) \\ +\alpha BK\left(\delta_i(t)+\xi_i(t)-\zeta_i(t)-e_i(t)\right). \end{aligned} \tag{3.31}$$

Let $\xi(t) = \left[\xi_1(t)^T, \xi_2(t)^T, \ldots, \xi_N(t)^T\right]^T$ and $\zeta(t) = \left[\zeta_1(t)^T, \zeta_2(t)^T, \ldots, \zeta_N(t)^T\right]^T$, (3.31) can be rewritten as follows:

$$\dot{\delta}(t) = (I_N \otimes A)\,\delta(t) + ((\alpha I_N + \mathcal{L}) \otimes BK)\,(\delta(t) - e(t) + \xi(t) - \zeta(t))\,. \tag{3.32}$$

Let $\epsilon(t) = \left[\xi(t)^T, \zeta(t)^T\right]^T$. Similarly, (3.14) is rewritten as:

$$\dot{z}(t) = \tilde{A}z(t) + \tilde{B}\epsilon(t), \tag{3.33}$$

where $\tilde{A}$ is defined in (3.15), and

$$\tilde{B} = \begin{bmatrix} (\alpha I_N + \mathcal{L}) \otimes BK & -(\alpha I_N + \mathcal{L}) \otimes BK \\ 0 & 0 \end{bmatrix}. \tag{3.34}$$

The controller gain K and observer gain L_o are calculated by Theorem 1, thus, the closed-loop system in (3.33) is Hurwitz. The following Lemma is necessary for Theorem 2 in order to demonstrate that the closed-loop system (3.33) is stable in a bounded region.

Lemma 1 ([159]). *Let $\Gamma \in \mathbb{R}^{2nN \times 2nN}$ be an invertible matrix such that $\tilde{A} = \Gamma \mathcal{J} \Gamma^{-1}$ where $\mathcal{J} = diag\left(\gamma_1(\tilde{A}), \gamma_2(\tilde{A}), \ldots, \gamma_{2nN}(\tilde{A})\right)$ where $\gamma_1(\tilde{A}), \gamma_2(\tilde{A}), \ldots, \gamma_{2nN}(\tilde{A})$ are the eigenvalues of $\tilde{A}$, then it follows that:*

$$\left\|\mathbf{e}^{\tilde{A}t}\right\| \leq \|\Gamma\| \left\|\Gamma^{-1}\right\| \mathbf{e}^{\gamma(\tilde{A})t} = \kappa(\Gamma)\mathbf{e}^{\gamma(\tilde{A})t} \tag{3.35}$$

with $\kappa(\Gamma) = \|\Gamma\| \left\|\Gamma^{-1}\right\|$ and $\gamma(\tilde{A})$ is the maximum real eigenvalue of the matrix $\tilde{A}$.

Then, lemma 1 is used in the following Theorem which demonstrate that the closed-loop system (3.33) is stable in a bounded region.

Theorem 2. *Consider the multi-agent system in (3.1) with the control law in (3.28). The event function is given by*

$$g_i(\xi_i(t)) = \|\xi_i(t)\| - \left(c_1 + c_2\mathbf{e}^{\beta t}\right) \tag{3.36}$$

with constants $c_1 > 0$ and $c_2 > 0$, and $\gamma(\tilde{A}) < \beta < 0$ where $\gamma(\tilde{A})$ is the maximum real eigenvalue of the matrix $\tilde{A}$. Then, for all initial conditions, the closed-loop system (3.33) does not exhibit Zeno behavior, this is, the event-triggered control does not demand an infinite numbers of events in a finite time.

Proof. The analytical solution of (3.33) is given by

$$z(t) = \mathbf{e}^{\tilde{A}t}z(0) + \int_0^t \mathbf{e}^{\tilde{A}(t-s)}\tilde{B}\epsilon(s)ds. \tag{3.37}$$

Thus, the closed-loop is bounded by

$$\|z(t)\| \leq \left\|\mathbf{e}^{\tilde{A}t}z(0)\right\| + \int_0^t \left\|\mathbf{e}^{\tilde{A}(t-s)}\tilde{B}\epsilon(s)\right\| ds. \tag{3.38}$$

Lemma 1 yields

$$\|z(t)\| \leq \mathbf{e}^{\gamma(\tilde{A})t}\kappa(\Gamma)\,\|z(0)\| + \kappa(\Gamma)\int_0^t \mathbf{e}^{\gamma(\tilde{A})(t-s)}\left\|\tilde{B}\epsilon(s)\right\| ds. \tag{3.39}$$

Since $\left\|\tilde{B}\epsilon(t)\right\| \leq \left\|\tilde{B}\right\| \|\epsilon(t)\|$ and $\|\epsilon(t)\| \leq \sqrt{N}\left(c_1 + c_2\mathbf{e}^{\beta t}\right)$, it is obtained:

$$\begin{aligned}\|z(t)\| \leq & \mathbf{e}^{\gamma(\tilde{A})t}\kappa(\Gamma)\|z(0)\| + \kappa(\Gamma)\left\|\tilde{B}\right\|\sqrt{N}\int_0^t \mathbf{e}^{\gamma(\tilde{A})(t-s)}\left(c_1 + c_2\mathbf{e}^{\beta s}\right)ds \\ = & \kappa(\Gamma)\left[-\frac{c_1\left\|\tilde{B}\right\|\sqrt{N}}{\gamma(\tilde{A})} + \frac{c_2\left\|\tilde{B}\right\|\sqrt{N}}{\beta-\gamma(\tilde{A})}\mathbf{e}^{\beta t} + \left(\|z(0)\| + \left\|\tilde{B}\right\|\sqrt{N}\left(\frac{c_1}{\gamma(\tilde{A})} - \frac{c_2}{\beta-\gamma(\tilde{A})}\right)\right)\mathbf{e}^{\gamma(\tilde{A})t}\right].\end{aligned} \tag{3.40}$$

By Theorem 1, the closed-loop matrix $\tilde{A}$ is Hurwitz, thus, all the real eigenvalues of $\tilde{A}$ are negative, then, $\gamma(\tilde{A}) < \beta < 0$. Inequality (3.40) can be upper bounded by:

$$\|z(t)\| \leq -\frac{c_1\kappa(\Gamma)\left\|\tilde{B}\right\|\sqrt{N}}{\gamma(\tilde{A})} + \frac{c_2\kappa(\Gamma)\left\|\tilde{B}\right\|\sqrt{N}}{\beta-\gamma(\tilde{A})}\mathbf{e}^{\beta t} + \|z(0)\|\kappa(\Gamma)\mathbf{e}^{\gamma(\tilde{A})t}. \tag{3.41}$$

Let us define t^* as the last event occurrence, $\|\xi_i(t^*)\| = 0$, and $g_i(t^*) = -(c_1 + c_2\mathbf{e}^{\beta t}) < 0$. From (3.30), $\dot{\epsilon}_i(t) = -\dot{z}_i(t)$ between two consecutive events, thus,

$$\|\epsilon_i(t)\| \leq \int_{t^*}^t \|\dot{z}_i(s)\|\,ds. \tag{3.42}$$

From (3.33), it can be derived

$$\|\dot{z}(t)\| \leq \left\|\tilde{A}\right\|\|z(t)\| + \left\|\tilde{B}\right\|\|\epsilon(t)\|. \tag{3.43}$$

By definition $\|\dot{z}_i(t)\| \leq \|\dot{z}(t)\|$, if the last event occurs at time $t^* > 0$, then using (3.42) in (3.43), it is obtained:

$$\|\epsilon_i(t)\| \leq \int_{t^*}^t \left(\left\|\tilde{A}\right\|\|z(s)\| + \left\|\tilde{B}\right\|\|\epsilon(s)\|\right)ds \tag{3.44}$$

and $\|z(t)\| \leq \|z(t^*)\|$ holds in (3.40). Thus, the error can be bounded as follows:

$$\|\epsilon_i(t)\| \leq \int_{t^*}^t \left(\mathbf{K}_1 + \mathbf{K}_2\mathbf{e}^{\beta s} + \mathbf{K}_3\mathbf{e}^{\gamma(\tilde{A})s}\right)ds \tag{3.45}$$

where

$$\mathbf{K}_1 = c_1\left\|\tilde{B}\right\|\sqrt{N}\left(\frac{\kappa(\Gamma)\left\|\tilde{A}\right\|}{\gamma(\tilde{A})} + 1\right), \mathbf{K}_2 = c_2\left\|\tilde{B}\right\|\sqrt{N}\left(\frac{\kappa(\Gamma)\left\|\tilde{A}\right\|}{\beta-\gamma(\tilde{A})} + 1\right), \mathbf{K}_3 = \kappa(\Gamma)\left\|\tilde{A}\right\|\|z(0)\|.$$

Because $\mathbf{e}^{\gamma(\tilde{A})s} \leq \mathbf{e}^{\gamma(\tilde{A})t^*}$ and $\mathbf{e}^{\beta s} \leq \mathbf{e}^{\beta t^*}$, it holds that

$$\|\epsilon_i(t)\| \leq \int_{t^*}^t \left(\mathbf{K}_1 + \mathbf{K}_2\mathbf{e}^{\beta t^*} + \mathbf{K}_3\mathbf{e}^{\gamma(\tilde{A})t^*}\right)ds = \left(\mathbf{K}_1 + \mathbf{K}_2\mathbf{e}^{\beta t^*} + \mathbf{K}_3\mathbf{e}^{\gamma(\tilde{A})t^*}\right)(t - t^*) \tag{3.46}$$

The next event will not be triggered before (3.36) crosses zero. Thus, a positive lower bound on the inter-event times is given by $\frac{c_1}{\mathbf{K}_1+\mathbf{K}_2+\mathbf{K}_3}$. □

Remark 1. *Theorem 2 provides an event function such that the control in (3.28) is updated with the sequence of event-times $t^i_{k+1} = inf\{t : t > t^i_k, g_i(\xi_i(t)) > 0\}$ and it does not exhibit Zeno behavior. In order to illustrate the effectiveness of the event-triggered leader-following control, a comparison of two numerical examples is presented.*

In the following section, the fault-tolerant leader-following control is design for multi-agent systems under actuator faults.

3.2.3 Event-triggered leader-following consensus in second-order multi-agent systems

Double integrator models are used to describe the agent dynamics when accelerations rather than velocities are directly manipulated. According to [28], it is possible to represent a fleet of UVAs as a double integrator multi-agent system. Therefore, a double integrator multi-agent system is considered for the following simulations. The states of each agent are positions and velocities, as well as the accelerations are considered inputs in a three-dimensional Euclidean space. Only positions are assumed as measurable states. The dynamics of each agent (3.1) are described by the following matrices:

$$A = \begin{bmatrix} 0_{3\times3} & I_3 \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}, \; B = \begin{bmatrix} 0_{3\times3}, \\ I_3 \end{bmatrix}, \; C = \begin{bmatrix} I_3 & 0_{3\times3} \end{bmatrix}.$$

Four agents are considered ($N = 4$) with the fixed undirected communication topology described by Fig. 3.3

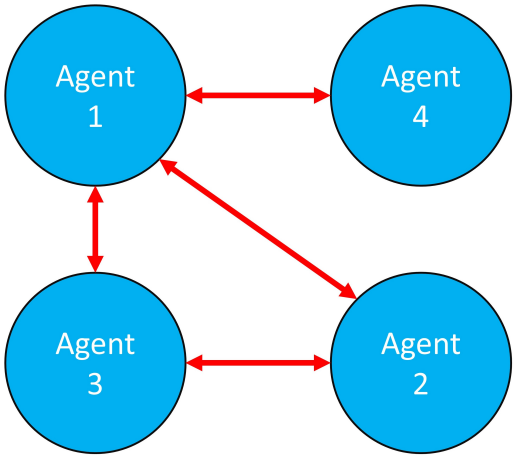


Fig. 3.3: Communication topology for simulations in this subsection.

and the corresponding Laplacian matrix is presented as follows:

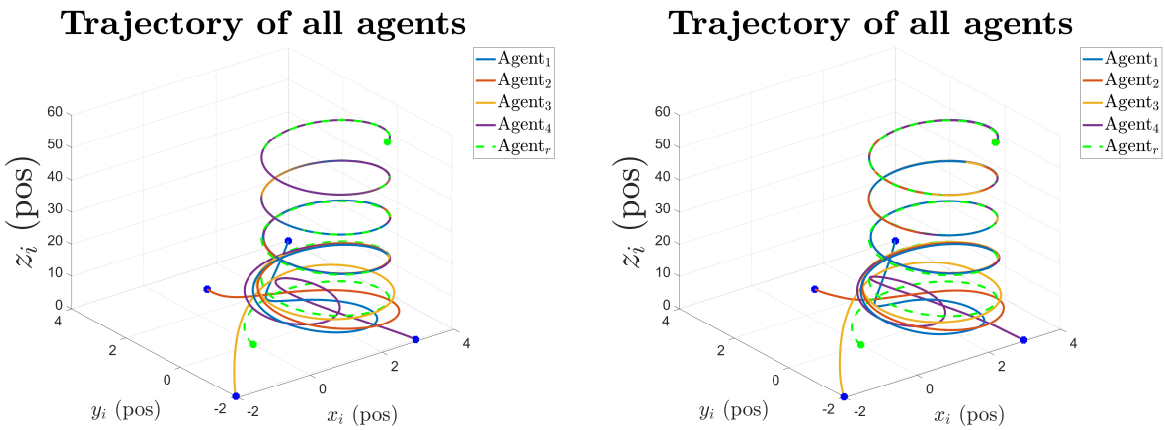
$$\mathcal{L} = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 2 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}.$$

Communication range limitations are not considered in this example. For the computation of the controller and observer gains, the LMI parameters in (3.21) are selected as follows: $\alpha = 1$, $\mu_1 = 2$, and $\mu_2 = 1000$. A comparison of two simulations are exposed. Simulation A consists of the observer-based time-triggered leader-following control and Simulation B consists of the event-triggered control. In both simulations, the agents' initial conditions are presented in Table 3.1.

Table 3.1: Initial positions and velocities of the agents.

Agent	Initial condition
1	$x_1(0) = [4, 4, 0, 0, 0, 0]^T$
2	$x_2(0) = [1, 3, 0, 0, 0, 0]^T$
3	$x_3(0) = [-2, -2, 0, 0, 0, 0]^T$
4	$x_4(0) = [3, -2, 0, 0, 0, 0]^T$

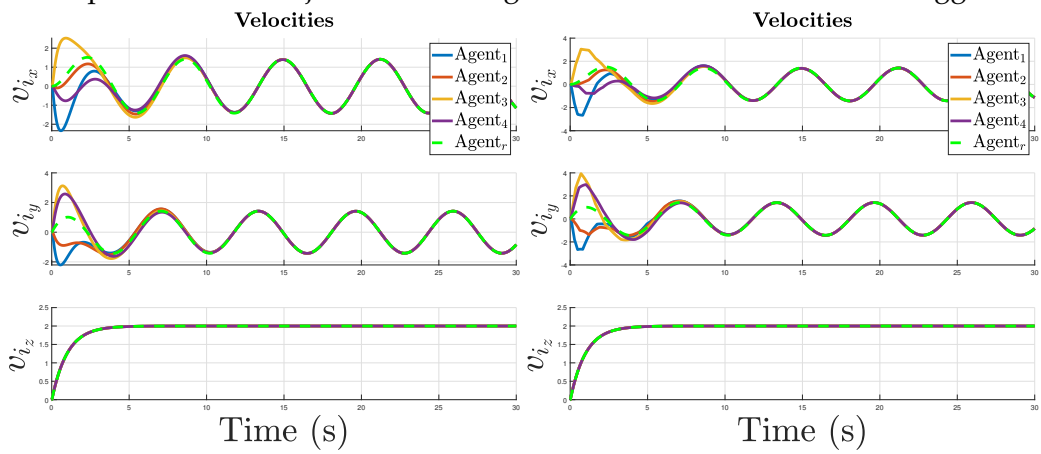
The leader's initial condition is $x_r(0) = [0, 0, 0, 0, 0, 0]^T$. In order to generate a desired behavior in the reference model, a generator trajectory is defined by the following control $u_r(t) = [2sin(t), 2cos(t), 2t]^T - x_r(t)$. Figs. 3.4 and 3.5 shows the profile evolution of the agent states (positions and velocities) along 30s of Simulation A (time-triggered control) and Simulation B (Event-triggered control). The leader's trajectories are illustrated in green color.



(a) Displacement of the agents using the time-triggered control

(b) Displacement of the agents using the event-triggered control

Fig. 3.4: Comparison of the trajectories of all agents between time- and event-triggered control.



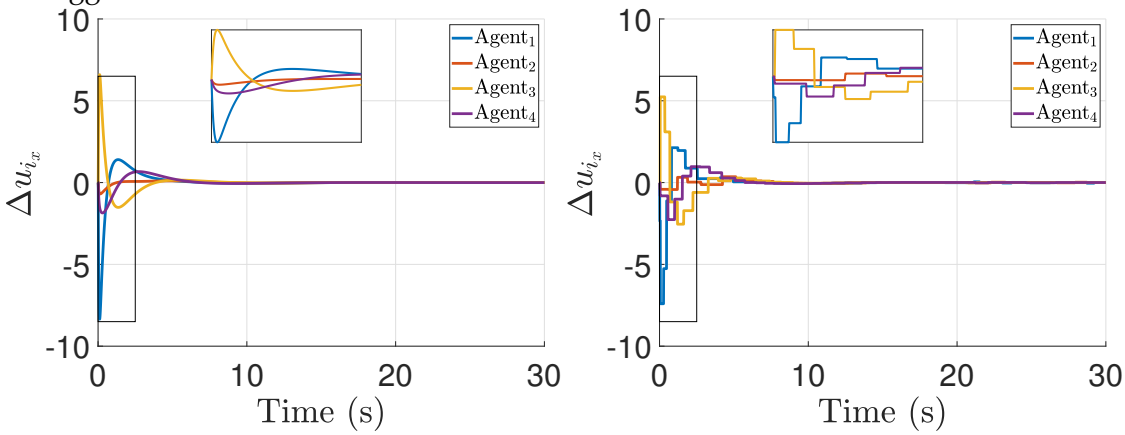
(a) Velocities of the agents using the time-triggered control

(b) Velocities of the agents using the event-triggered control

Fig. 3.5: Comparison of the velocities of all agents between time- and event-triggered control.

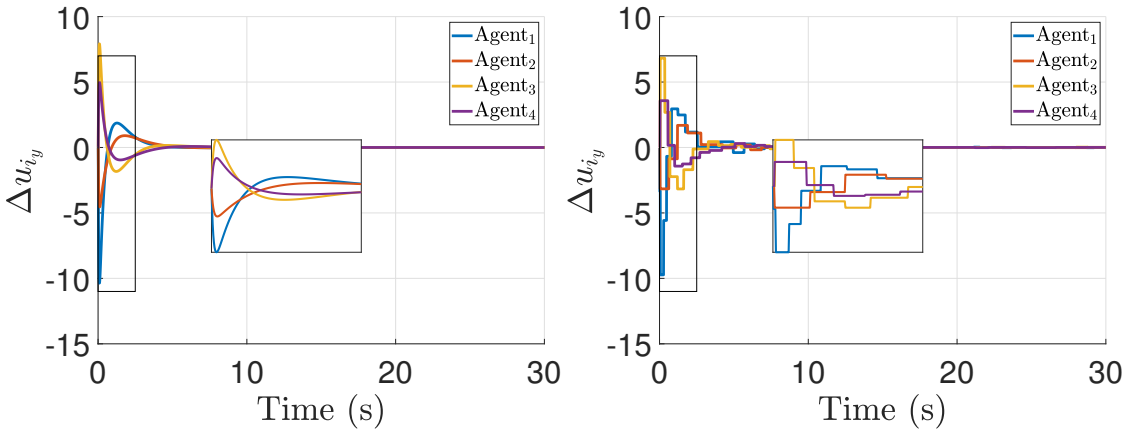
In order to compare the control law rate between the continuous and the event-triggered,

Figs. 3.6, 3.7, and 3.8 illustrate the profile of the time-triggered control law and the event-triggered control law .



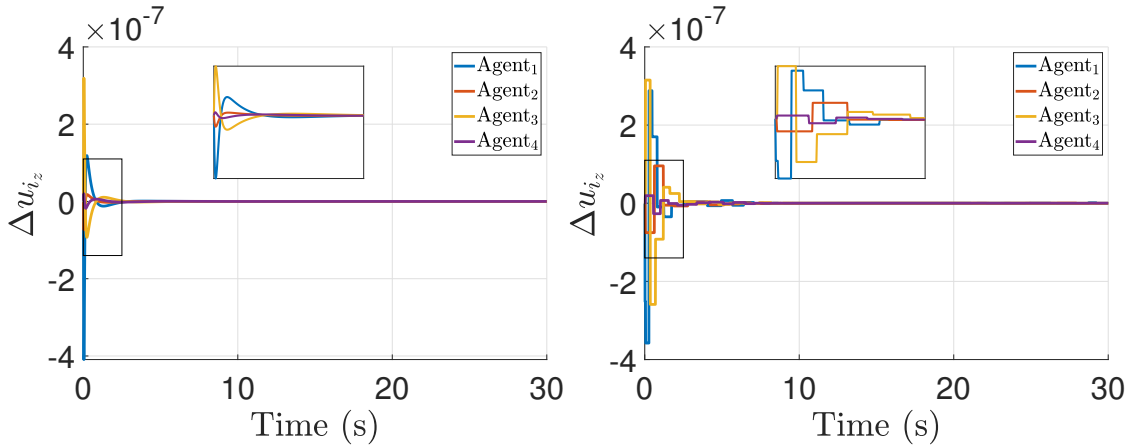
(a) Time-triggered (b) Event-triggered

Fig. 3.6: Comparison of the control profile $\Delta u_{i_x}(t)$ between time- and event-triggered control.



(a) Time-triggered (b) Event-triggered

Fig. 3.7: Comparison of the control profile $\Delta u_{i_y}(t)$ between time- and event-triggered control.



(a) Time-triggered (b) Event-triggered

Fig. 3.8: Comparison of the control profile $\Delta u_{i_z}(t)$ between time- and event-triggered control.

The event-triggered control reduces the exchange of information between neighboring agents and the update rate of the control law. Then, the min/max/mean event-time are presented in Table 3.2 for each agent as a measurement performance.

Table 3.2: Min/max/mean event-time

	Min	Max	Mean
Agent 1	0.02s	2.45s	0.79s
Agent 2	0.04s	3.01s	0.24s
Agent 3	0.05s	2.81s	0.10s
Agent 4	0.04s	2.53s	0.05s

This means that the maximum time between updates of the control law in agent 2 is 3.01s under the conditions described for this numerical example. In Fig. 3.9, the event function is shown on top when the inequality in (3.36) is fulfilled. On the bottom, the event error profile and the threshold is illustrated. The maximum threshold is given by $c_1 + c_2$ at $t = 0$.

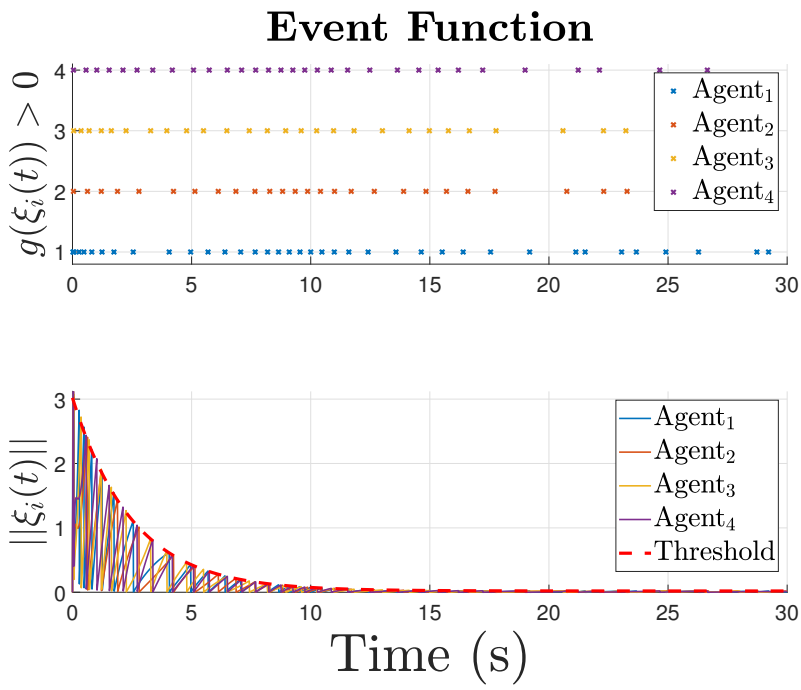


Fig. 3.9: Events of all the agents.

3.3 Observer-based fault-tolerant leader-following control for multi-agent systems

3.3.1 Leader-following consensus for multi-agent systems with actuator faults

Actuator faults are considered such that the nominal multi-agent system (3.1) is changed as follows:

$$\begin{aligned} \dot{x}_i(t) &= Ax_i(t) + B_f(\sigma_i(t))u_i(t), \\ y_i(t) &= Cx_i(t), \forall i = 1, 2, \ldots, N, \end{aligned} \tag{3.47}$$

where $u_i(t)$ is the reconfigured control input and the multiplicative actuator faults are represented in the matrix $B_f(\sigma_i(t))$ defined as follows:

$$B_f(\sigma_i(t)) = B\Upsilon(\sigma_i(t)) \tag{3.48}$$

where

$$\begin{aligned} &\Upsilon(\sigma_i(t)) = diag\left(\sigma_{i_1}(t), \sigma_{i_2}(t), \ldots, \sigma_{i_m}(t)\right), \\ &\text{with } 0 \leq \sigma_{i_k}(t) \leq 1, \end{aligned} \tag{3.49}$$

$\sigma_{i_k}(t)$ represents the *k-th* actuator effectiveness, such that $\sigma_{i_k}(t) = 0$ is a total failure of the *k-th* actuator and $\sigma_{i_k}(t) = 1$ is the healthy k-th actuator of agent i. The proposed fault-tolerant strategy is based on the reconfiguration of the multi-agent system (3.47) with a virtual actuator such that the agents follow the leader's trajectories in spite of the occurrence of faults. The faulty synchronization error dynamics are obtained as follows:

$$\begin{aligned} \dot{\delta}_i(t) &= A\delta_i(t) + B_f(\sigma_i(t))u_i(t), \\ y_{\delta_i}(t) &= C\delta_i(t). \end{aligned} \tag{3.50}$$

The following assumptions hold in this section.

Assumption 5. *The pair (A, B) is stabilizable (see Appendix 4).*

Assumption 6. *The pair (A, C) is detectable (see Appendix 4).*

Assumption 7. *The graph $\mathcal{G}$ is strongly connected as it is explained in the subsection 2.2.*

Assumption 8. *The magnitude of the faults are considered known, this is, $\hat{\sigma}_i(t) \simeq \sigma_i(t)$ based on an efficient FDI (fault detection and isolation) module in each agent.*

The virtual actuator can be either static or dynamic depending on whether the following rank condition is satisfied:

$$rank\left(B_f\left(\sigma_i(t)\right)\right) = rank\left(B\right) \neq 0. \tag{3.51}$$

If (3.51) holds, because the faults are only partial or a lost actuator can be expressed as a linear combination of the remaining actuators, the tolerance can be achieved through a redistribution of the control inputs. In this case, the reconfiguration structure is defined as follows:

$$u_i(t) = N_v(\hat{\sigma}_i(t))u_{n_i}(t) \tag{3.52}$$

where $\hat{\sigma}_i(t)$ is estimates of $\sigma_i(t)$. The nominal control input is defined as follows

$$u_{n_i}(t) = K_c\left[\sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{x}_i(t) - \hat{x}_j(t)\right) + \alpha_i\left(\hat{x}_i(t) - x_r(t)\right)\right] \tag{3.53}$$

where $K_c \in \mathbb{R}^{m\times n}$ is the control gain to be designed, α_i is the leader adjacency where $\alpha_i > 0$ if there is a directed edge from the leader to the agent i, and $\alpha_i = 0$ otherwise as reported in [86], [160], $\hat{x}_i(t)$ is the estimated state vector of the agent i, and $\hat{x}_j(t)$ is the estimated state vector corresponding to the neighboring agent j, and $N_v(\hat{\sigma}_i(t))$ is given by:

$$N_v(\hat{\sigma}_i(t)) = B_f(\hat{\sigma}_i(t))^{\dagger}B, \tag{3.54}$$

where $B_f(\hat{\sigma}_i(t))^{\dagger}$ denotes the pseudo inverse of $B_f(\hat{\sigma}_i(t))$. In cases where (3.51) is not satisfied, a simple redistribution of the control inputs is not enough. Hence, the fault tolerance is achieved through a dynamical virtual actuator described by a particular value of the matrix:

$$B^* = B_f(\sigma_i(t))N_v(\hat{\sigma}_i(t)). \tag{3.55}$$

Thus, the reconfiguration structure is expressed in such cases by:

$$u_i(t) = N_v(\hat{\sigma}_i(t))\left(u_{n_i}(t) - M_v x_{v_i}(t)\right), \tag{3.56}$$

where $M_v \in \mathbb{R}^{m\times n}$ is the virtual actuator gain matrix while the virtual actuator state $x_{v_i}(t)$ is calculated as follows:

$$\begin{aligned} \dot{x}_{v_i}(t) &= \left(A + B^*M_v\right)x_{v_i}(t) + \left(B - B^*\right)u_{n_i}(t), \\ y_{v_i}(t) &= Cx_{v_i}(t). \end{aligned} \tag{3.57}$$

The observer (3.7) is also modified as follows:

$$\begin{aligned} \dot{\hat{x}}_i(t) &= A\hat{x}_i(t) + Bu_{n_i}(t) + L_o\left(y_i(t) + y_{v_i}(t) - \hat{y}_i(t)\right), \\ \hat{y}_i(t) &= C\hat{x}_i(t). \end{aligned} \tag{3.58}$$

Let $\delta = \left[\delta_1^T, \delta_2^T, \ldots, \delta_N^T\right]^T$, $x_v = \left[x_{v_1}^T, x_{v_2}^T, \ldots, x_{v_N}^T\right]^T$, $\hat{x} = \left[\hat{x}_1^T, \hat{x}_2^T, \ldots, \hat{x}_N^T\right]^T$, then (3.50), (3.57), and (3.58) can be rewritten as follows:

$$\begin{aligned} \dot{x}_v &= \left(I_N \otimes (A + B^* M_v)\right) x_v + \left(\bar{\mathcal{L}} \otimes (B - B^*) K_c\right) \hat{\delta}, \\ \dot{\delta} &= \left(I_N \otimes A\right) \delta + \left(\bar{\mathcal{L}} \otimes B^* K_c\right) \hat{\delta} - \left(I_N \otimes B^* M_v\right) x_v, \\ \dot{\hat{x}} &= \left(I_N \otimes A\right) \hat{x} + \left(\bar{\mathcal{L}} \otimes B K_c\right) \hat{x} + \left(I_N \otimes L_o C\right) \left(x + x_v - \hat{x}\right). \end{aligned} \tag{3.59}$$

Let us introduce a state transformation in order to introduce new states $z_1 = x_v$, $z_2 = \delta + x_v$, $z_3 = \delta + x_v - \hat{\delta} = x + x_v - \hat{x}$, and $z = \left[z_1^T, z_2^T, z_3^T\right]^T$, then equation (3.59) can be rewritten as follows:

$$\begin{aligned} \dot{z} &= \tilde{A} z, \\ \tilde{A} &= \begin{bmatrix} I_N \otimes (A + B^* M_v) & \bar{\mathcal{L}} \otimes (B - B^*) K_c & -\bar{\mathcal{L}} \otimes (B - B^*) K_c \\ 0 & I_N \otimes A + \bar{\mathcal{L}} \otimes B K_c & -\bar{\mathcal{L}} \otimes B K_c \\ 0 & 0 & I_N \otimes (A - L_o C) \end{bmatrix} \end{aligned} \tag{3.60}$$

Remark 2. *If (3.51) is satisfied the problem can be solved without virtual actuator compensation. However, in cases where (3.51) is not satisfied, the virtual actuator compensation is necessary.*

3.3.2 Observer-based fault-tolerant leader-following control design

The following theorem extends the controller/observer design procedure proposed in [136] to take into account the virtual actuator-based reconfiguration strategy, and provides LMI-based conditions for the computation of the gains.

Theorem 3. *Consider the closed-loop system in (3.60). Given eigenvalues $\lambda_j(\bar{\mathcal{L}})$ ($j = 1, 2, \ldots, N$), assume that there exist symmetric matrices $\bar{P}_1 > 0$, $\bar{P}_2 > 0$, $P_3 > 0$, matrices N_v, N_c, M_o, and scalars $\mu_1 > 0$, $\mu_2 > 0$ such that the inequality*

$$\begin{bmatrix} Q_1 & Q_{2_j} & 0 & -Q_{2_j} & 0 & 0 & 0 \\ * & Q_{3_j} & 0 & 0 & 0 & -Q_{4_j} & 0 \\ * & * & Q_5 & 0 & I & 0 & I \\ * & * & * & -\mu_1^{-1}\bar{P}_2 & 0 & 0 & 0 \\ * & * & * & * & -\mu_1 \bar{P}_2 & 0 & 0 \\ * & * & * & * & * & -\mu_2^{-1}\bar{P}_2 & 0 \\ * & * & * & * & * & * & -\mu_2 \bar{P}_2 \end{bmatrix} < 0 \tag{3.61}$$

holds, where $Q_1 = He\{A\bar{P}_1 + B^* N_v\}$, $Q_{2_j} = \lambda_j (B - B^*) N_c$, $Q_{3_j} = He\{A\bar{P}_2 + \lambda_j B N_c\}$, $Q_{4_j} = \lambda_j B N_c$, and $Q_5 = He\{P_3 A - M_o C\}$. If the observer gain is calculated as $L_o = P_3^{-1} M_o$, the control gain is calculated by $K_c = N_c \bar{P}_2^{-1}$, and the virtual actuator gain is calculated as $M_v = N_v \bar{P}_1^{-1}$, then the observer-based fault-tolerant leader-following control problem for the multi-agent system under actuator faults is solved.

Proof. Let us choose a candidate Lyapunov function for the closed-loop system in (3.60) as follows:

$$V_2 = z^T diag\left(I_N \otimes P_1, I_N \otimes P_2, I_N \otimes P_3\right) z, \tag{3.62}$$

where $P_1 = P_1^T > 0$, $P_2 = P_2^T > 0$, and $P_3 = P_3^T > 0$. The derivative of V_2 along the trajectories of (3.60) is given by:

$$\begin{aligned}\dot{V}_2 =& 2z^T diag\left(I_N \otimes P_1, I_N \otimes P_2, I_N \otimes P_3\right)\dot{z},\\ =& 2z_1^T\left(I_N \otimes P_1\left(A + B^* M_v\right)\right) z_1 + 2z_1^T\left(\bar{\mathcal{L}} \otimes P_1\left(B - B^*\right) K_c\right) z_2 - 2z_1^T\left(\bar{\mathcal{L}} \otimes P_1\left(B - B^*\right) K_c\right) z_3\\ &+ 2z_2^T\left(I_N \otimes P_2 A + \bar{\mathcal{L}} \otimes P_2 B K_c\right) z_2 - 2z_2^T\left(\bar{\mathcal{L}} \otimes P_2 B K_c\right) z_3 + 2z_3^T\left(I_N \otimes P_3\left(A - L_o C\right)\right) z_3.\end{aligned} \tag{3.63}$$

Let us perform a spectral decomposition of the matrix $\bar{\mathcal{L}}$, such that $\bar{\mathcal{L}} = TJT^{-1}$ with an orthogonal matrix $T \in \mathbb{R}^{N\times N}$ and a diagonal matrix $J = diag\left(\lambda_1, \lambda_2, \ldots, \lambda_N\right) \in \mathbb{R}^{N\times N}$. Let us define the following change of coordinates:

$$\begin{aligned}\varphi_1 &= \left(T^{-1} \otimes I_N\right) z_1,\\ \varphi_2 &= \left(T^{-1} \otimes I_N\right) z_2,\\ \varphi_3 &= \left(T^{-1} \otimes I_N\right) z_3.\end{aligned} \tag{3.64}$$

Replacing (3.64) in (3.63) leads to:

$$\begin{aligned}\dot{V}_2 =& 2\varphi_1^T\left(I_N \otimes P_1\left(A + B^* M_v\right)\right)\varphi_1 + 2\varphi_1^T\left(J \otimes P_1\left(B - B^*\right) K_c\right)\varphi_2 - 2\varphi_1^T\left(J \otimes P_1\left(B - B^*\right) K_c\right)\varphi_3\\ &+ 2\varphi_2^T\left(I_N \otimes P_2 A + J \otimes P_2 B K_c\right)\varphi_2 - 2\varphi_2^T\left(J \otimes P_2 B K_c\right)\varphi_3 + 2\varphi_3^T\left(I_N \otimes P_3\left(A - L_o C\right)\right)\varphi_3.\end{aligned} \tag{3.65}$$

By Lemma 3, all the eigenvalues λ_j are positive; then, the above equation can be rewritten as follows:

$$\begin{aligned}\dot{V}_2 = \sum_{j=1}^{N} \varphi_{1_j}^T \Bigg[& He\left\{P_1\left(A + B^* M_v\right)\right\}\varphi_{1_j} + 2\varphi_{1_j}^T\left(\lambda_j P_1\left(B - B^*\right) K_c\right)\varphi_{2_j} - 2\varphi_{1_j}^T\left(\lambda_j P_1\left(B - B^*\right) K_c\right)\varphi_{3_i}\\ &+ \varphi_{2_i}^T He\left\{P_2 A + \lambda_j P_2 B K_c\right\}\varphi_{2_i} - 2\varphi_{2_j}^T\left(\lambda_j P_2 B K_c\right)\varphi_{3_j} + \varphi_{3_j}^T He\left\{P_3\left(A - L_o C\right)\right\}\varphi_{3_j}\Bigg].\end{aligned} \tag{3.66}$$

Let us define $\phi_j = \left[\varphi_{1_j}^T, \varphi_{2_j}^T, \varphi_{3_j}^T\right]^T$ so that:

$$\begin{aligned}\dot{V}_2 &= \sum_{j=1}^{N} \phi_j^T \Theta_{2_j} \phi_j,\\ \Theta_{2_j} &= \begin{bmatrix}\Theta_{11_j} & \lambda_j P_1\left(B - B^*\right) K_c & -\lambda_j P_1\left(B - B^*\right) K_c\\ * & \Theta_{22_j} & -\lambda_j P_2 B K_c\\ * & * & \Theta_{33}\end{bmatrix},\\ \Theta_{11} &= \mathrm{He}\left\{P_1\left(A + B^* M_v\right)\right\},\\ \Theta_{22_j} &= \mathrm{He}\left\{P_2\left(A + \lambda_j B K_c\right)\right\},\\ \Theta_{33} &= \mathrm{He}\left\{P_3\left(A - L_o C\right)\right\}.\end{aligned} \tag{3.67}$$

If $\Theta_{2_j} < 0, \forall j = 1, 2, \ldots, N$, then $\dot{V} < 0$. Due to the products of decision variables, this condition represents a bilinear matrix inequality, which is much harder to handle computationally due to lack of convexity [161]. With the goal of obtaining simpler-to-handle LMIs, the matrix

inequality Θ_j is pre- and post- multiplied by $diag(P_1^{-1}, P_2^{-1}, I)$, with $\bar{P}_1 = P_1^{-1}$ and $\bar{P}_2 = P_2^{-1}$ obtaining:

$$\begin{bmatrix} \mathscr{Q}_1 & \mathscr{Q}_{2_j} & -\mathscr{Q}_{2_j} \\ * & \mathscr{Q}_{3_j} & -\lambda_j BK_c \\ * & * & \mathscr{Q}_4 \end{bmatrix} < 0, \tag{3.68}$$

where $\mathscr{Q}_1 = \text{He}\left\{(A + B^* M_v)\,\bar{P}_1\right\}$, $\mathscr{Q}_{2_j} = \lambda_j\,(B - B^*)\,K_c\bar{P}_2$, $\mathscr{Q}_{3_j} = \text{He}\left\{\left(A + \lambda_j BK_c\right)\bar{P}_2\right\}$, and $\mathscr{Q}_4 = \text{He}\left\{P_3\,(A - L_o C)\right\}$. Note that (3.68) can be rewritten as follows:

$$\begin{bmatrix} \mathscr{Q}_1 & \mathscr{Q}_{2_j} & 0 \\ * & \mathscr{Q}_{3_j} & 0 \\ * & * & \mathscr{Q}_4 \end{bmatrix} + \text{He}\left\{\begin{bmatrix} -\mathscr{Q}_{2_j} \\ 0 \\ 0 \end{bmatrix}\begin{bmatrix} 0 & 0 & I \end{bmatrix}\right\} + \text{He}\left\{\begin{bmatrix} 0 \\ -\lambda_j BK_c \\ 0 \end{bmatrix}\begin{bmatrix} 0 & 0 & I \end{bmatrix}\right\} < 0. \tag{3.69}$$

Applying the Young relation (Lemma 4) in (3.69), the following inequality is obtained:

$$\begin{aligned} &\begin{bmatrix} \mathscr{Q}_1 & \mathscr{Q}_{2_j} & 0 \\ * & \mathscr{Q}_{3_j} & 0 \\ * & * & \mathscr{Q}_4 \end{bmatrix} + \mu_1 \begin{bmatrix} -\mathscr{Q}_{2_j} \\ 0 \\ 0 \end{bmatrix} \bar{P}_2 \begin{bmatrix} -\mathscr{Q}_{2_j}^T & 0 & 0 \end{bmatrix} + \mu_1^{-1} \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix} \bar{P}_2^{-1} \begin{bmatrix} 0 & 0 & I \end{bmatrix} \\ &+ \mu_2 \begin{bmatrix} 0 \\ -\lambda_j BK_c \\ 0 \end{bmatrix} \bar{P}_2 \begin{bmatrix} 0 & -\lambda_j\,(BK_c)^T & 0 \end{bmatrix} + \mu_2^{-1} \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix} \bar{P}_2^{-1} \begin{bmatrix} 0 & 0 & I \end{bmatrix} < 0, \end{aligned} \tag{3.70}$$

where $\mu_1 > 0$ and $\mu_2 > 0$. Using Schur complement (Lemma 5) in (3.70) and selecting $N_v = M_v\bar{P}_1$, $N_c = K_c\bar{P}_2$, and $M_o = P_3 L_o$, (3.61) is obtained, and this completes the proof. □

If (3.61) is satisfied, then the observer-based fault-tolerant leader-following control problem for the multi-agent system (3.60) with actuator faults is solved. The following section presents numerical examples in order to show the effectiveness of the proposed strategies.

3.3.3 Effectiveness of the fault-tolerant control strategy through numerical examples

In order to show the effectiveness of the fault-tolerant control design, the following example considers a group of 5 agents with the parameters:

$$A = \begin{bmatrix} -0.05 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & -3 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ -1 & -1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

The model has four states, two inputs, and two outputs. The communication topology for this example is described in Fig 3.10.

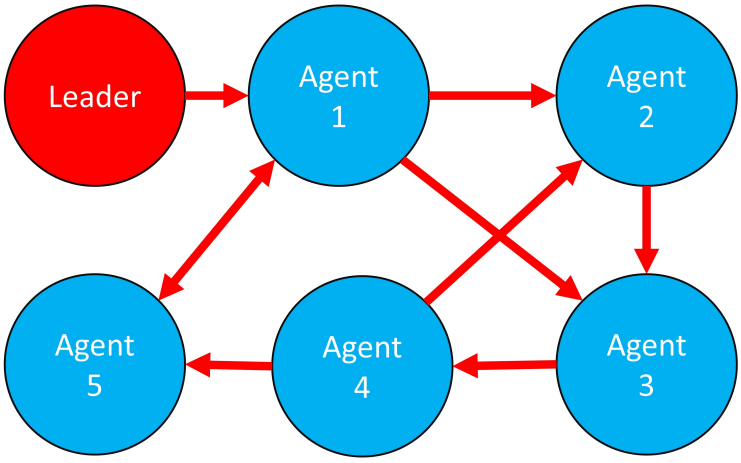


Fig. 3.10: Communication topology for simulations in this subsection.

The initial conditions of the agents are presented in Table 3.3. The LMI in Theorem 3 is solved with the following fixed values: $\mu_1 = 1$ and $\mu_2 = 0.01$, and the initial conditions of the leader are $x_r(0) = [-1, -1, 0.05, -0.05]^T$.

Table 3.3: Initial conditions of the agents in this subsection

Agent	Initial conditions
1	$x_1(0) = [0, 0, 0, 0]^T$
2	$x_2(0) = [1, 3, 0, 0]^T$
3	$x_3(0) = [-2, -2, 0, 0]^T$
4	$x_4(0) = [4, -2, 0, 0]^T$
5	$x_5(0) = [5, 3, 0, 0]^T$

Fig. 3.11 shows the performance of the synchronization error in the free fault case.

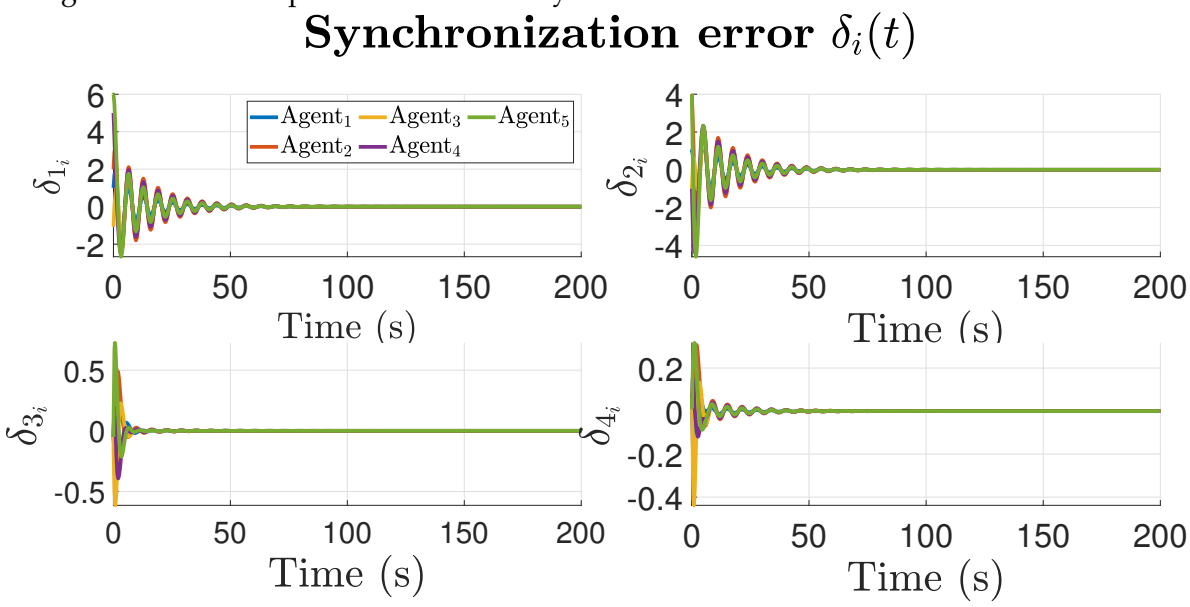
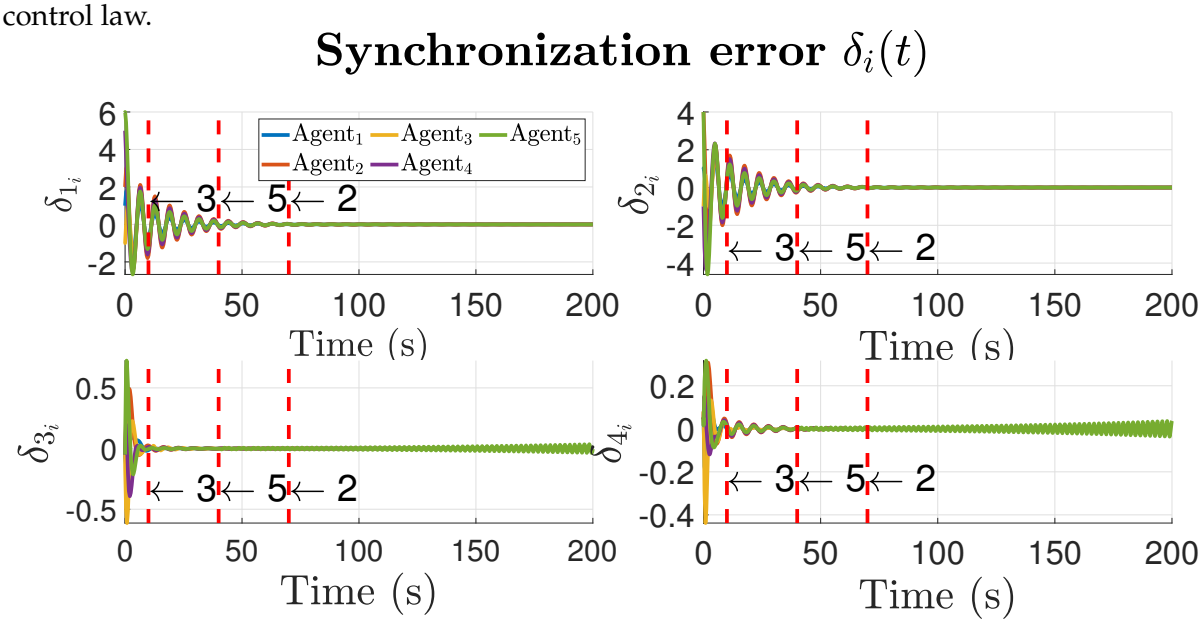


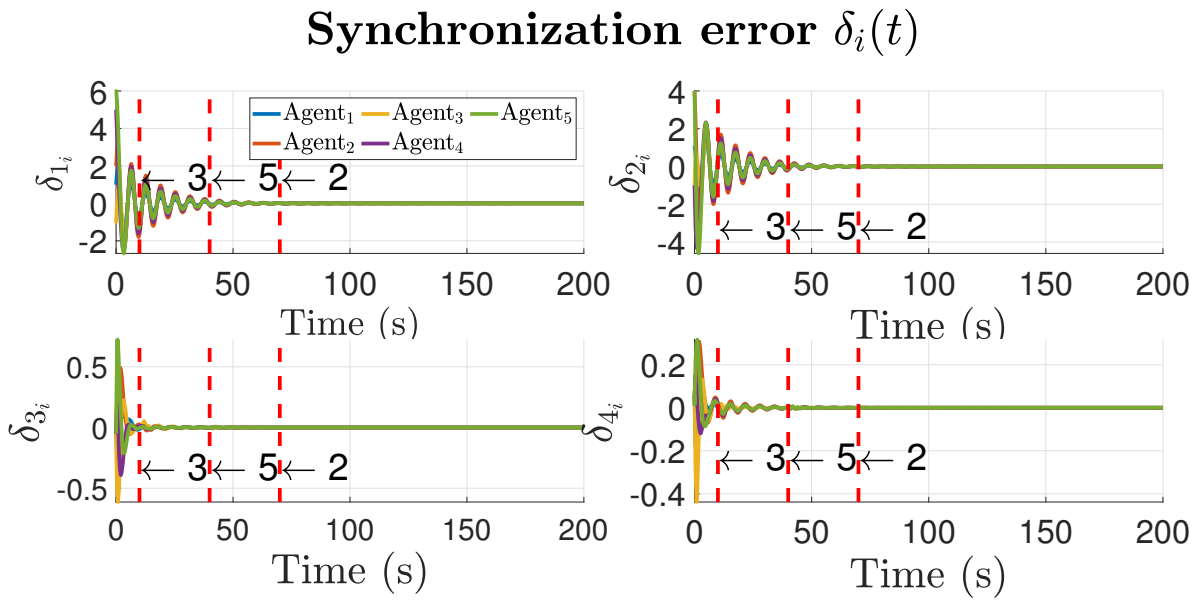
Fig. 3.11: Free fault case performance of the synchronization error between the leader and the agents.

The fault scenario in this example is considered as follows: agent 3 with a partial fault

diag $(\sigma_{3_1} = 1, \sigma_{3_2} = 0.5)$ starting from $t = 10$s, agent 5 with a lost actuator *diag* $(\sigma_{5_1} = 1, \sigma_{5_2} = 0)$ starting from $t = 40$s, and agent 2 with a lost actuator *diag* $(\sigma_{2_1} = 0, \sigma_{2_2} = 1)$ starting from $t = 70$s. Fig. 3.12(a) shows how the synchronization error becomes unstable due to the faults when the reconfiguration structure is not considered. After $t = 70$ due to the lost in the actuator 2 in agent 5 the consensus diverge. For the reconfiguration part, it is assumed that the magnitude of the faults is known and there is a delay in the fault detection and isolation when a fault occurs after that delay the virtual actuators reconfigure the control law. The delay between the fault occurrence and the fault detection is considered as 1s. Fig. 3.12(b) illustrates the synchronization error when the virtual actuators are considered in the reconfiguration of the control law.



(a) No reconfiguration



(b) Reconfiguration

Fig. 3.12: Comparison of the control performances with and without reconfiguration.

Fig. 3.13 shows the control law effort when the reconfiguration technique is used. There is a small overshot when the fault in agent 5 in actuator 1 appears due to the reconfiguration.

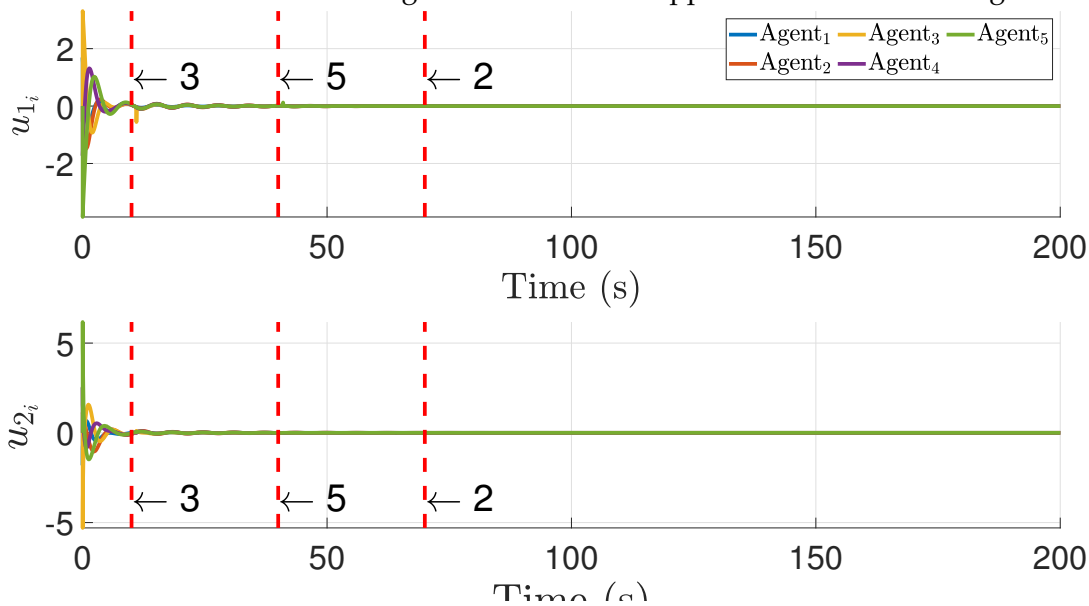


Fig. 3.13: Control law without reconfiguration.

3.4 Conclusions

In this chapter, an observer-based event-triggered and a fault-tolerant leader-following control for multi-agent system with actuator faults have been presented. Sufficient conditions have been obtained to guarantee that the controller, observer, and virtual actuator gains are computed such that all agents follow the leader agents' trajectories in spite of actuator faults. The main advantage of using virtual actuators is that the nominal consensus protocol has been used without re-tuning it. It has been shown that the event-triggered mechanism can be introduced to reduce the exchange of information between agents and the maximum update rate of the control law among agents. Simulation results have shown the effectiveness through numerical examples. It is worth mentioning that the design of the event-triggered mechanism only ensures that the consensus converges to a bounded region reducing the exchange of information and the rate of control law update. The control is designed separately from the triggering mechanism, however, it is interesting to consider both designs at the same time. Future will aim at considering an event-triggered fault-tolerant control combining both strategies. The following chapter addresses the problem where the agents present communication faults which are dependent on the distance, in addition, the proposed strategies have been implemented in a real platform of a fleet of UAVs.

Chapter 4

Time- and Event-triggered Formation Control for Multi-agent Systems Under Communication Faults

4.1 Introduction

A key point in leader-following consensus is the information exchange through digital networks [29]. However, delays, bandwidth limitations, or packet dropouts are challenges in real applications [30]. Delays is one of the most addressed problems for multi-agent systems considering additive, multiplicative, and measurement noises [75], in chaotic and Euler-Lagrange systems under switching topologies [72], including self-triggered and even-triggered mechanism in fractional-order systems [114], stochastic systems [115], time-varying delays [162], adaptive approaches [116], or event-triggered mechanism with constant delays [118].

As presented in Chapter 2, formation control can be achieved by using consensus algorithms. Works [16], [40]–[42] have studied that consensus algorithms can be extended to solve the formation control problem for mobile multi-agent systems by correctly satisfying some constraints. However, the problem of faults in the information exchange has not been widely studied when a degradation in the communication is considered. The information exchange is crucial in order to achieve the desired final formation. However, in many works the information exchange has been considered instantaneous. When network malfunctions occur, the communication may be delayed due to network degradation. The longest delay to reach a consensus is determined in [62] as $\tau_{ij} < \pi/\left(2\lambda_N(\mathcal{L})\right)$, where $\lambda_N(\mathcal{L})$ is the maximum eigenvalue of the Laplacian matrix. Nevertheless, this delay is considered invariant, therefore, in cases where delays are due to unexpected malfunctions in the communication network, the delays vary depending on time and other specific factors. In [21], communication faults are modeled as a modification in the weights of the adjacency matrix as a result of a malfunction in the information exchange. The communication faults in this work are considered as a delay-dependent on the distances between agents.

This chapter addresses the design of a time- and event-triggered formation control for multi-agent systems under communication faults which are modeled as smooth-varying delays dependent on the distance between agents. The main contribution of one control

strategy is to synthesize a robust control gain which guarantees stability of the synchronization error in spite of faults in the information exchange. Compared to existing literature, communication faults are clearly depending on the distance which will provide a more realistic assumption from real point of view. Fig. 4.1 shows the control strategy scheme of the time-triggered formation control which tolerates communication faults in the network as time-varying delays.

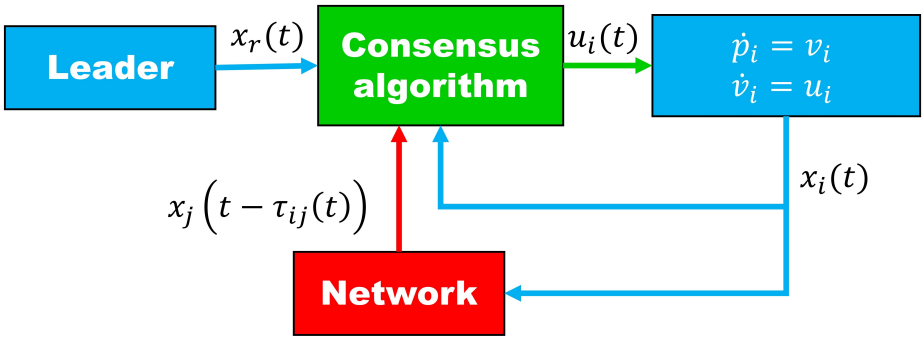


Fig. 4.1: Robust formation control strategy scheme for multi-agent systems under communication faults.

The second control strategy extends the first mentioned strategy adding an event-triggered mechanism based on the closed-loop system in order to reduce the information exchange and the control update rate. Based on such consideration, this development will be as close as possible to a relevant problem of multi-agent systems where reduced communication is a crucial point for energy consumption. All the agents follow the trajectories of a leader in a desired formation in spite of communication faults. Then, the proposed strategies are shown through numerical examples and in an experimental platform of a fleet of UAVs shaping a triangle. Fig. 4.2 illustrates the control strategy scheme using an event-triggered mechanism and the robust controller in order to reduce the information exchange, the control update rate, and the effects of the communication faults.

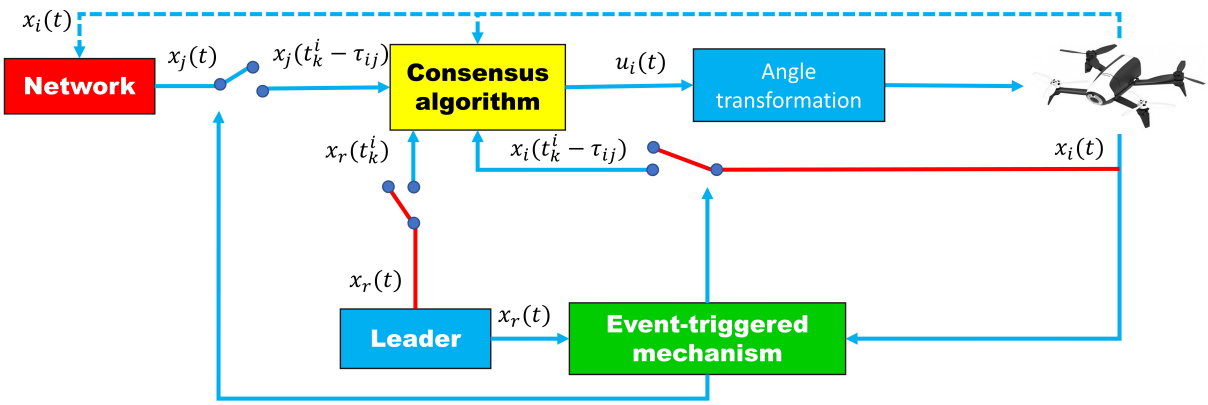


Fig. 4.2: Event-triggered formation control strategy scheme in a fleet of UAVs under communication faults.

This chapter is organized as follows. The time-triggered leader-following formation control is designed in Section 4.2. The event-triggered leader-following formation control design is provided in Section 4.3. A comparison of the proposed approach and the classical formation control is illustrated in Section 4.4. Finally, the conclusions of this chapter are drawn in Section

4.6.

4.2 Time-triggered leader-following formation control design

Consider a double integrator multi-agent system of N agents:

$$\begin{aligned} \dot{p}_i(t) &= v_i(t), \\ \dot{v}_i(t) &= u_i(t), \end{aligned} \tag{4.1}$$

$p_i(t), v_i(t), u_i(t) \in \mathbb{R}^n$ are respectively the position, velocity, and acceleration input ($\forall i = 1, 2, \ldots, N$) in a n dimensional Euclidean space with i as the *ith* agent. Let us define a virtual leader described by $p_r(t)$ and $v_r(t)$, where $p_r(t)$ and $v_r(t) \in \mathbb{R}^n$ are respectively the position and velocity of the virtual agent. According to [1] to achieve a desire formation using a consensus algorithm, a rigid desired-position formation is necessary to define from the agent i and its neighbors j as h_i and $h_j \in \mathbb{R}^n$. Then, the classical leader-following formation control reported in [1], [41] under communication faults is defined as follows:

$$\begin{aligned} u_i(t) = &\sum_{j \in \mathcal{N}_i} a_{ij} \left[p_j(t - \tau_{ij}(t)) - p_i(t - \tau_{ij}(t)) - (h_j - h_i) \right] \\ &+ \sum_{j \in \mathcal{N}_i} a_{ij} \left[v_j(t - \tau_{ij}(t)) - v_i(t - \tau_{ij}(t)) \right] - (v_i(t) - v_r(t)) \end{aligned} \tag{4.2}$$

where $\mathcal{N}_i$ is the set of i's neighbors, $\tau_{ij}(t)$ is a communication fault between agent i and its neighbors [76]. The following definition is given to define quantitatively when the formation has been reached based on agent i and its neighboring agents.

Definition 2. *The agents in (4.1) under (4.2) reach the formation in a consensus, if:*

$$\begin{aligned} &\lim_{t \to \infty} \left\| (p_i(t) - h_i(t)) - \left(p_j(t) - h_j(t) \right) \right\| \to 0, \\ &\forall i = 1, \ldots, N. \end{aligned} \tag{4.3}$$

Remark 3. *When $\tau_{ij}(t) = 0$, the leader-following formation control can be solved as reported in [1] if and only if $\mathcal{G}$ contains a spanning tree. When $\tau_{ij}(t) > 0$, the longest delay to reach a consensus is determined in [62] as $\tau_{ij} < \pi / \left(2\lambda_N(\mathcal{L})\right)$, where $\lambda_N(\mathcal{L})$ is the maximum eigenvalue of the Laplacian matrix. In [62], the delay is considered constant and with the same value for all agents ($\tau_{ij}(t) = \tau$). This chapter considers varying communication faults which are modeled as smooth delays dependent on the distances between agents. A control strategy is designed in order to tolerate longer delays.*

The communication faults are considered dependent on the agent positions and described by the following function:

$$\tau_{ij}(t) = \left(\gamma - \gamma \mathbf{e}^{-\beta_1 \left\| p_i(t) - p_j(t) \right\|} \right) \left(0.5 + 0.5 tanh \left(\beta_2 (t - t_f) \right) \right), \tag{4.4}$$

where γ, β_1, and β_2 are positive scalars; t_f is the time at which the fault occurs. The proposed function $\tau_{ij}(t)$ has been constructed such that the maximum value approximately is $\tau_{ij}(t) = \gamma$ because as the agents move away from each other, the function $\mathbf{e}^{-\beta_1 \left\| p_i(t) - p_j(t) \right\|}$ decreases to zero. So that the faults are not abrupt, the term $\left(0.5 + 0.5 tanh \left(\beta_2 (t - t_f) \right) \right)$ is added. The domain of this term is between zero and one which depends on the time of fault occurrence. Communication faults can be associated to a degradation of the communication between agents in link with their distance. Similarly, such assumption has been considered in stochastic approach [163]. The following elemental assumptions are hold in this chapter.

Assumption 9. *The graph $\mathcal{G}$ is undirected and connected.*

Assumption 10. *All the agents have information of the virtual leader's states.*

Assumption 11. $\dot{\tau}_{ij}(t) \leq d_1 < 1, \forall i \neq j, j \in \mathcal{N}_i$.

Let us define the synchronization error between the agent i and the virtual agent $\bar{p}_i(t) = p_i(t) - p_r(t)$, $\bar{v}_i(t) = v_i(t) - v_r(t)$. Let $\delta_i(t) = \left[\bar{p}_i(t)^T - h_i^T, \bar{v}_i(t)^T\right]^T$. Then, the synchronization error dynamic is defined as follows:

$$\dot{\delta}_i(t) = A\delta_i(t) + Bu_i(t), \text{ with } A = \begin{bmatrix} 0_{n\times n} & I_n \\ 0_{n\times n} & 0_{n\times n} \end{bmatrix}, B = \begin{bmatrix} 0_{n\times n} \\ I_n \end{bmatrix}. \tag{4.5}$$

The control (4.2) is modified as follows:

$$\begin{aligned} u_i(t) = &\sum_{j\in\mathcal{N}_i} a_{ij}\left[\bar{p}_j(t-\tau_{ij}(t)) - \bar{p}_i(t-\tau_{ij}(t)) - (h_j - h_i)\right] \\ &+ \sum_{j\in\mathcal{N}_i} a_{ij}\left[\bar{v}_j(t-\tau_{ij}(t)) - \bar{v}_i(t-\tau_{ij}(t))\right] - \bar{v}_i(t) \end{aligned} \tag{4.6}$$

Based on change coordinates (4.5), adding the control gain K_c and the scalar α, the proposed leader-following formation control under communication faults when $\tau_{ij}(t) > 0$ is defined as follows:

$$u_i(t) = K_c\left[\sum_{j\in\mathcal{N}_i} a_{ij}\left(\delta_i(t-\tau_{ij}(t)) - \delta_j(t-\tau_{ij}(t))\right) + \alpha\delta_i(t)\right] \tag{4.7}$$

where $K_c \in \mathbb{R}^{n\times 2n}$ is the control gain to be designed and $\alpha > 0$ must be a positive constant. Note that, if $K_c = \begin{bmatrix} -I_n & -I_n \end{bmatrix}$ and $\alpha = 1$, the classical formation control (4.2), it recovers. Based on (4.7), (4.5) is equivalent to:

$$\dot{\delta}_i(t) = A\delta_i(t) + BK_c\left[\sum_{j\in\mathcal{N}_i} a_{ij}\left(\delta_i(t-\tau_{ij}(t)) - \delta_j(t-\tau_{ij}(t))\right) + \alpha\delta_i(t)\right]. \tag{4.8}$$

Let $\delta(t) = \left[\delta_1(t)^T, \delta_2(t)^T, \ldots, \delta_N(t)^T\right]^T$ and $\delta(t-\tau(t)) = \left[\delta_1(t-\tau_{1j}(t))^T, \delta_2(t-\tau_{2j}(t))^T, \ldots, \delta_N(t-\tau_{Nj}(t))^T\right]^T$, then, the synchronization error model (4.8) is rewritten as follows:

$$\dot{\delta}(t) = \left(I_N \otimes (A + \alpha BK_c)\right)\delta(t) + \left(\mathcal{L} \otimes BK_c\right)\delta(t-\tau(t)). \tag{4.9}$$

The following theorem provides LMI conditions for the computation of the control gain K_c which guarantees stability of the synchronization error.

Theorem 4. *Considering the closed-loop system in (4.9), given the non-zero eigenvalues $\lambda_i(\mathcal{L})$ of the Laplacian matrix ($i = 2, 3, \ldots, N$), the scalar $\alpha > 0$, scalars $\mu_1 > 0$, $\mu_2 > 0$, and $\dot{\tau}_{ij} \leq d_1 < 1$; if there exist matrices $P_1 = P_1^T > 0$, $P_2 = P_2^T > 0$, and K_c such that the inequality (4.10) holds, then the leader-following formation control problem for the system (4.1) under communication faults (4.4) is quadratically stable under (4.7).*

$$\begin{bmatrix} \mathrm{He}\{P_1A\} + I - \frac{2P_1}{\mu_1} + P_2 & 0 & -\lambda_i P_1 B & \frac{P_1}{\mu_1} + \mu_1\alpha BK_c \\ * & -(1-d_1)P_2 & -\mu_2 K_c^T & 0 \\ * & * & -2\mu_2 I & 0 \\ * & * & * & -I \end{bmatrix} < 0 \tag{4.10}$$

Proof. Let us define a Lyapunov functional as follows:

$$V_3 = \delta(t)^T \left(I_N \otimes P_1\right) \delta(t) + \int_{t-\tau}^{t} \delta(s)^T \left(I_N \otimes P_2\right) \delta(s) ds \tag{4.11}$$

The derivative V along the trajectories of (4.9) is given by:

$$\begin{aligned} \dot{V}_3 = 2\delta(t)^T \left(I_N \otimes P_1\right) \dot{\delta}(t) + \delta(t)^T \left(I_N \otimes P_2\right) \delta(t) \\ - (1 - \dot{\tau}(t))\delta(t-\tau(t))^T \left(I_N \otimes P_2\right) \delta(t-\tau(t)). \end{aligned} \tag{4.12}$$

According to [164], (4.12) is negative-definite when

$$\begin{aligned} \dot{V}_3 = 2\delta(t)^T \left(I_N \otimes P_1\right) \dot{\delta}(t) + \delta(t)^T \left(I_N \otimes P_2\right) \delta(t) \\ - (1 - d_1)\delta(t-\tau(t))^T \left(I_N \otimes P_2\right) \delta(t-\tau(t)) < 0. \end{aligned} \tag{4.13}$$

where $\dot{\tau}(t) \leq d_1 < 1$ according to Assumption 11. Thus,

$$\begin{aligned} \dot{V}_3 =& 2\delta(t)^T \left(I_N \otimes (P_1 A + \alpha P_1 B K_c)\right) \delta(t) + 2\delta(t)^T \left(\mathcal{L} \otimes P_1 B K_c\right) \delta(t-\tau(t)) \\ &+ \delta(t)^T \left(I_N \otimes P_2\right) \delta(t) - (1-d_1)\delta(t-\tau(t))^T \left(I_N \otimes P_2\right) \delta(t-\tau(t)). \end{aligned} \tag{4.14}$$

Let us perform a spectral decomposition of the Laplacian matrix, such that $\mathcal{L} = TJT^{-1}$ with an invertible matrix $T \in \mathbb{R}^{N\times N}$ and a diagonal matrix $J = diag\left(\lambda_1, \lambda_2, \lambda_N\right) \in \mathbb{R}^{N\times N}$ where $\lambda_1 = 0$. By Lemma 2, the eigenvectors associated with the eigenvalues of $\mathcal{L}$ form a base which are used to construct the invertible matrix T. Then, let us define the following change of coordinates:

$$\begin{aligned} \varphi(t) &= \left(T^{-1} \otimes I_N\right) \delta(t), \\ \varphi(t-\tau(t)) &= \left(T^{-1} \otimes I_N\right) \delta(t-\tau(t)). \end{aligned} \tag{4.15}$$

Replacing (4.15) in (4.14) leads to:

$$\begin{aligned} \dot{V}_3 =& 2\varphi(t)^T \left(I_N \otimes (P_1 A + \alpha P_1 B K_c)\right) \varphi(t) + 2\varphi(t)^T \left(J \otimes P_1 B K_c\right) \varphi(t-\tau(t)) \\ &+ \varphi(t)^T \left(I_N \otimes P_2\right) \varphi(t) - (1-d_1)\varphi(t-\tau(t))^T \left(I_N \otimes P_2\right) \varphi(t-\tau(t)). \end{aligned} \tag{4.16}$$

By Lemma 2, it is obtained $\varphi_1(t) = 0$ and $\varphi_1(t-\tau(t)) = 0$ due to $\lambda_1 = 0$, then (4.16) can be rewritten as follows:

$$\begin{aligned} \dot{V}_3 = \sum_{i=2}^{N} \varphi_i(t)^T \mathrm{He}\left\{P_1 A + \alpha P_1 B K_c\right\} \varphi_i(t) + 2\sum_{i=2}^{N} \varphi_i(t)^T \left(\lambda_i P_1 B K_c\right) \varphi_i(t-\tau) \\ + \sum_{i=2}^{N} \varphi_i(t)^T \left(P_2\right) \varphi_i(t) - (1-d_1) \sum_{i=2}^{N} \varphi_i(t-\tau)^T \left(P_2\right) \varphi_i(t-\tau). \end{aligned} \tag{4.17}$$

Then, (4.17) can be rewritten as follows:

$$\begin{aligned} \dot{V}_3 &= \sum_{i=2}^{N} \begin{bmatrix} \varphi_i(t) & \varphi_i(t-\tau) \end{bmatrix} \Theta_{3_i} \begin{bmatrix} \varphi_i(t) \\ \varphi_i(t-\tau) \end{bmatrix} < 0, \\ \Theta_{3_i} &= \begin{bmatrix} \mathrm{He}\left\{P_1 A + \alpha P_1 B K_c\right\} + P_2 & \lambda_i P_1 B K_c \\ * & -(1-d_1) P_2 \end{bmatrix}. \end{aligned} \tag{4.18}$$

If $\Theta_{3_i} < 0$ is definite-negative $\forall i = 2, 3, \ldots, N$, then, $\dot{V}_3 < 0$. Thus, the leader-following formation control problem for the system (4.1) under communication faults is quadratically stable under (4.7). The inequality (4.18) guarantees the asymptotic stability, however, the matrices cannot be calculated using conventional tools. Using Schur complement in (4.10), it is obtained:

$$\begin{bmatrix} Q_1 & 0 & -\lambda_i P_1 B \\ * & -(1-d_1) P_2 & -\mu_2 K_c^T \\ * & * & -2\mu_2 I \end{bmatrix} < 0. \tag{4.19}$$

where $Q_1 = \mathrm{He}\{P_1 A\} + I - \frac{2P_1}{\mu_1} + P_2 + \left(\frac{P_1}{\mu_1} + \mu_1 \alpha B K_c\right)^T \left(\frac{P_1}{\mu_1} + \mu_1 \alpha B K_c\right)$. Multiplying (4.19) the left and the right sides by $\begin{bmatrix} I & 0 & 0 \\ 0 & I & -K_c^T \end{bmatrix}$ and its transpose, it is obtained:

$$\begin{bmatrix} Q_1 & \lambda_i P_1 B K_c \\ * & -(1-d_1) P_2 \end{bmatrix} < 0. \tag{4.20}$$

Note that,

$$\begin{aligned} &\mathrm{He}\{P_1 A + \alpha P_1 B K_c\} + P_2 \leq \mathrm{He}\{P_1 A + \alpha P_1 B K_c\} + P_2 + \alpha^2 (P_1 B K_c)^T (P_1 B K_c) \\ &= \mathrm{He}\{P_1 A\} + P_2 - \frac{P_1^2}{\mu_1^2} + \left(\frac{P_1}{\mu_1} + \alpha B K_c\right)^T \left(\frac{P_1}{\mu_1} + \alpha B K_c\right), \end{aligned} \tag{4.21}$$

where $\mu_1 > 0$. The following inequality is introduced:

$$\left(I - \frac{P_1}{\mu_1}\right)\left(I - \frac{P_1}{\mu_1}\right) \geq 0, I - \frac{2P_1}{\mu_1} \geq -\frac{P_1^2}{\mu_1^2}. \tag{4.22}$$

Note that,

$$\begin{aligned} &\mathrm{He}\{P_1 A + \alpha P_1 B K_c\} + P_2 \leq \\ &\mathrm{He}\{P_1 A\} + I + P_2 - \frac{2P_1}{\mu_1} + \left(\frac{P_1}{\mu_1} + \alpha B K_c\right)^T \left(\frac{P_1}{\mu_1} + \alpha B K_c\right). \end{aligned} \tag{4.23}$$

Combining (4.23) and (4.20), it recovers:

$$\begin{bmatrix} \mathrm{He}\{P_1 A + \alpha P_1 B K_c\} + P_2 & \lambda_i P_1 B K_c \\ * & -(1-d_1) P_2 \end{bmatrix} < 0. \tag{4.24}$$

The inequality (4.24) is equal to (4.18), thus, the linear matrix inequality (4.10) satisfies (4.18) and the leader-following formation control problem for the system (4.1) under communication faults is quadratically stable under (4.7), thus completing the proof. □

If all conditions hold and the matrices exist, Theorem 4 guarantees a control gain design K_c which is used to achieve and to maintain a desired formation despite communication faults under Assumptions 11. When one or all the agents have communication faults, each faulty agent switches to the control gain K_c reducing the malfunction effects. The following section an even-triggered mechanism is added in order to reduce the information exchange and the control update rate.

4.3 Event-triggered leader-following formation control design

The update of the control law action in event-triggered approaches depends on an event error. This event error is calculated based on the last and the current state values. When the magnitude of the event error exceeds a threshold, the control law value is updated, otherwise, the control law keeps the last calculated value. The control in (4.7) is modified in order to design an event-triggered mechanism as follows:

$$u_i(t) = K_c \left[\sum_{j \in \mathcal{N}_i} a_{ij} \left(\delta_i \left(t_k^i - \tau_{ij}(t_k^i) \right) - \delta_j \left(t_k^i - \tau_{ij}(t_k^i) \right) \right) + \alpha \delta_i \left(t_k^i \right) \right], \tag{4.25}$$

where $\delta_i \left(t_k^i\right)$ and $\delta_j \left(t_k^i\right)$ are the last value of the synchronization error of the agent i and j, $\delta_i \left(t_k^i - \tau_{ij}(t_k^i)\right)$ and $\delta_j \left(t_k^i - \tau_{ij}(t_k^i)\right)$ are the last value of the delayed synchronization error of the agent i and j, at time t_k^i which is the last time event of the agent i with $k \in \mathbb{Z}^+$. Te sequence of event-times $0 \leq t_0^i \leq t_1^i \ldots$ of the agent i is defined as $t_{k+1}^i = \inf\{t : t > t_k^i, f_i(\zeta_i(t)) > 0\}$. The agent i requests the information of the agent j at the event t_{k+1}^i in order to update the control law, otherwise, the control law keeps the last computed value. Let us define the event error as follows:

$$\zeta_i(t) = \delta_i(t_k^i) - \delta_i(t), \tag{4.26}$$

According to [27], if the leader-following consensus is quadratically stable, then, the following event function can be considered:

$$f_i(t) = \|\zeta_i(t)\| - \left(c_1 + c_2 \mathbf{e}^{-c_3 t}\right), \tag{4.27}$$

where c_1 and c_2 are positive scalars, $0 < c_3 < \left|\lambda_{min}(\bar{A})\right|$ and $\lambda_{min}\left(\bar{A}\right)$ is the minimum eigenvalue of $\bar{A}$ where $\bar{A} = A + \alpha B K_c$. In order to illustrate the effectiveness of the proposed strategy, simulation, and real implementations in a fleet of UAVs are presented in the following section.

4.4 Comparison of the proposed approach and the classical formation control

A comparison of six simulations is considered. Simulation A1 and Simulation A2 use the classical algorithm with one faulty agent and with faults in the all agents respectively (this is $K_c = -\begin{bmatrix} I_3 & I_3 \end{bmatrix}$); Simulation B1 and Simulation B2 use the time-triggered proposed strategy with one faulty agent and with faults in the all agents respectively; Simulation C1 and Simulation C2 use the event-triggered proposed strategy with one faulty agent and with faults in the all agents respectively. A fleet of six agents ($N = 6$) is considered. Table 4.1 presents the desired positions and the agents' initial positions used in these example.

Table 4.1: Initial positions of the agents and the desired final configuration.

Desired positions	Agents' initial positions
$p_1(0) = [0,0,0]^T$	$h_1 = [0,0,0,0,0,0]^T$
$p_2(0) = [1,3,0]^T$	$h_2 = [4,0,0,0,0,0]^T$
$p_3(0) = [-2,-2,0]^T$	$h_3 = [6,2\sqrt{3},0,0,0,0]^T$
$p_4(0) = [4,-2,0]^T$	$h_4 = [4,4\sqrt{3},0,0,0,0]^T$
$p_5(0) = [5,3,0]^T$	$h_5 = [0,4\sqrt{3},0,0,0,0]^T$
$p_6(0) = [5,8,0]^T$	$h_6 = [-2,2\sqrt{3},0,0,0,0]^T$

The agents' initial velocities are: $v_i(0) = 0, \forall i = [1,\ldots,6]$. The target velocity is $v_r = [0,0,0.2]^T$. The communication topology used in these simulations is described in Fig. 4.3 and the following Laplacian matrix:

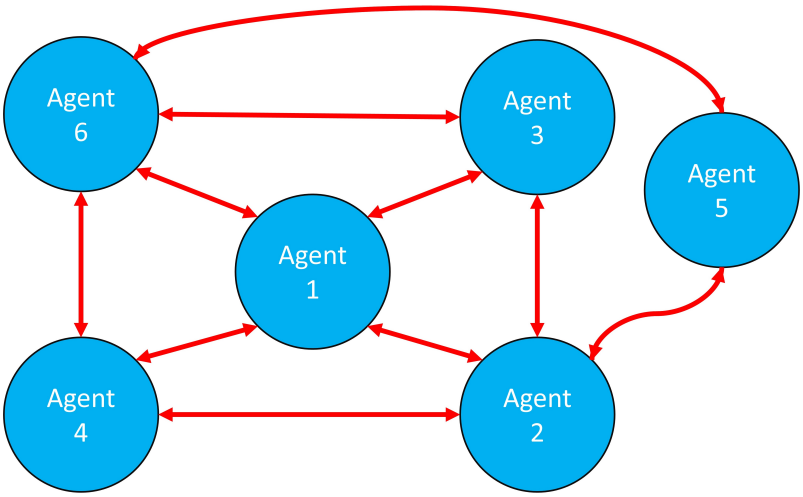


Fig. 4.3: Topology of the communication used in these simulations.

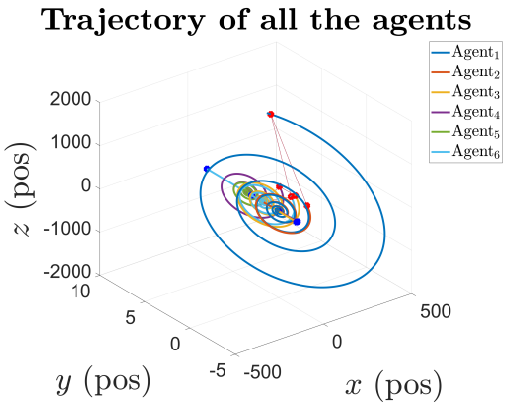
$$\mathcal{L} = \begin{bmatrix} 4 & -1 & -1 & -1 & 0 & -1 \\ -1 & 4 & -1 & -1 & -1 & 0 \\ -1 & -1 & 3 & 0 & 0 & -1 \\ -1 & -1 & 0 & 3 & 0 & -1 \\ 0 & -1 & 0 & 0 & 2 & -1 \\ -1 & 0 & -1 & -1 & -1 & 4 \end{bmatrix}.$$

The communication fault parameters are $\gamma = 0.62$, $\beta_1 = 1$, and $\beta_2 = 0.6$. For Simulation B1, B2, C1, and C2, the control gain K_c is computed with following parameters: $\mu_1 = 1$, $\mu_2 = 5$, $d_1 = 0.2$, $\alpha = 1$ obtaining the following matrix:

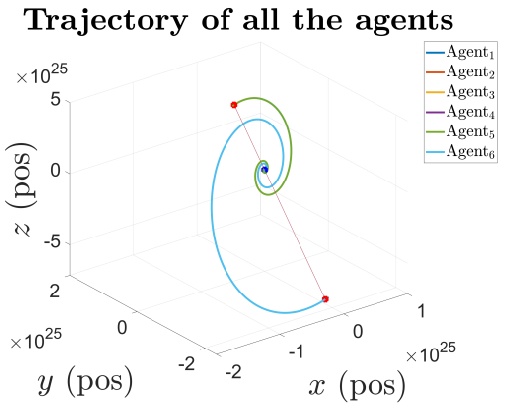
$$K_c = \begin{bmatrix} -0.1185 & 0.0000 & 0.0000 & -0.2488 & -0.0000 & -0.0000 \\ 0.0000 & -0.1185 & 0.0000 & -0.0000 & -0.2488 & -0.0000 \\ 0.0000 & 0.0000 & -0.1185 & -0.0000 & -0.0000 & -0.2488 \end{bmatrix}.$$

This control gain is used in both the time-triggered and event-triggered approach. The event-triggered function has the following parameters: $c_1 = 0.02$, $c_2 = 3$, and $c_3 = 0.1$. All simulations start free fault. After 5s for simulation A1 and B1, the agent 1 presents communication faults and for simulations A2 and B2, all agents present communication faults. Fig. 4.4 shows a comparative of the final position of the agent. The agents cannot maintain

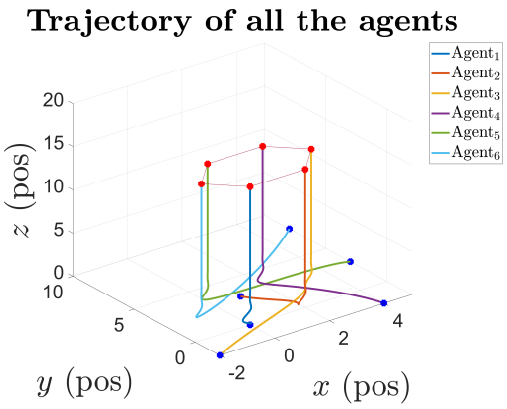
the desired formation and they become unstable in Figs. 4.4(a) and 4.4(c) which correspond to Simulation A1 and A2. All agents maintain the formation despite the communication fault in Figs. 4.4(b), 4.4(d), 4.4(e), and 4.4(f) which correspond to the time- and event-triggered strategies, however, the event-triggered strategy presents oscillations which decrease.



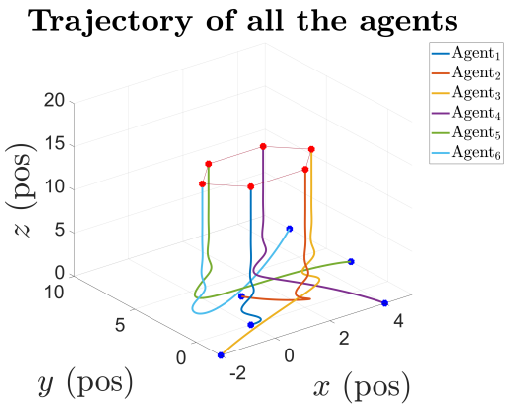
(a) Classical formation control with one faulty agent (Simulation A1)



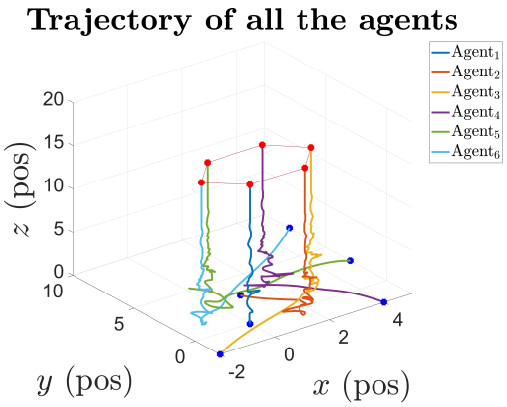
(b) Time-triggered strategy with one faulty agent (Simulation B1)



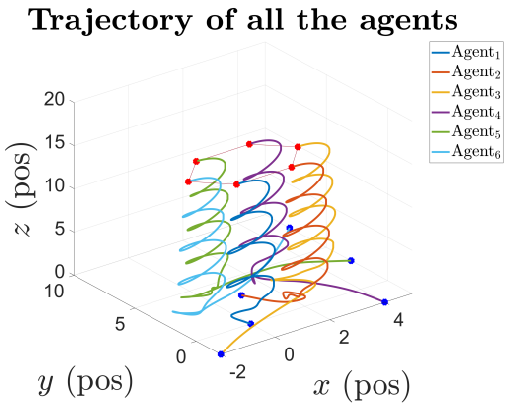
(c) Classical formation control with all the agents are faulty (Simulation A2)



(d) Time-triggered strategy with all the agents are faulty (Simulation B2)



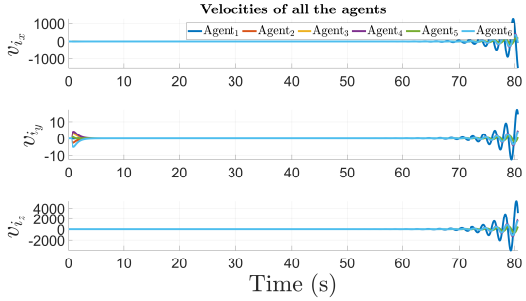
(e) Event-triggered strategy with one faulty agent (Simulation C1)



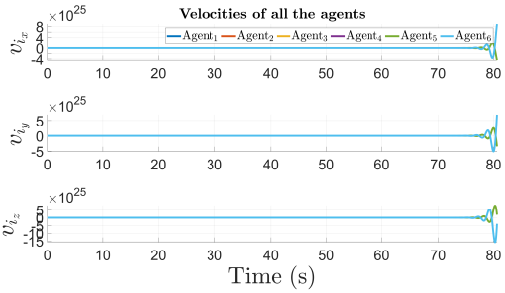
(f) Event-triggered strategy with all the agents are faulty (Simulation C2)

Fig. 4.4: Comparison of the classical formation control, time-, and event-triggered strategies.

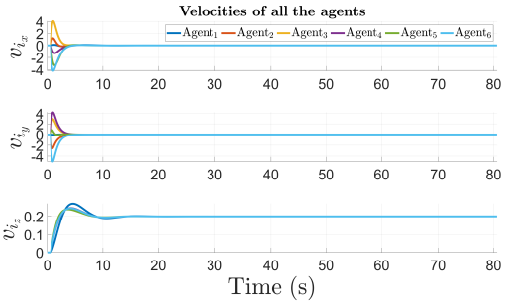
Fig. 4.5 shows a comparison of the velocities between the simulations. In Figs. 4.5(a) and 4.5(c), the velocities become unstable due to the communication faults. In Figs. 4.5(b), 4.5(d), and 4.5(e) the agents follow the velocity of the leader agent. Fig. 4.5(f) presents oscillations when the event-triggered strategy is used.



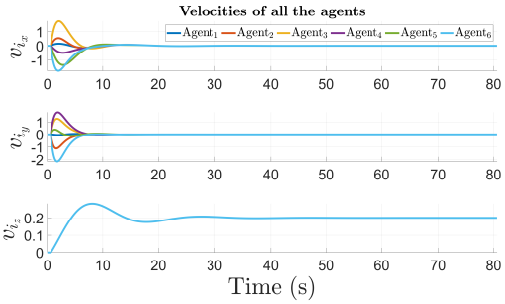
(a) Classical formation control with one faulty agent (Simulation A1)



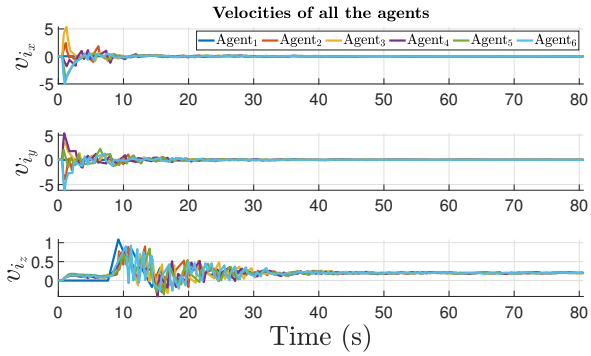
(b) Time-triggered strategy with one faulty agent (Simulation B1)



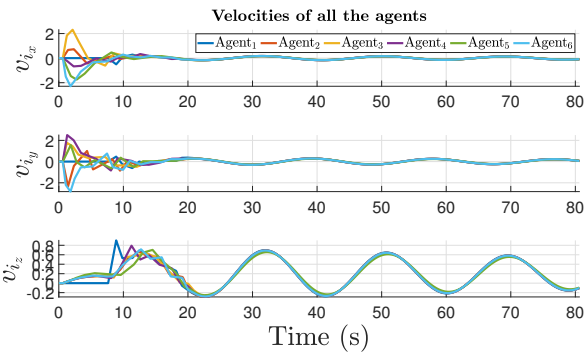
(c) Classical formation control with all the agents are faulty (Simulation A2)



(d) Time-triggered strategy with all the agents are faulty (Simulation B2)



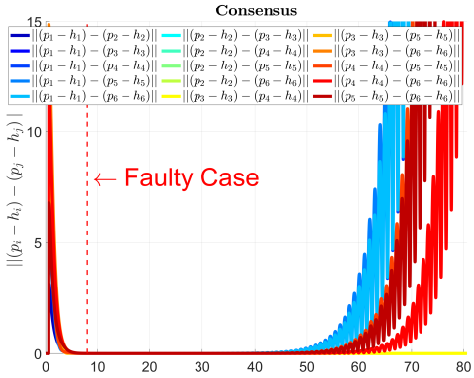
(e) Event-triggered strategy with one faulty agent (Simulation C1)



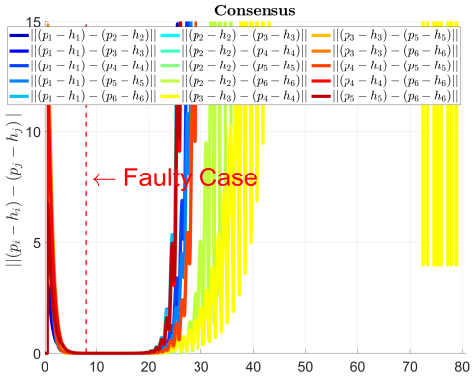
(f) Event-triggered strategy with all the agents are faulty (Simulation C2)

Fig. 4.5: Comparison of the velocities in the classical formation control, time-, and event-triggered strategies.

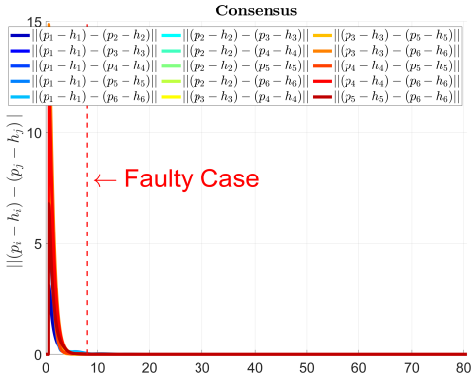
Figs. 4.6 and 4.7 illustrate the performance of the desired formation and the performance between the velocity of each agent $\|v_i(t) - v_r(t)\|$. When the communication fault occurs, in Simulation A1 and A2, corresponding to Figs. 4.6(a) and 4.6(c), the agents lose the desired formation and the agent velocities become unstable. In Simulation B1, B2, C1, and C2, corresponding to Figs. 4.6(d), 4.6(d), 4.6(e), and 4.6(f), the agents maintain the formation and follow the target velocity.



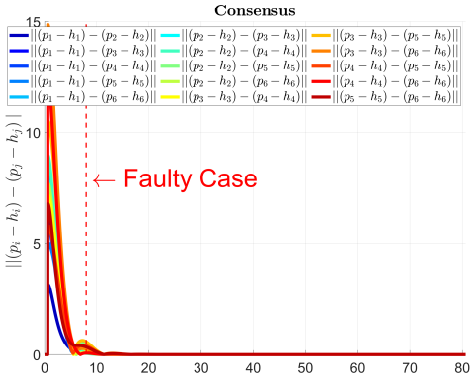
(a) Classical formation control with one faulty agent (Simulation A1)



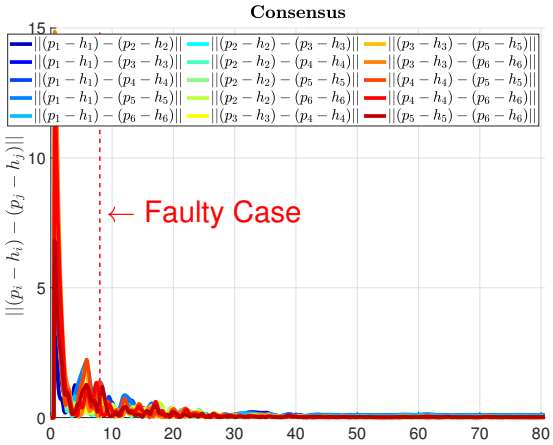
(b) Time-triggered strategy with one faulty agent (Simulation B1)



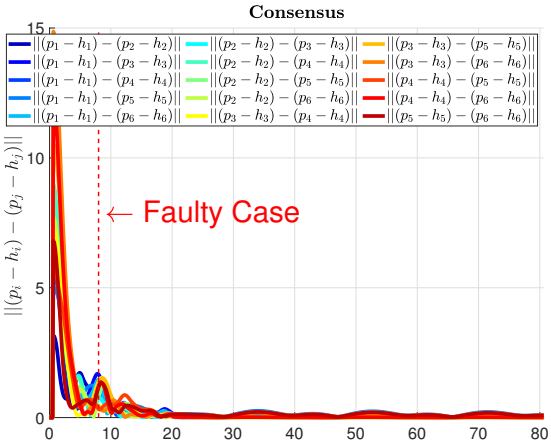
(c) Classical formation control with all the agents are faulty (Simulation A2)



(d) Time-triggered strategy with all the agents are faulty (Simulation B2)

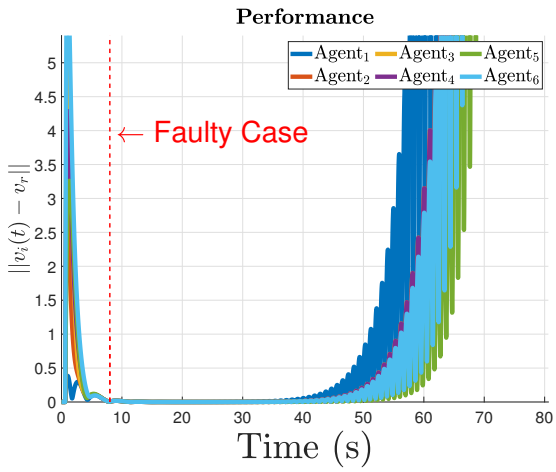


(e) Event-triggered strategy with one faulty agent (Simulation C1)

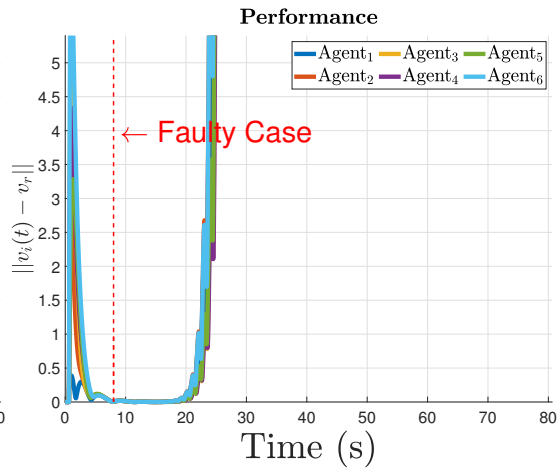


(f) Event-triggered strategy with all the agents are faulty (Simulation C2)

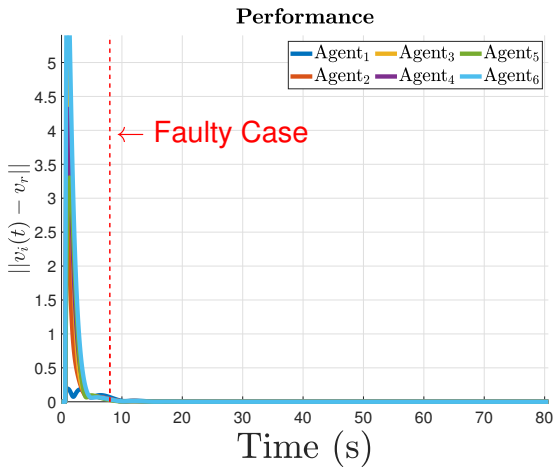
Fig. 4.6: Comparison of the desired formation performance.



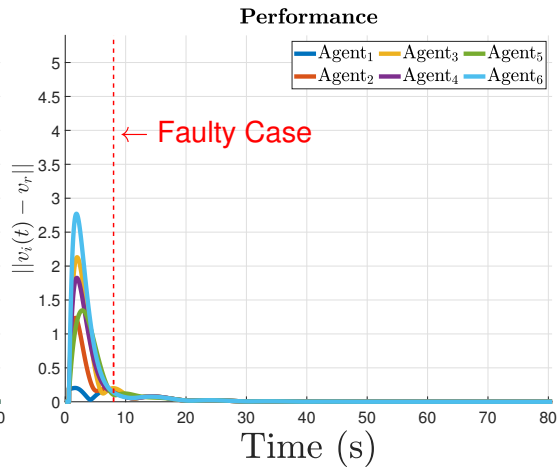
(a) Classical formation control with one faulty agent (Simulation A1)



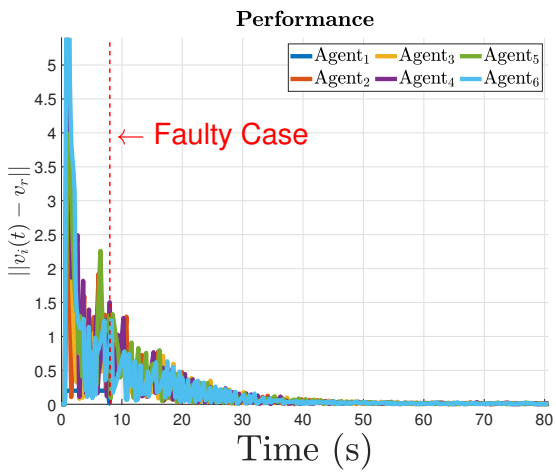
(b) Time-triggered strategy with one faulty agent (Simulation B1)



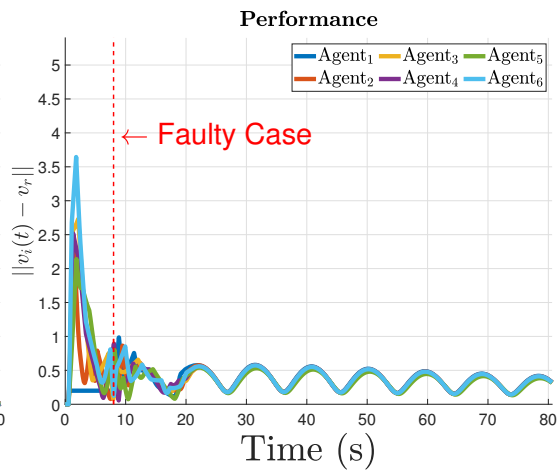
(c) Classical formation control with all the agents are faulty (Simulation A2)



(d) Time-triggered strategy with all the agents are faulty (Simulation B2)



(e) Event-triggered strategy with one faulty agent (Simulation C1)



(f) Event-triggered strategy with all the agents are faulty (Simulation C2)

Fig. 4.7: Comparison of the agents' performance following the leader agent.

In the following section, an experimental application of the proposed strategy in a fleet of UAVs under communication faults is presented.

4.5 Fleet of unmanned aerial vehicles under communication faults

The experimental platform is described and some experimental results are shown. A video corresponding to the results can be found at the following link `https://youtu.be/Lo_kuGY9Wq4`.

4.5.1 Experimental platform description

The experimental platform used for this implementation consists of: an Optitrack system to recognize the UAVs in a three-dimensional space by image processing using cameras Prime 17W; motive 2.1.1 is the software to manipulate the Optitrack which uses VRPN protocol with communication to a virtual machine; Ubuntu 16.04 is installed in the virtual machine with ROS Kinetic to manipulate the UAVs in parallel with Motive; identical Bebop 2 parrot are the UAVs (see Fig. 4.8). The code is developed in Python 2.7. The sample time is 0.02s.

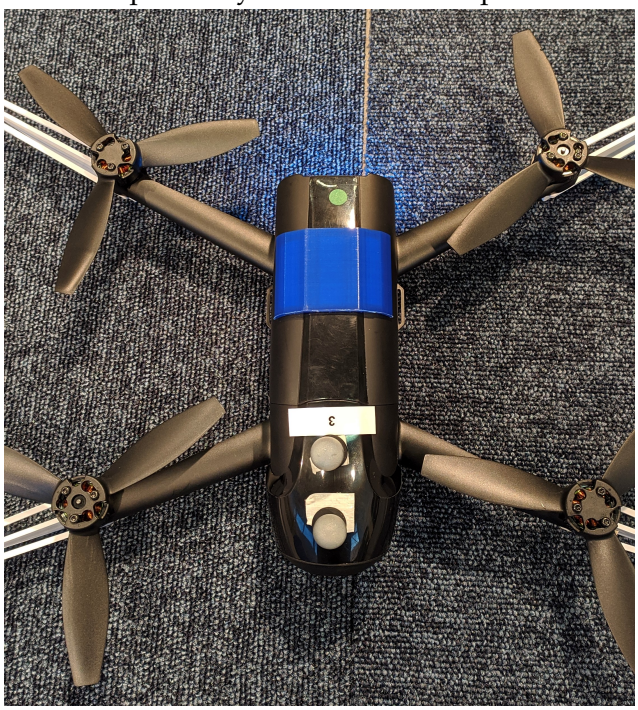


Fig. 4.8: Bebop 2 parrot.

According to [28], a fleet of UAVs can be described as a second-order multi-agent system if an inner closed-loop control is considered for each UAV employing their angles with the angle references in Appendix 3. The mass of each UAV is m_s=0.5Kg, which are considered to be homogeneous, and $g = 9.806 m/s^2$ is the acceleration of gravity. The goal of the 3 UAVs is to form an isosceles triangle and follow the trajectories of a virtual agent with the following desired formation $h_1 = [0, 0]^T$, $h_2 = [0, 1.5]^T$, and $h_3 = [0.75, 1.3]^T$. Table 4.2 shows the initial values of the UAVs.

Table 4.2: Initial positions and velocities of the agents.

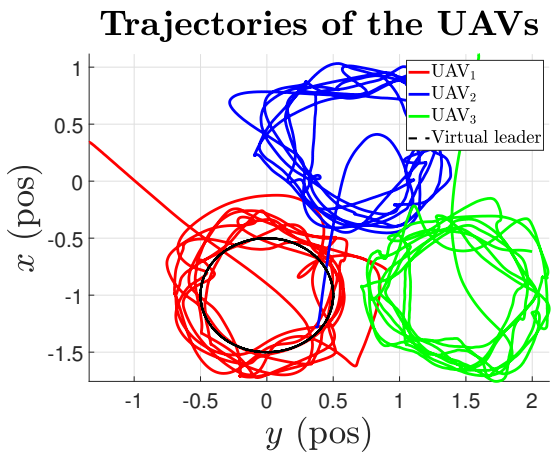
Agents' initial positions	Agents' initial velocities
$p_1(0) = [-0.7249, -0.7232]^T$	$v_1(0) = [-0.0084, 0.0357]^T$
$p_2(0) = [0.0330, -0.3345]^T$	$v_2(0) = [0.0129, -0.0107]^T$
$p_3(0) = [1.5115, 1.0167]^T$	$v_3(0) = [-0.0239, 0.0082]^T$

The communication fault is implemented through an artificial function with the following parameters: $\beta_1 = 0.8$, $\beta_2 = 1$, $\beta_3 = 0.6$, and $t_f = 10$s. All the UAVs are affected by the communication fault. The event-function has the following parameters: $c_1 = 0.03$, $c_2 = 3$, and $c_3 = 0.1$. Three implementations have been carried out for comparing the performance of the classical formation control (4.2), the time-triggered robust approach (4.7), and the event-triggered approach (4.27). In the case of the classical formation control, the experiment had to be stopped to avoid the UAVs to crash. The classical formation control uses the control gain as $K_c = \begin{bmatrix} I_3 & I_3 \end{bmatrix}$; the time- and the event-triggered strategy use the following control gain:

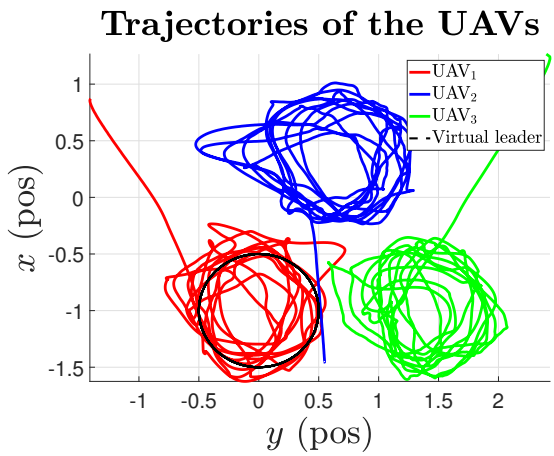
$$K_c = \begin{bmatrix} -0.2487 & 0.0000 & 0.0000 & -0.4926 & -0.0000 & -0.0000 \\ 0.0000 & -0.2487 & 0.0000 & -0.0000 & -0.4926 & -0.0000 \\ 0.0000 & 0.0000 & -0.2487 & -0.0000 & -0.0000 & -0.4926 \end{bmatrix}.$$

4.5.2 Experimental results

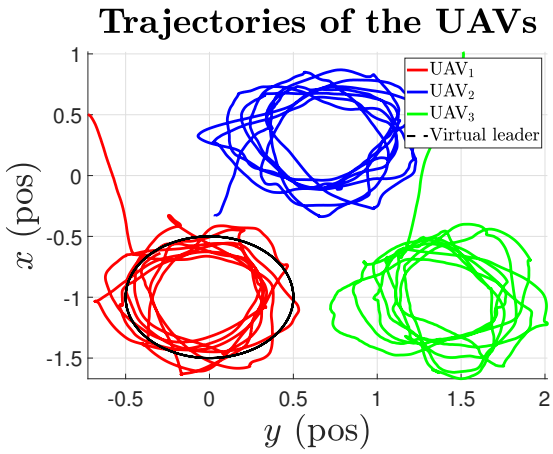
Fig. 4.9 shows a comparison of the performance of the agents' trajectories between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. Fig. 4.9(a) illustrates the obtained trajectories of the UAVs using the classical formation control. The UAVs should follow the trajectory of the virtual agent in black, however, due to the communication faults, they start to oscillate, so they cannot maintain the formation. Fig. 4.9(b) shows the UAVs trajectories obtained using the time-triggered robust control. The UAVs present a decrease in the oscillations with respect to the previous case. Moreover, they maintain the desired formation. Fig. 4.9(c) presents the UAVs trajectories when the event-triggered mechanism is used. The UAVs present a better performance, maintaining the formation despite the communication faults.



(a) Classical formation control



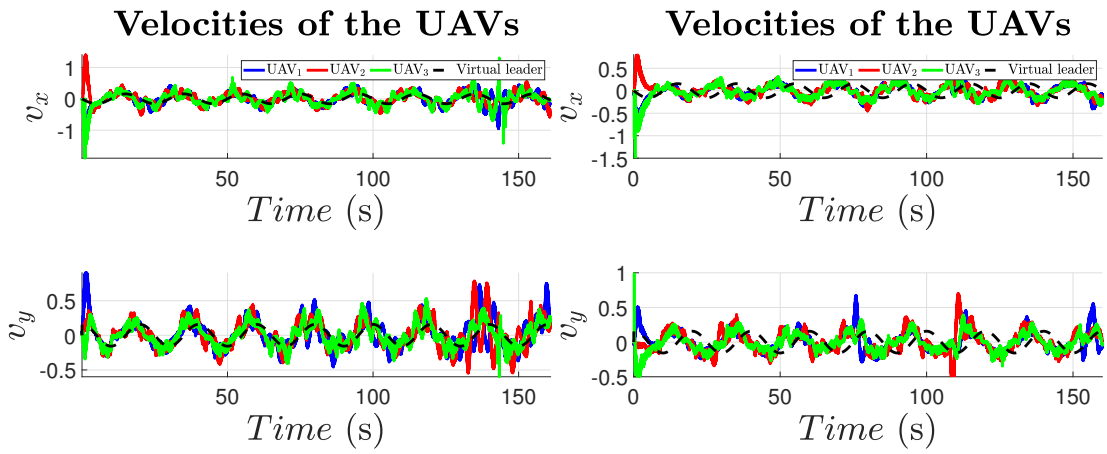
(b) Time-triggered proposed approach



(c) Event-triggered proposed strategy

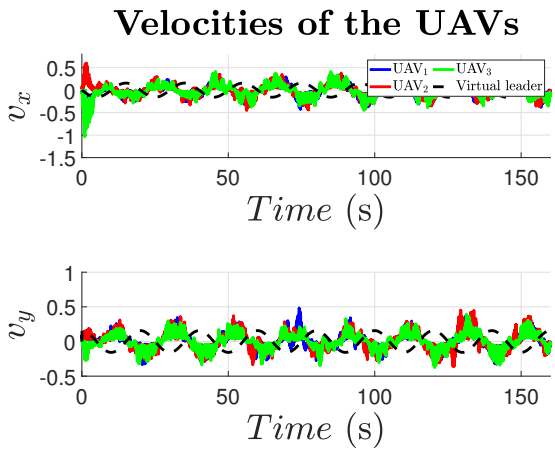
Fig. 4.9: Comparison of the trajectories' performance between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy.

Fig. 4.10 gives a comparison of the velocities' performance of the agents' velocities between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. Fig. 4.10(a) presents the UAVs velocities obtained using the classical formation control. After approximately 140s, the UAVs start to show stronger oscillations. As mentioned earlier, in order to preserve the integrity of the UAVs, the experiment had to be stopped. Fig. 4.10(b) shows the UAVs' velocities using the time-triggered robust control. The oscillations decrease when compared to the classical formation control. However, there is still a small offset between the leader velocities and the velocities of the UAVs. Also, an offset is observed induced by the fact that the control gain is smaller than in the classical formation control. Fig. 4.10(c) illustrates the UAVs' velocities using the event-triggered control. The oscillations are smaller compared to the other two approaches. However, the offset is still present due to the control gain. It is worth highlighting that the event-triggered control reduces the information exchange between the agents and the update rate of the control law.



(a) Classical formation control

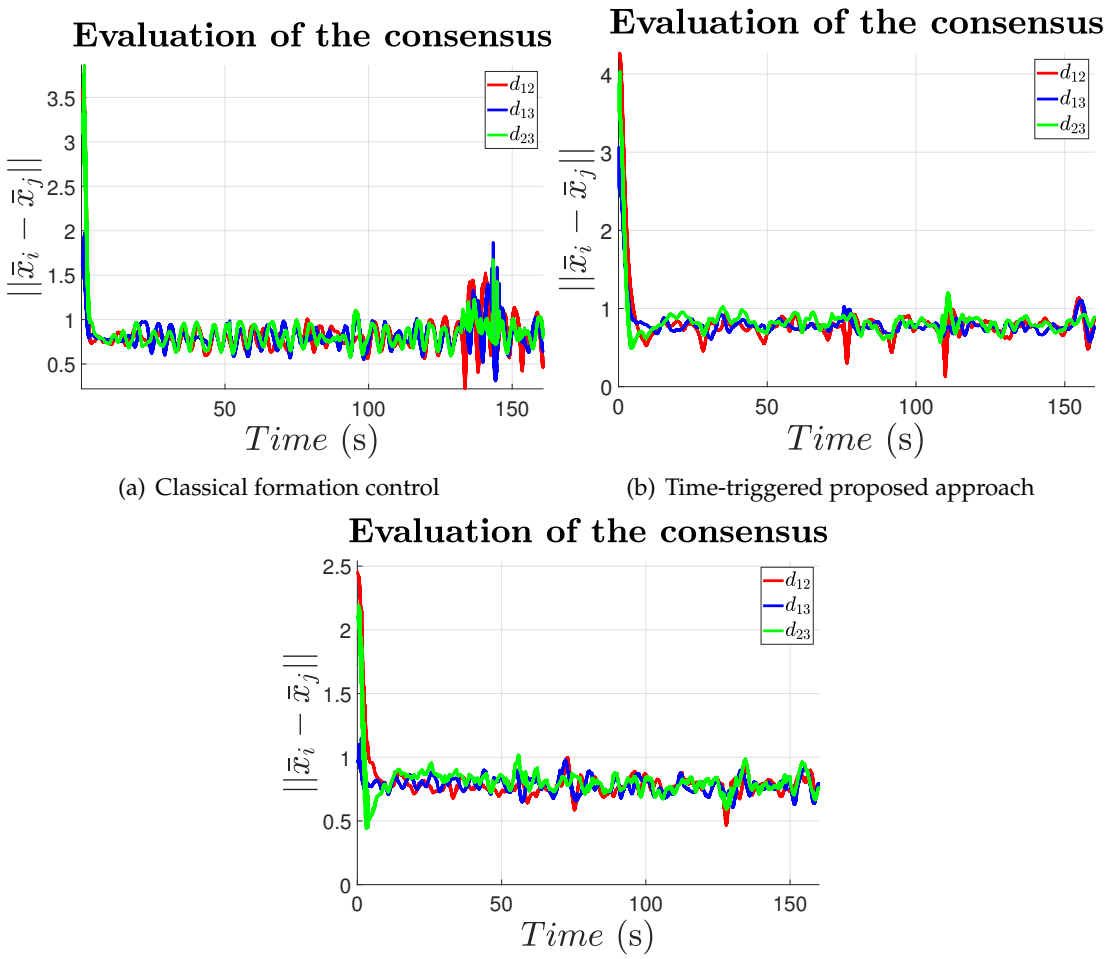
(b) Time-triggered proposed approach



(c) Event-triggered proposed strategy

Fig. 4.10: Comparison of the velocities' performance between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy.

In order to measure the performance of the consensus, let us define $d_{ij} = \|\bar{x}_i - \bar{x}_j\|$, where $\bar{x}_i = [p_i - h_i, v_i]^T$ and $\bar{x}_j = [p_j - h_j, v_j]^T$. Fig. 4.11 gives a comparison of the consensus' performance between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. Fig. 4.11(a) illustrates the evaluation of the performance of the consensus using the classical formation control. The performance presents oscillations after 140s due to the communication faults. Fig. 4.11(b) shows the evaluation of the performance of the consensus using the time-triggered robust control. Compared to Fig. 4.11(a), the performance has been improved. Fig. 4.11(c) presents the evaluation of the consensus performance using the event-triggered control. Compared to Fig. 4.11(b), the performance is smaller than the threshold value 1 due to the desired formation.



(a) Classical formation control

(b) Time-triggered proposed approach

(c) Event-triggered proposed strategy

Fig. 4.11: Comparison of the consensus' performance between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy.

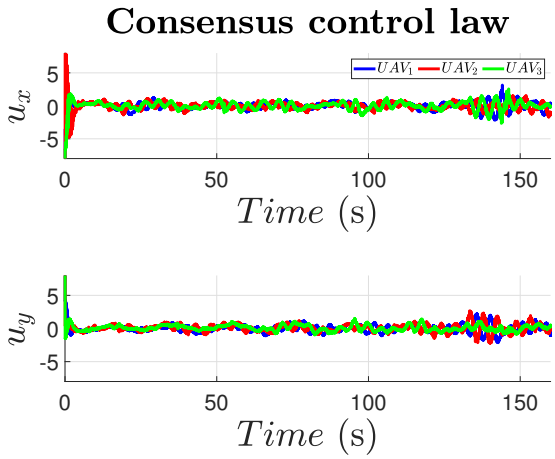
In order to quantify the performance between the approaches, root mean square (RMS) metric is used. In Table 4.3, the RMS value of d_{ij} is presented for each combination of UAVs corresponding to the classical formation control, the time-triggered robust control, and the event-triggered control.

Table 4.3: Comparison of the consensus RMS.

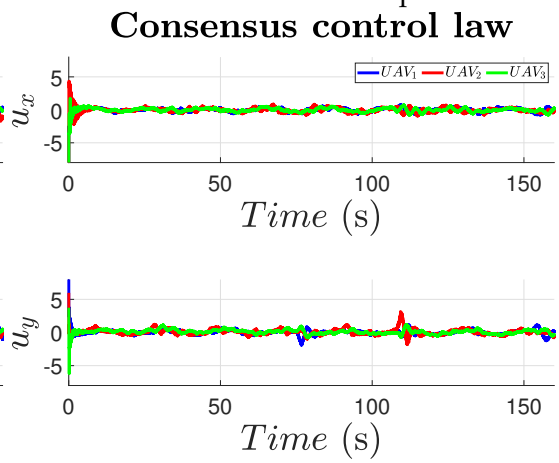
d_{ij}	Classical formation control	Time-triggered approach	Event-triggered strategy
d_{12}	0.9129	0.8947	0.8932
d_{13}	0.9060	0.8910	0.8818
d_{23}	0.9189	0.9088	0.8978

Fig. 4.12 provides a comparison of consensus control law between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy. Fig. 4.12(a) shows the consensus control law of the classical formation control which presents a greater effort to maintain the desired formation. Fig. 4.12(b) illustrates control law profile of the time-triggered approach. The control effort decrease compared to the control effort of the classical formation control law. Fig. 4.12(c) presents the event-triggered control law. The time

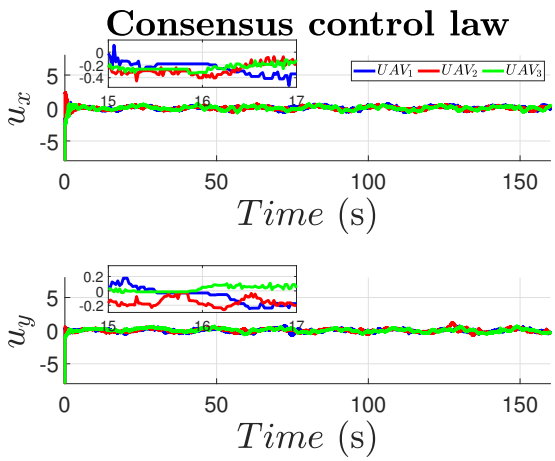
interval between 15s and 17 s is zoomed to illustrate when the control law keeps the last value.



(a) Classical formation control



(b) Time-triggered proposed approach



(c) Event-triggered proposed strategy

Fig. 4.12: Comparison of consensus control law between the classical formation control, the time-triggered proposed approach, and the event-triggered proposed strategy.

Fig. 4.13 presents the profile of the events for the event-triggered control. It is considered 1 for the UAV one, 2 for the UAV two, 3 for the UAV three if an event occurs, respectively, and 0 if there is not an event. A zoom is considered in some intervals in order to show when an event occurs.

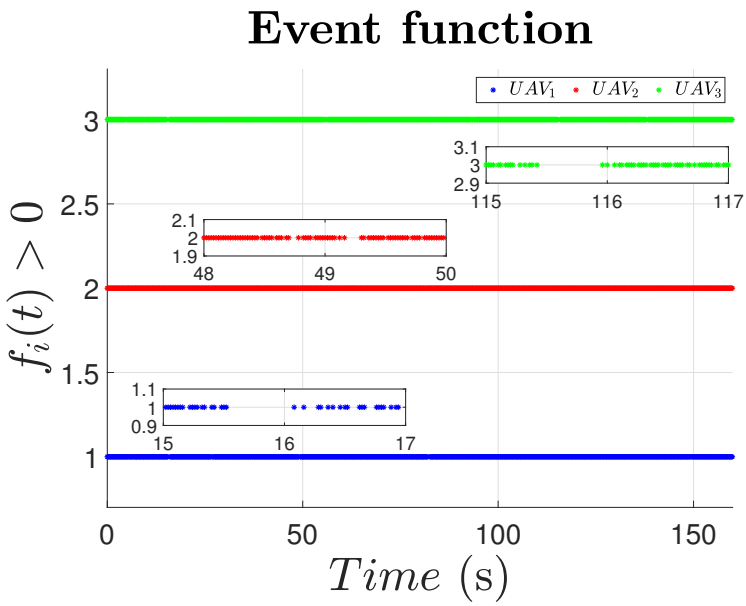


Fig. 4.13: Event-triggered profile of the agents.

Fig. 4.14 shows the total number of events in each UAV. "No event" means that the control law and the exchange of information are not updated. For example, UAV_1 has 1907 of no events compare with 6556 events. In contrast with time-triggered, the update of the information and the control law has been reduced.

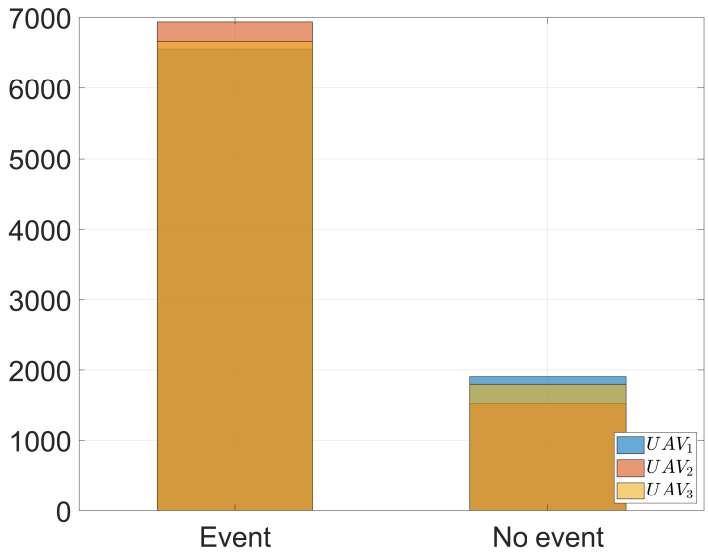


Fig. 4.14: Number of events in each UAV.

4.6 Conclusions

The design of a time- and event-triggered formation control have been presented in this Chapter. The leader-following consensus in second-order multi-agent systems has been solved in order to all the agents follow a virtual leader agent. The time-triggered formation control

has shown robustness against communication faults and the event-triggered formation control has improved the performance when the information exchange and the control law update rate are reduced. Simulation and experimental results have been used in order to show the effectiveness of the proposed strategy in a experimental platform of a fleet of UAVs. It should be noted that the type of communication fault is considered as a delay dependent on the distance between the agents, whose derivative is bounded. It would be interesting for future work to consider stochastic delays in communication faults and external disturbances.

Chapter 5

Quadratic boundedness leader-following control for multi-agent systems: Passive FTC approach

5.1 Introduction

Disturbances/uncertainties/faults affect the performance of the control systems in real applications. Rejection of these is one of the objectives in controller design [22]. As discussed in subsection 2.3.2, multi-agent systems are also subject to these type of undesired inputs which disturb the performance of the consensus [23]. To achieve a consensus in the context of multi-agent systems means that the agents are synchronized with respect to their states. If the performance of the consensus is affected by disturbances, uncertainties, or faults the tasks of synchronization are not well executed. Such tasks are described in subsection 2.2.1 for mobile agents as rendezvous, formation control, and flocking. The agents cannot reach the desired point in the case of rendezvous, in formation control the final shape is not the desired formation, and in flocking the agents can collide.

In order to reject external disturbances in multi-agent systems, $\mathscr{H}_\infty$ optimization problem has been classically studied from the robust control theory in different works such as nonlinear systems [87], nonlinear systems with asynchronous sampling [88], second-order systems [89], discrete-time nonlinear multi-agent systems with missing measurements [90], and systems under delays and parameter uncertainties [91], among others.

Adaptive control is another control strategy found in the literature to solve these type of problems in multi-agent systems. The idea of adaptive control is to identify online the model parameters and then the control parameters are tuned in order to obtain a better performance [22]. In multi-agent systems, adaptive control strategies have been investigated with actuator failures [92], state constraints [94], disturbance-compensation based on learning control [93], including deterministic disturbances [95], or bounded external disturbances [96], among others.

An alternative found in the control theory to solve bounded exogenous inputs in linear time-invariant systems is the quadratic boundedness. It is said that a system is quadratically bounded if all its solutions are bounded and this behavior can be guaranteed with a quadratic

Lyapunov function [97]. Nevertheless, as explained in subsection 2.3.2, the concept of quadratic boundedness consensus has not been considered in the literature in order to solve such problem in multi-agent systems. The quadratic boundedness consensus can be applied to design a robust controller and observer in multi-agent systems affected by bounded disturbances so that both the synchronization and the estimation error are ultimately inside an invariant ellipsoid.

As mentioned previously, one of the important issues in fault-tolerant control systems is to maintain the system performance close to the desirable one and to preserve stability conditions after occurrence of a fault [120]. Fault-tolerant control systems can be classified into two types: passive and active. Passive controllers are fixed and designed to be robust against faults, and they are very restrictive because all the expected faults cannot be known *a priori* [119]. Active controllers react actively to the faults by reconfiguring the controller so that the stability and an acceptable performance of the system can be ensured [120]. The quadratic boundedness consensus is robust against bounded inputs then the quadratic boundedness consensus can be applied to design a passive fault-tolerant control in multi-agent systems without requiring a fault detection and isolation block.

According to the state-of-the-art presented in Chapter 2, the quadratic boundedness consensus has not been found in the literature to solve such the problem of multi-agent systems subject to disturbance/actuator faults; only $\mathscr{H}_\infty$ performance or adaptive controllers can be found to solve this problem. The main contribution of this Chapter lies in developing a quadratic boundedness leader-following consensus protocol for multi-agent systems subject to exogenous bounded disturbances/faults. The advantage of the proposed technique implies that the leader-following consensus is reached in a positively invariant and attractive ellipsoid. By using a quadratic Lyapunov function, all the agents' state trajectories converge to a neighborhood of the leader's state. Linear matrix inequalities (LMIs) are obtained to synthesize the controller and observer gains such that all the agents' trajectories follow the virtual leader in the presence of bounded disturbances. The proposed approach is illustrated by solving the formation control problem in a fleet of unmanned aerial vehicles (UAVs) subject to wind turbulence and synchronizing a team of fourth-order lateral F-8 aircraft under bounded faults.

Fig. 5.1 shows a scheme of the problem under consideration in red color with the external bounded input and the solution in green color.

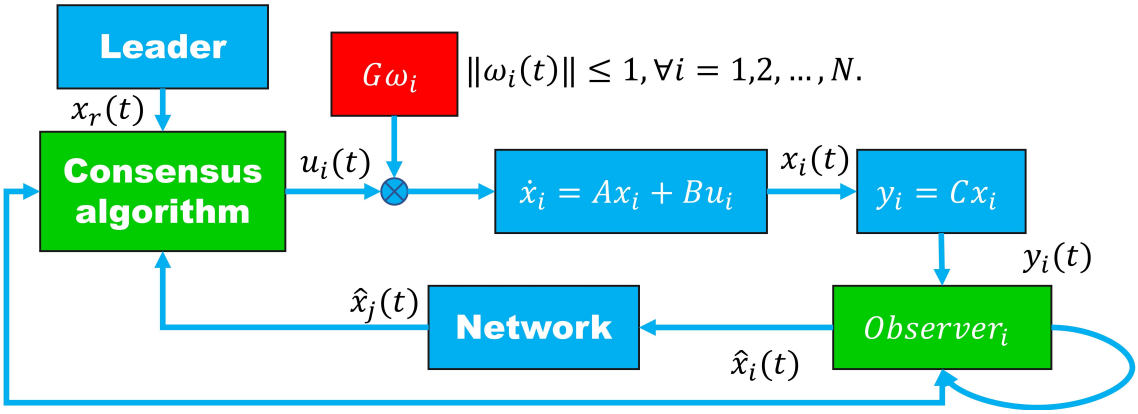


Fig. 5.1: Control scheme for an agent subject to unknown bounded inputs.

This chapter is organized as follows. The formulation problem is presented in Section 5.2. Quadratic boundedness of multi-agent systems is described in Section 5.2. In Sections 5.3 and 5.4, numerical examples considering the formation control problem and the synchronization of a fleet of fourth-order lateral F-8 aircraft under bounded faults are used to illustrate the effectiveness of the proposed strategy. Finally, the main conclusions of this Chapter are drawn in Section 5.5.

5.2 Quadratic boundedness consensus in multi-agent systems

Consider the following multi-agent system:

$$\begin{aligned} \dot{x}_i(t) &= Ax_i(t) + Bu_i(t) + G\omega_i(t), \\ y_i(t) &= Cx_i(t), \end{aligned} \tag{5.1}$$

where $x_i(t) \in \mathbb{R}^n$ is the state vector of the agent i, $u_i(t) \in \mathbb{R}^m$ is the input vector of the agent i, $y_i(t) \in \mathbb{R}^p$ is the measurement output vector of the agent i, $\omega_i(t) \in \Omega \subset \mathbb{R}^f$ represents an exogenous disturbance which is assumed to satisfy $\|\omega_i(t)\| \leq 1$ ($\forall i = 1, 2, \ldots, N$) with matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $G \in \mathbb{R}^{n\times f}$. The following standard assumptions hold for this chapter.

Assumption 12. *The pair (A, B) is stabilizable (see Appendix 4).*

Assumption 13. *The pair (A, C) is detectable (see Appendix 4).*

Assumption 14. *The graph $\mathcal{G}$ is strongly connected.*

A leader-following protocol for the multi-agent system in (5.1) is designed such that all the agents follow the reference trajectories in a quadratically bounded consensus. In this light, the dynamic of the leader agent is considered as follows:

$$\dot{x}_r(t) = Ax_r(t), \tag{5.2}$$

where $x_r(t) \in \mathbb{R}^n$ is the leader's state vector. Let us define the error between each agent i and the leader as follows:

$$\delta_i(t) = x_i(t) - x_r(t), \tag{5.3}$$

then, the error's dynamics is obtained:

$$\begin{aligned}\dot{\delta}_i(t) &= \dot{x}_i(t) - \dot{x}_r(t) \\ &= Ax_i(t) + Bu_i(t) + G\omega_i(t) - Ax_r(t) = A\delta_i(t) + Bu_i(t) + G\omega_i(t)\end{aligned} \tag{5.4}$$

According to [86], the observer-based leader-following control is given by the following equation:

$$u_i(t) = K_c \left[\sum_{j \in \mathcal{N}_i} a_{ij} \left(\hat{x}_i(t) - \hat{x}_j(t) \right) + \alpha_i \left(\hat{x}_i(t) - x_r(t) \right) \right] \tag{5.5}$$

where $K_c \in \mathbb{R}^{m \times n}$ is the control gain to be designed, α_i is the leader adjacency where $\alpha_i > 0$ if there is a directed edge from the leader to the agent i, and $\alpha_i = 0$ otherwise as reported in [86], [160], $\hat{x}_i(t)$ is the estimated state vector of the agent i, and $\hat{x}_j(t)$ is the estimated state vector corresponding to the neighboring agent j. Note that from (5.5):

$$\begin{aligned}&BK_c \sum_{j \in \mathcal{N}_i} a_{ij} \left(\hat{x}_i(t) - \hat{x}_j(t) \right) + \alpha_i BK_c \left(\hat{x}_i(t) - x_r(t) \right) \\ &= BK_c \sum_{j \in \mathcal{N}_i} a_{ij} \left(\hat{\delta}_i(t) - \hat{\delta}_j(t) \right) + \alpha_i BK_c \hat{\delta}_i(t).\end{aligned} \tag{5.6}$$

So that, after connecting the error dynamic (5.4) with the control (5.5), the following is obtained:

$$\dot{\delta}_i(t) = A\delta_i(t) + BK_c \sum_{j \in \mathcal{N}_i} a_{ij} \left(\hat{\delta}_i(t) - \hat{\delta}_j(t) \right) + \alpha_i BK_c \hat{\delta}_i(t) + G\omega_i(t). \tag{5.7}$$

Let $\delta(t) = \left[\delta_1(t)^T, \delta_2(t)^T, \ldots, \delta_N(t)^T\right]^T$, $\hat{\delta}(t) = \left[\hat{\delta}_1(t)^T, \hat{\delta}_2(t)^T, \ldots, \hat{\delta}_N(t)^T\right]^T$, $\omega(t) = \left[\omega_1(t)^T, \omega_2(t)^T, \ldots, \omega_N(t)^T\right]^T$, $\Lambda = diag\left(\alpha_1, \alpha_2, \ldots, \alpha_N\right)$, and $\bar{\mathcal{L}} = \mathcal{L} + \Lambda$. Then, the error is rewritten using the Kronecker product as follows:

$$\dot{\delta}(t) = \left(I_N \otimes A\right) \delta(t) + \left(\bar{\mathcal{L}} \otimes BK_c\right) \hat{\delta}(t) + \left(I_N \otimes G\right) \omega(t). \tag{5.8}$$

The full state vector is not always available, thus, the leader-following control uses the estimation provided by the following observer:

$$\begin{aligned}\dot{\hat{x}}_i(t) &= A\hat{x}_i(t) + Bu_i(t) + L_o \left(y_i(t) - \hat{y}_i(t) \right), \\ \hat{y}_i(t) &= C\hat{x}_i(t),\end{aligned} \tag{5.9}$$

where $\hat{x}_i(t) \in \mathbb{R}^n$ is the estimated state vector, $\hat{y}_i(t) \in \mathbb{R}^p$ is the estimated output vector, and the matrix $L_o \in \mathbb{R}^{n \times p}$ is the observer gain to be designed. Let $\hat{x}(t) = \left[\hat{x}_1(t)^T, \hat{x}_2(t)^T, \ldots, \hat{x}_N(t)^T\right]^T$ then equation (5.9) can be expressed as follows:

$$\dot{\hat{x}}(t) = \left(I_N \otimes A\right) \hat{x}(t) + \left(\bar{\mathcal{L}} \otimes BK_c\right) \hat{x}(t) + \left(I_N \otimes L_o C\right) \left(x(t) - \hat{x}(t)\right). \tag{5.10}$$

Let us define $z_1 = \delta$, $z_2 = \delta - \hat{\delta} = x - \hat{x}$, and $z = \left[z_1^T, z_2^T\right]^T$, then (5.8) and (5.10) can be rewritten as follows:

$$\dot{z} = \tilde{A}z + \tilde{G}\omega, \tag{5.11}$$

where $\tilde{A} = \begin{bmatrix} I_N \otimes A + \bar{\mathcal{L}} \otimes BK_c & -\bar{\mathcal{L}} \otimes BK_c \\ 0 & I_N \otimes (A - L_o C) \end{bmatrix}$ and $\tilde{G} = \begin{bmatrix} I_N \otimes G \\ I_N \otimes G \end{bmatrix}$.

The following definition provides a description of the quadratic boundedness concept for the closed-loop system (5.11).

Definition 3. *The system (5.11) is said to be quadratically bounded with Lyapunov matrix $\tilde{P} = \tilde{P}^T > 0$, if $z^T\tilde{P}z > 1$ implies that $z^T\tilde{P}\left(\tilde{A}z + \tilde{G}\omega\right) < 0, \forall\omega \in \Omega$.*

Remark 4. *If the system (5.11) is quadratically bounded with Lyapunov matrix $\tilde{P}$, then the set $\varepsilon = \left\{z \in \mathbb{R}^{2nN} : z^T\tilde{P}z \leq 1\right\}$ is positively invariant and attractive, that is, the set ε contains the synchronization and estimation errors.*

In order to guarantee the quadratically bounded leader-following consensus, the following theorem presents sufficient conditions.

Theorem 5. *For given eigenvalues $\lambda_j(\bar{\mathcal{L}}), \forall j = 1, 2, \ldots, N$, the system (5.11) is quadratically bounded if there exist symmetric matrices $P_1 > 0$, $P_2 > 0$, and a scalar $\beta > 0$ such that the following inequality holds*

$$\begin{bmatrix} \mathrm{He}\left\{P_1\left(A + \lambda_j P_1 BK_c\right)\right\} + \beta P_1 & -\lambda_j P_1 BK_c & P_1 G \\ * & \mathrm{He}\left\{P_2\left(A - L_o C\right)\right\} + \beta P_2 & P_2 G \\ * & * & -\beta I \end{bmatrix} \leq 0. \tag{5.12}$$

Proof. Let us calculate the time derivative of the Lyapunov function $V = z^T\tilde{P}z$ along any solution of the system (5.11) given by:

$$\dot{V}_4 = 2z^T\tilde{P}\dot{z} = 2z^T\tilde{P}\left(\tilde{A}z + \tilde{G}\omega\right), \tag{5.13}$$

where $\tilde{P} = diag\left(I_N \otimes P_1, I_N \otimes P_2\right)$, $P_1 > 0$, and $P_2 > 0$ are symmetric matrices. From (5.13), it is obtained:

$$\begin{aligned} \dot{V}_4 =& 2z_1^T\left(I_N \otimes P_1 A + \bar{\mathcal{L}} \otimes P_1 BK_c\right) z_1 - 2z_1^T\left(\bar{\mathcal{L}} \otimes P_1 BK_c\right) z_2 \\ &+ 2z_1^T\left(I_N \otimes P_1 G\right)\omega + 2z_2^T\left(I_N \otimes P_2(A - L_o C)\right) z_2 + z_2^T\left(I_N \otimes P_2 G\right)\omega. \end{aligned} \tag{5.14}$$

Let us perform a spectral decomposition of the matrix $\bar{\mathcal{L}}$ as reported in [136], such that $\bar{\mathcal{L}} = TJT^{-1}$ with an invertible matrix $T \in \mathbb{R}^{N\times N}$ and a diagonal matrix $J = diag\left(\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_N\right) \in \mathbb{R}^{N\times N}$. Let us define the following change of coordinates:

$$\begin{aligned} \varphi_1 &= \left(T^{-1} \otimes I_N\right) z_1, \\ \varphi_2 &= \left(T^{-1} \otimes I_N\right) z_2, \\ \eta &= \left(T^{-1} \otimes I_N\right)\omega. \end{aligned} \tag{5.15}$$

Replacing (5.15) in (5.14) leads to:

$$\begin{aligned} \dot{V}_4 =& 2\varphi_1^T\left(I_N \otimes P_1 A + J \otimes P_1 BK_c\right)\varphi_1 - 2\varphi_1^T\left(J \otimes P_1 BK_c\right)\varphi_2 \\ &+ 2\varphi_1^T\left(I_N \otimes P_1 G\right)\eta + 2\varphi_2^T\left(I_N \otimes P_2(A - L_o C)\right)\varphi_2 + \varphi_2^T\left(I_N \otimes P_2 G\right)\eta. \end{aligned} \tag{5.16}$$

By Lemma 3 all the eigenvalues λ_j are positive, then (5.16) can be rewritten as follows:

$$\begin{aligned} \dot{V}_4 =& \sum_{j=1}^{N}\varphi_{1_j}^T \mathrm{He}\left\{P_1 A + \lambda_j P_1 BK_c\right\}\varphi_{1_j} - 2\sum_{j=1}^{N}\varphi_{1_j}^T\lambda_j P_1 BK_c\varphi_{2_j} \\ &+ 2\sum_{j=1}^{N}\varphi_{1_j}^T P_1 G\eta_j + \sum_{j=1}^{N}\varphi_{2_j}^T \mathrm{He}\left\{P_2(A - L_o C)\right\}\varphi_{2_j} + 2\sum_{j=1}^{N}\varphi_{2_j}^T P_2 G\eta_j. \end{aligned} \tag{5.17}$$

Let $\varphi_j = \left[\varphi_{1_j}^T, \varphi_{2_j}^T\right]^T$, then (5.17) is rewritten as follows:

$$\dot{V}_4 = \sum_{j=1}^{N} \varphi_j^T \left(\mathscr{A}_j \varphi_j + 2\mathscr{P}\bar{G}\eta_j\right) \tag{5.18}$$

where $\mathscr{A}_j = \begin{bmatrix} \mathrm{He}\left\{P_1\left(A + \lambda_j P_1 B K_c\right)\right\} & -\lambda_j P_1 B K_c \\ * & \mathrm{He}\left\{P_2\left(A - L_o C\right)\right\} \end{bmatrix}$, $\mathscr{P}$ =diag(P_1, P_2), and $\bar{G} = \begin{bmatrix} G \\ G \end{bmatrix}$. From Definition 3, the following condition must hold when $V_4 > 1$:

$$\max\left\{\sum_{j=1}^{N} \varphi_j^T \left(\mathscr{A}_j \varphi_j + 2\mathscr{P}\bar{G}\eta_j\right) : \|\eta_j\| \leq 1\right\} < 0, \tag{5.19}$$

which is true if for $V_4 > 1$:

$$\sum_{j=1}^{N} \left(\varphi_j^T \mathscr{A}_j \varphi_j + 2\left\|\varphi_j \mathscr{P}\bar{G}\right\|\right) \leq 0, \tag{5.20}$$

or equivalently:

$$\sum_{j=1}^{N} \left(\varphi_j^T \mathscr{A}_j \varphi_j + 2\left(\varphi_j^T \mathscr{P}\bar{G}\bar{G}^T \mathscr{P}\varphi_j\right)^{1/2}\right) \leq 0. \tag{5.21}$$

Using Lemma 6, from (5.21) the following matrix inequality is obtained:

$$\mathscr{A}_j + \beta\mathscr{P} + \beta^{-1}\mathscr{P}\bar{G}\bar{G}^T\mathscr{P} \leq 0, \ \forall j = 1, 2, \ldots, N, \tag{5.22}$$

Using Schur complement (Lemma 5), inequality (5.22) is equivalent to (5.12), thus completing the proof. □

5.2.1 Controller and observer gains synthesis

Due to the products between decision variables, the controller and observer gains cannot be calculated using conventional tools in the inequality (5.12). The following theorem provides LMIs-based conditions in order to design the controller and observer gains.

Theorem 6. *The system (5.11) is quadratically bounded if there exist matrices $\bar{P}_1 = \bar{P}_1^T > 0$, $P_2 = P_2^T > 0$, N_1, M_o, and scalars $\beta > 0$, $\mu > 0$ such that the following inequality holds $\forall j = 1, 2, \ldots, N$:*

$$\begin{bmatrix} Q_{11_j} + \beta\bar{P}_1 & 0 & G & -\lambda_j B N_1 & 0 \\ * & Q_{22} + \beta P_2 & P_2 G & 0 & I \\ * & * & -\beta I & 0 & 0 \\ * & * & * & -\mu^{-1}\bar{P}_1 & 0 \\ * & * & * & * & -\mu\bar{P}_1 \end{bmatrix} \leq 0, \tag{5.23}$$

where $Q_{11_j} = \mathrm{He}\left\{A\bar{P}_1 + \lambda_j B N_1\right\}$, $Q_{22} = \mathrm{He}\left\{P_2 A - M_o C\right\}$, and the controller and observer gains are calculated as $K_c = N_1 \bar{P}_1^{-1}$ and $L_o = P_2^{-1} M_o$ respectively.

Proof. The matrix inequality in (5.12) is pre- and post-multiplied by *diag* $\left(P_1^{-1}, I, I\right)$, thus obtaining:

$$\begin{bmatrix} \mathscr{Q}_j + \beta\bar{X} & S\bar{G} \\ * & -\beta I \end{bmatrix} \leq 0, \tag{5.24}$$

where $\mathscr{Q}_j = \begin{bmatrix} \mathscr{Q}_{11_j} & \mathscr{Q}_{12_j} \\ * & \mathscr{Q}_{22} \end{bmatrix}$, $\mathscr{Q}_{11_j} = \text{He}\left\{A\bar{P}_1 + \lambda_j BK_c\bar{P}_1\right\}$, $\mathscr{Q}_{12_j} = -\lambda_j BK_c$, $\mathscr{Q}_{22} = \text{He}\left\{P_2\left(A - L_oC\right)\right\}$, $\bar{X}$ =diag$(\bar{P}_1, P_2)$, S =diag(I, P_2), and $\bar{P}_1 = P_1^{-1}$. Note that (5.24) can be rewritten as follows:

$$\begin{bmatrix} \begin{bmatrix} \mathscr{Q}_{11_j} & 0 \\ * & \mathscr{Q}_{22} \end{bmatrix} + \beta\bar{X} & S\bar{G} \\ * & -\beta I \end{bmatrix} + \text{He}\left\{ \begin{bmatrix} \mathscr{Q}_{12_j} \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & I & 0 \end{bmatrix} \right\} \leq 0. \tag{5.25}$$

Applying the Young relation (Lemma 4) in (5.25), the next inequality is obtained:

$$\begin{aligned} &\begin{bmatrix} \begin{bmatrix} \mathscr{Q}_{1_j} & 0 \\ * & \mathscr{Q}_2 \end{bmatrix} + \beta\bar{X} & S\bar{G} \\ * & -\beta I \end{bmatrix} + \mu \begin{bmatrix} \mathscr{Q}_{12_j} \\ 0 \\ 0 \end{bmatrix} \bar{P}_1 \begin{bmatrix} \mathscr{Q}_{12_j}^T & 0 & 0 \end{bmatrix} \\ &+ \mu^{-1} \begin{bmatrix} 0 \\ I \\ 0 \end{bmatrix} \bar{P}_1^{-1} \begin{bmatrix} 0 & I & 0 \end{bmatrix} \leq 0, \end{aligned} \tag{5.26}$$

where $\mu > 0$. Using Schur complement (Lemma 5) in (5.26) and selecting $N_1 = K_c\bar{P}_1$ and $M_o = P_2L_o$, the following inequality is obtained:

$$\begin{bmatrix} \begin{bmatrix} \begin{bmatrix} Q_{11_j} & 0 \\ * & Q_{22} \end{bmatrix} + \beta\bar{X} & S\bar{G} \\ * & -\beta I \end{bmatrix} & \begin{bmatrix} Q_{12_j} & 0 \\ 0 & I \\ 0 & 0 \end{bmatrix} \\ * & \begin{bmatrix} -\mu^{-1}\bar{P}_1 & 0 \\ * & -\mu\bar{P}_1 \end{bmatrix} \end{bmatrix} \leq 0, \tag{5.27}$$

where $Q_{11_j} = \text{He}\left\{A\bar{P}_1 + \lambda_j BN_1\right\}$, $Q_{12_j} = -\lambda_j BN_1$, and $Q_{22} = \text{He}\left\{P_2A - M_oC\right\}$. Thus, condition (5.27) is equal to (5.23), which completes the proof. □

It is worth highlighting that Theorem 5 provides sufficient conditions for the quadratic boundedness of the closed-loop system (5.11) which represents the synchronization error and the estimation error. Nevertheless, it is not possible to guarantee the quadratic boundedness of (5.11) if each gain is calculated separately because the external disturbances affect each system and observer. According to the separation principle for linear time-invariant systems reported in Appendix 5, it is possible to compute separately the controller and observer gains if the eigenvalues of the state feedback controller and the observer have the real part negative which permits one to design the observer and the full-state feedback control independently, with the assurance that the eigenvalues of the closed-loop dynamic system will be the eigenvalues selected for the full-state feedback system and the observer. Nonetheless, this is not possible when the objective has constraints such as the quadratic boundedness performance or another index performance. Theorem 5 guarantees that the whole closed-loop system is quadratic boundedness with the controller and observer.

5.3 Formation control in a fleet of UAVs subject to wind turbulence

In order to show the effectiveness of the proposed design approach, the following example presents a comparison between a classical leader-following formation control and our proposed control in a fleet of unmanned aerial vehicles (UAVs) subject to wind turbulence.

The general model of the UAV is described in Appendix 3. According to [28], a fleet of UAVs can be represented as a double integrator multi-agent system manipulating the angles of the quadrotor with the references described in Appendix 3. Wind turbulence affecting the fleet of UAVs is considered as reported in [165], [166]. Air mass motion is represented as follows:

$$\mathbf{w}_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i)}(t) = \mathbf{w}_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),0} + \sum_{s=1}^{S} a_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),s} \sin\left(\Upsilon_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),s} t + \varrho_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),s}\right) \tag{5.28}$$

where $\Upsilon_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),s}$ and $\varrho_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),s}$ are randomly selected frequencies and phase shifts in each component of each UAV, S is the number of sinusoids, $\mathbf{w}_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),0}$ is the static wind vector, and the coefficients $a_{(\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i),s}$ define the power spectral density as reported in [165]. Then, by considering the wind in each axes $\mathbf{w}_i(t) = \left[\mathbf{w}_{\mathbf{x}_i}, \mathbf{w}_{\mathbf{y}_i}, \mathbf{w}_{\mathbf{z}_i}\right]^T$, the second-order multi-agent system subject to wind turbulence is modified as follows:

$$\begin{aligned} \dot{p}_i(t) &= v_i(t), \\ \dot{v}_i(t) &= u_i(t) + \mathbf{w}_i(t). \end{aligned} \tag{5.29}$$

The leader dynamics are considered as follows:

$$\dot{p}_r(t) = v_r(t). \tag{5.30}$$

Let us define h_i and $h_j \in \mathbb{R}^{n_d}$ as the rigid desired-position formation which is considered to be constant. The error between the agent i and the virtual agent $\bar{p}_i(t) = p_i(t) - p_r(t)$, $\bar{v}_i(t) = v_i(t) - v_r(t)$, and $\bar{\delta}_i(t) = \left[\bar{p}_i(t)^T - h_i^T, \bar{v}_i(t)^T\right]^T$, thus the error dynamic (5.29) and (5.30) can be presented as follows:

$$\begin{aligned} \dot{\bar{\delta}}_i(t) &= \bar{A}\bar{\delta}_i(t) + \bar{B}u_i(t) + \bar{G}\mathbf{w}_i(t), \\ \bar{y}_i(t) &= \bar{C}\bar{\delta}_i(t) \\ \bar{A} &= \begin{bmatrix} 0_{n_d\times n_d} & I_{n_d} \\ 0_{n_d\times n_d} & 0_{n_d\times n_d} \end{bmatrix}, \ \bar{B} = \begin{bmatrix} 0_{n_d\times n_d} \\ I_{n_d} \end{bmatrix}, \ \bar{C} = \begin{bmatrix} I_{n_d} & 0_{n_d\times n_d} \end{bmatrix}, \end{aligned} \tag{5.31}$$

matrix $\bar{G}$ is chosen $\bar{G} = 0.1 \begin{bmatrix} 0_{n_d\times n_d} \\ I_{n_d} \end{bmatrix}$. Thus, the leader-following control (5.5) is modified to get a leader-following formation control as follows:

$$u_i = \bar{K}_c \sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{\bar{x}}_i - \hat{\bar{x}}_j\right) + \alpha_i \bar{K}_c\left(\hat{\bar{x}}_i - \bar{x}_r\right), \tag{5.32}$$

where $\bar{K}_c$ is the control gain to be calculated, $\hat{\bar{x}}_i = \left[\hat{\bar{p}}_i(t)^T - h_i^T, \hat{\bar{v}}_i(t)^T\right]^T$ is the estimated state vector of the second-order agent i, $\hat{\bar{x}}_j = \left[\hat{\bar{p}}_j(t)^T - h_j^T, \hat{\bar{v}}_j(t)^T\right]^T$ is the estimated state vector of the second-order agent j, and $\bar{x}_r = \left[p_r(t)^T, v_r(t)^T\right]^T$. The formation control (5.32) can be rewritten as follows:

$$u_i = \bar{K}_c\left[\sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{\bar{\delta}}_i - \hat{\bar{\delta}}_j\right)\right] + \alpha_i \bar{K}_c \hat{\bar{\delta}}_i. \tag{5.33}$$

After connecting (5.31) and (5.33), the closed-loop system can be represented as in (5.8).

The leader-following controller gain K_c is calculated offline using the LMI provided by Theorem 6 and the matrices of the system (5.31); then, the control law is computed by the protocol (5.32) using the calculated K_c online; and finally, the transformation (2) is used in order to calculate the angle reference for each UAV. In this way, it is assumed that there is an inner control for each UAV which calculates T_i, τ_{ϕ_i}, τ_{θ_i}, and τ_{ψ_i}.

The UAV parameters used in this example are: $g = 9.81$, $J_x = 21.6 \times 10^{-3}$, $J_y = 21.6 \times 10^{-3}$, $J_z = 43.2 \times 10^{-3}$, $m_s = 1.9$. The LMI in Theorem 6 is solved with the following parameters: $\mu = 0.01$, $\beta = 0.5$, and $\Lambda = diag\,(1, 0, 0, 0, 0, 0)$. The value obtained of the control gain in this section is presented as follows:

$$\bar{K}_c = \begin{bmatrix} -9.3949 & 0.0027 & 0.0027 & -18.8902 & 0.0020 & 0.0020 \\ 0.0027 & -9.3949 & 0.0027 & 0.0020 & -18.8902 & 0.0020 \\ 0.0027 & 0.0027 & -9.3949 & 0.0020 & 0.0020 & -18.8902 \end{bmatrix}.$$

The value obtained of the observer gain in this section is presented as follows:

$$\bar{L}_o = \begin{bmatrix} 24.0212 & 0.0112 & 0.0112 \\ 0.0112 & 24.0212 & 0.0112 \\ 0.0112 & 0.0112 & 24.0212 \\ 199.2804 & 0.2125 & 0.2125 \\ 0.2125 & 199.2804 & 0.2125 \\ 0.2125 & 0.2125 & 199.2804 \end{bmatrix}.$$

Six agents are considered shaping a hexagon with the parameters in Table 5.1.

Table 5.1: Initial agent position, velocities, and the desired shape configuration.

Agent	Initial position	Initial velocity	Desired positions
1	$p_1(0) = [-4, -1, 0]^T$	$v_1(0) = [0, 0, 0]^T$	$h_1 = [0, 0, 0]^T$
2	$p_2(0) = [-1, 1, 0]^T$	$v_2(0) = [0, 0, 0]^T$	$h_2 = [4, 0, 0,]^T$
3	$p_3(0) = [-2, -2, 0]^T$	$v_3(0) = [0, 0, 0]^T$	$h_3 = [6, 2\sqrt{3}, 0]^T$
4	$p_4(0) = [-2, 4, 0]^T$	$v_4(0) = [0, 0, 0]^T$	$h_4 = [4, 4\sqrt{3}, 0]^T$
5	$p_5(0) = [3, 2, 0]^T$	$v_5(0) = [0, 0, 0]^T$	$h_5 = [0, 4\sqrt{3}, 0]^T$
6	$p_6(0) = [2, 3, 0]^T$	$v_6(0) = [0, 0, 0]^T$	$h_6 = [-2, 2\sqrt{3}, 0]^T$

The target velocity is $v_r = [0, 0, 0.2]^T$. The communication topology is shown in Figure 5.2 with the following Laplacian matrix:

$$\mathcal{L} = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 2 & 0 & -1 \\ -1 & 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}.$$

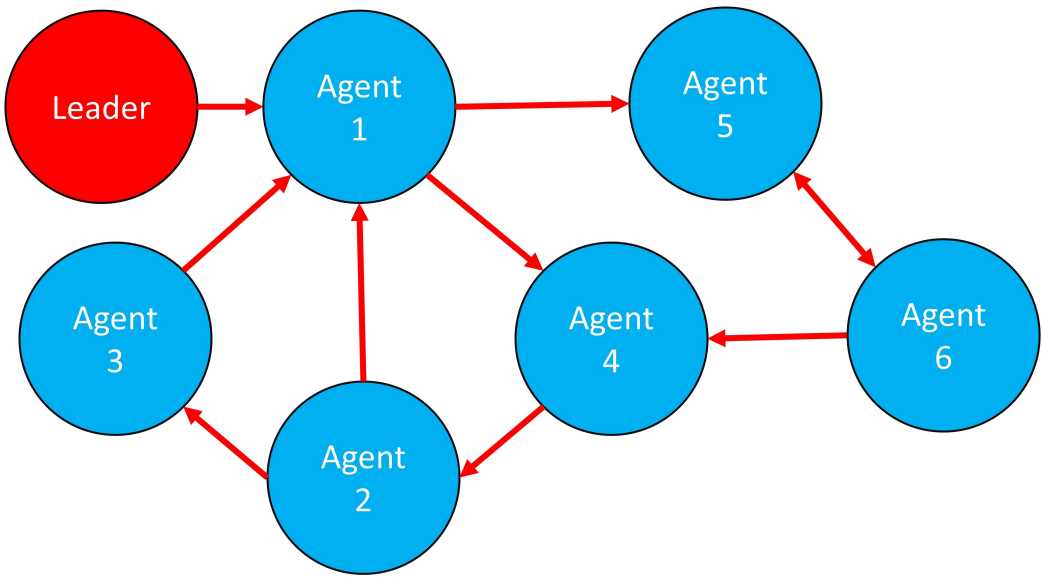
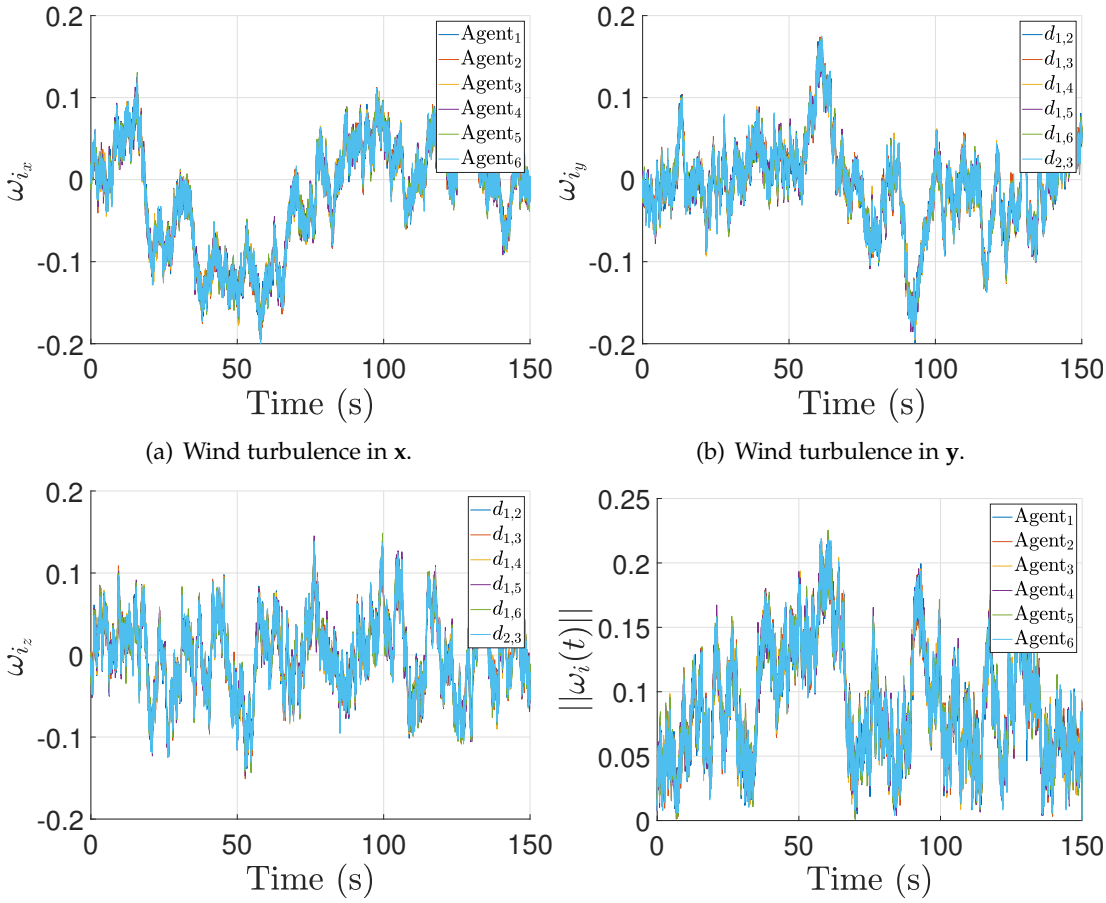


Fig. 5.2: Communication topology between agents in these examples.

A comparison of two simulations is presented in this subsection. Simulation A1 uses a formation control gain as reported in [42] ($K_c = -[I_{n_d}, I_{n_d}]$), this is:

$$u_i = -\left[\sum_{j\in\mathcal{N}_i} a_{ij}\left((p_i - p_j) - (h_i - h_j) + (v_i - v_j)\right)\right] - \alpha_i\left((p_i - p_r) + (v_i - v_r)\right).$$

Simulation B1 uses the formation control gain calculated by Theorem 6 presented in this work. In Figure 5.3, the profile of the wind turbulence which affects the fleet of the UAVs is shown (note that at each time, the wind turbulence fulfills the condition $\|\mathbf{w}_i(t)\| \leq 1$, see Fig. 5.3(d)).



(a) Wind turbulence in **x**.

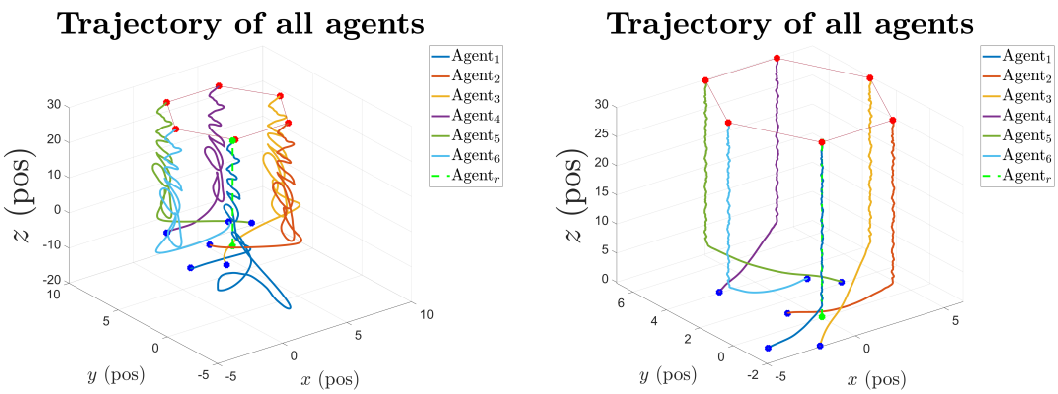
(b) Wind turbulence in **y**.

(c) Wind turbulence in **z**.

(d) Norm of the wind turbulence $\|\omega_i(t)\|$.

Fig. 5.3: Profile of the wind turbulence affecting the fleet of UAVs.

In Figure 5.4, the profile of the UAVs' displacements are shown. In Simulation A1, the agents cannot maintain the formation due to the wind. In Simulation B1, the agents maintain the formation and follow the leader agent.



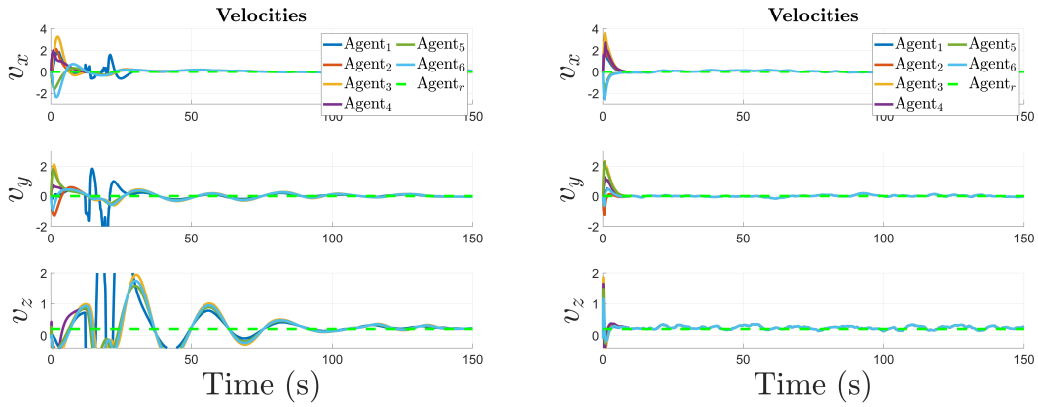
(a) Trajectories in Simulation A1.

(b) Trajectories with the proposed approach. in Simulation B1.

Fig. 5.4: Profile of the UAV trajectories.

In Figure 5.5, the profile of the UAV velocities are shown. In Simulation A1, the agents

cannot follow the velocity of the leader agent and the velocities oscillate around the target velocity. In Simulation B1, the agents follow the target velocity of the leader agent and reduce the wind effects.

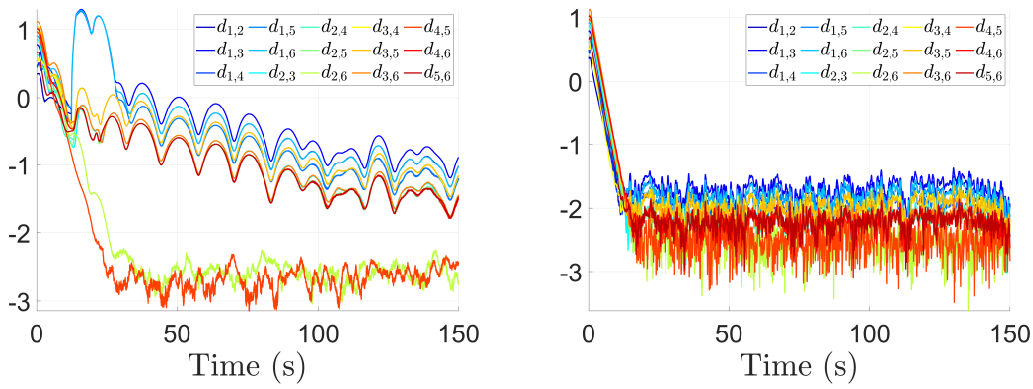


(a) Velocities in Simulation A1

(b) Velocities in Simulation B1 with the proposed approach

Fig. 5.5: Profile of the UAV velocities subject to wind turbulence.

Let us define $d_{ij} = \log_{10}\left(\left\|(p_i - p_j) - (h_i - h_j)\right\|\right)$ in order to measure the performance of the desired formation using a logarithmic scale. From Figure 5.6, it can be seen that in Simulation A1 there are synchronization errors with $d_{ij} > -1$, whereas in Simulation B1, most of the synchronization errors are under the value of -1, which indicates a more uniform formation in spite of the presence of the wind turbulence.



(a) Desired formation performance in Simulation A1

(b) Desired formation performance in Simulation B1 with the proposed approach

Fig. 5.6: Comparison of the desired formation performance.

In order to quantify the synchronization between the agents, root mean square (RMS) metric is used. In Table 5.2 and 5.3, the RMS value of d_{ij} is presented for each combination of agents corresponding to Simulation A1 and B1, respectively. The RMS is used to measure the deviations of the samples. A value of 0 in this case, indicates that the agents are well synchronized, nevertheless, in real practices, they are not due to the disturbance. Table 5.2 has shown a higher value compare to Table 5.3 where the agents use the quadratic boundedness leader-following control gain.

Table 5.2: Root mean square of the desired formation performance (Simulation A1).

$d_{1,2} = 0.5726$	$d_{1,3} = 0.687$	$d_{1,4} = 0.5823$	$d_{1,5} = 0.6205$	$d_{1,6} = 0.7064$
$d_{2,3} = 0.4894$	$d_{2,4} = 0.454$	$d_{2,5} = 0.5282$	$d_{2,6} = 0.5035$	$d_{3,4} = 0.5534$
$d_{3,5} = 0.6494$	$d_{3,6} = 0.6560$	$d_{4,5} = 0.4602$	$d_{4,6} = 0.5672$	$d_{5,6} = 0.4262$

Table 5.3: Root mean square of the desired formation performance (proposed approach Simulation B1).

$d_{1,2} = 0.2301$	$d_{1,3} = 0.3453$	$d_{1,4} = 0.2525$	$d_{1,5} = 0.3790$	$d_{1,6} = 0.3811$
$d_{2,3} = 0.3497$	$d_{2,4} = 0.2728$	$d_{2,5} = 0.4153$	$d_{2,6} = 0.3916$	$d_{3,4} = 0.2526$
$d_{3,5} = 0.4680$	$d_{3,6} = 0.4039$	$d_{4,5} = 0.4039$	$d_{4,6} = 0.4245$	$d_{5,6} = 0.2837$

The proposed strategy based on quadratic boundedness has shown robustness in reducing the effect of bounded disturbances in the fleet of UAVs. All the UAVs have followed the trajectories of a virtual leader and shaped the final figure. It is worth mentioning that the root mean square of the desired formation performance has been less in the proposed strategy.

5.4 Fourth-order lateral F-8 aircraft under bounded faults

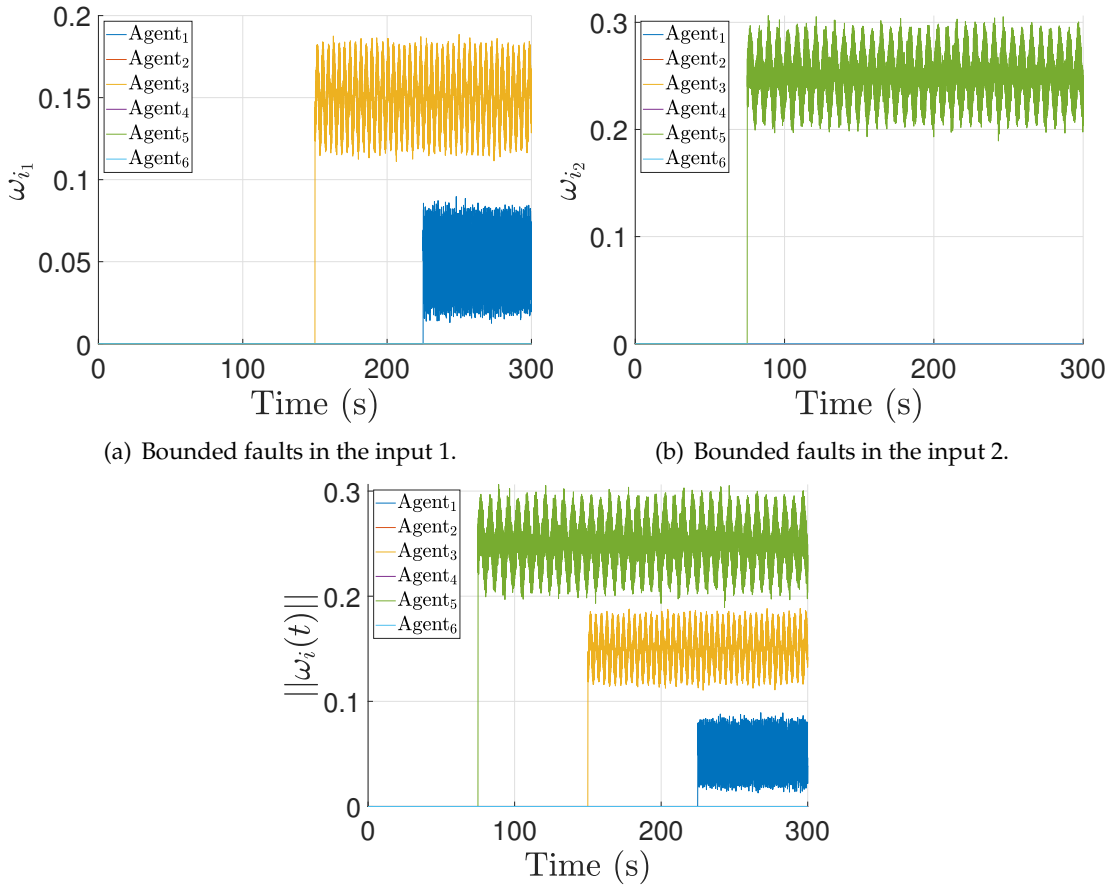
In order to illustrate the effectiveness of the proposed approach, a comparison of two simulations of a fourth order lateral F-8 aircraft [167] under additive bounded actuator faults is presented in the following example. One of the simulation (Simulation A2) uses the separation principle to synthesize the controller and observer gains, this is, the controller and observer gains are calculated independently to ensure that the consensus is stable and the estimation error separately. The separation principle can be found in Appendix 5. The second simulation (Simulation B2) uses the quadratic boundedness developed in this chapter. The linearized aircraft model has the following values:

$$A = \begin{bmatrix} -3.5980 & 0.1968 & -35.1800 & 0 \\ -0.0377 & -0.3576 & 5.8840 & 0 \\ 0.0688 & -0.9957 & -0.2163 & 0.0733 \\ 0.9947 & 0.1027 & 0 & 0 \end{bmatrix}, \; B = \begin{bmatrix} 14.6500 & 6.5380 \\ 0.2179 & -3.0870 \\ -0.0054 & 0.0516 \\ 0 & 0 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

and G is related to the inputs with B. Using Theorem 6 with the following parameters $\mu = 0.01$, $\beta = 0.1$; the computed controller and observer gains are obtained as follows:

$$K_c = \begin{bmatrix} -0.9167 & -3.3146 & 8.4646 & -0.9935 \\ 1.5769 & 6.9778 & -16.1618 & 1.7912 \end{bmatrix}, \; L_o = \begin{bmatrix} 822.6012 & -72.7985 \\ -44.6880 & 153.3311 \\ -16.4896 & 8.1629 \\ -27.2957 & 19.4179 \end{bmatrix}.$$

Fig. 5.7 shows the bounded faults considered in the F-8 model in agent 1, 3, and 5. The fault scenario is considered as follows: agent 5 with faults in actuator 2 as presented in Fig. 5.7(b) in green color starting from $t = 75$, agent 3 with faults in actuator 1 as presented in Fig. 5.7(a) in yellow color starting from $t = 150$, and agent 1 with faults in actuator 1 as presented in Fig. 5.7(a) in blue color starting from $t = 225$. Fig. 5.7(c) illustrates the norm of the bounded faults in order to show that the bounded faults satisfy that $\|\omega_i(t)\| \leq 1$ ($\forall i = 1, 2, \ldots, N$).



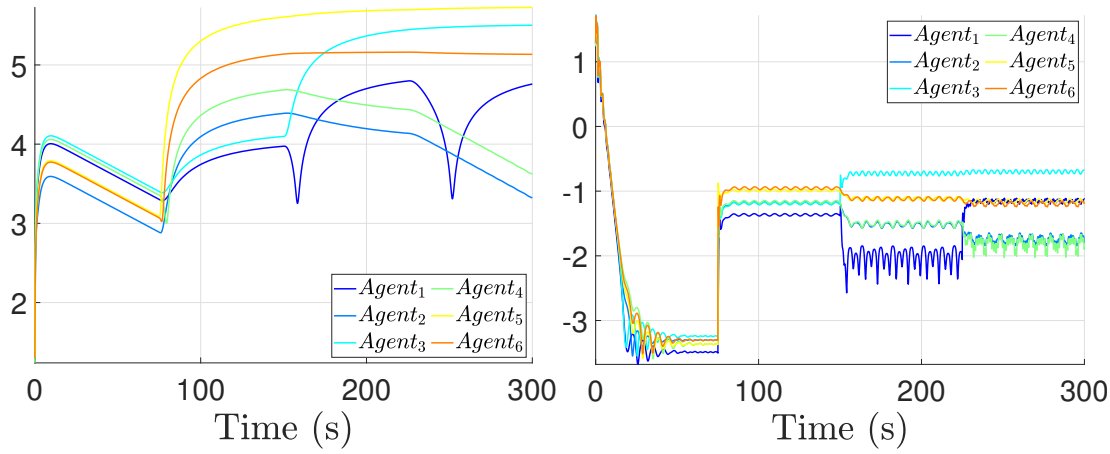
(a) Bounded faults in the input 1.

(b) Bounded faults in the input 2.

(c) Norm of the bounded fault $||\omega_i(t)||$.

Fig. 5.7: Profile of the bounded faults.

Fig. 5.8 shows a comparison of the performance of the consensus using the separation principle (Simulation A2) and the proposed quadratic boundedness approach (Simulation B2). Fig. 5.8(a) illustrates that even calculating the gains separately, it is not possible to reach a consensus. In contrast with Fig. 5.8(b), where the agents reach a consensus in spite of the bounded faults.



(a) Consensus in Simulation A2

(b) Consensus in Simulation B2 with the proposed approach

Fig. 5.8: Comparison of the consensus' performance.

In order to quantify the synchronization between the agents and the leader ($\delta_i(t)$), root mean square (RMS) metric is used. In Table 5.4 and 5.5, the RMS value of $\delta_i(t)$ is presented for each agents corresponding to Simulation A2 and B2, respectively. Simulation A2 has been shown that the separation principle cannot be used when a robust performance is required. In contrast with Simulation B2 where the quadratic boundedness technique has shown similar results than the $\mathscr{H}_\infty$ optimization developed in [168]. The quadratic boundedness approach has shown robustness against bounded external inputs.

Table 5.4: Root mean square of the consensus (Simulation A2).

$\delta_1 = 143.3907$	$\delta_2 = 102.3241$	$\delta_3 = 351.3182$
$\delta_4 = 144.9095$	$\delta_5 = 554.8304$	$\delta_6 = 303.7185$

Table 5.5: Root mean square of the consensus (proposed approach Simulation B2).

$\delta_1 = 0.4632$	$\delta_2 = 0.5193$	$\delta_3 = 0.6030$
$\delta_4 = 0.5059$	$\delta_5 = 0.5406$	$\delta_6 = 0.5535$

5.5 Conclusions

This chapter has presented a quadratically bounded leader-following protocol design for multi-agent systems. Sufficient conditions have been obtained that guarantee the design of the controller and observer gains such that all agents follow the leader agent's trajectories in a consensus in the presence of bounded disturbances. The advantage of the proposed technique implies that the leader-following consensus is reached in the sense of the quadratic boundedness, nevertheless, only under the assumption that the inputs are bounded. It is worth mentioning that in some special cases, it is possible to consider bounded actuator faults using the quadratic boundedness technique to tolerated such kind of faults. In the first example, *formation control in a fleet of UAVs subject to wind turbulence*, a linear control strategy for a fleet of UAVs, which have nonlinear dynamics, has been considered under the assumption that UAVs work inside a linear operating point obtaining a good performance compared to classic control. Therefore, it is possible that considering a nonlinear control strategy will improve the

performance. In order to reduce the exchange of information, the control update rate, and the energy consumption, event-triggered-based control design can be considered for future works as well as saturation problems due to the degradation of the actuators and more realistic faults.

General conclusions and perspectives

1 Conclusions

This thesis has presented the design of fault-tolerant methods for multi-agent systems subject to communication/actuator faults. The leader-following consensus problem is solved in order to all the agents follow the trajectory of a virtual leader agent.

In Chapter 3, the design of an observer-based leader-following consensus which has an event-triggered mechanism based on the eigenvalues of the closed-loop multi-agent system reducing the information exchange between agents and the update rate of the control law, and the design of an observer-based fault-tolerant control in multi-agent systems under actuator faults preserving an acceptable consensus performance using distributed virtual actuators for each agent. However, there are several limitations of these techniques. First of all, the event-triggered mechanism is designed based on the eigenvalues of the closed-loop system, then, as higher the initial conditions of the agents, the controller is updated more times, and it is necessary to design an effective leader-following control before applying the event-triggered mechanism. In the case of the distributed virtual actuators, the controller, observer, and the virtual actuators gains should be calculated for each fault scenario. The gains are calculated off-line and then, they are applied based on a fault detection and isolation module. A more realistic delay between the fault detection and the actuator reconfiguration must be necessary to consider in the design for each faulty agent.

A time- and event-triggered leader-following consensus for multi-agent systems in order to all the agents follow the trajectories of a virtual leader and maintain a desired formation in spite of communication faults have been developed and synthesized in Chapter 4. The proposed strategies have been evaluated in an experimental platform of a fleet of UAVs. Nevertheless, the communication faults are smooth and bounded but in most of the realistic faults, they are stochastic. The existence of a fault detection module is assumed, but currently, the fault detection in the networks is a very vast area of opportunity for research.

The design of a quadratic boundedness leader-following consensus protocol for multi-agent systems subject to exogenous bounded disturbances/faults has been proposed in Chapter 5. The leader-following consensus is reached in a positively invariant and attractive ellipsoid. The main limitation of this approach is that the exogenous input must be bounded to guarantee the quadratic boundedness and the information exchange is considered instantaneous. There are a limited number of solutions for the LMI depending on the values of the fixed scalars and the particular system.

2 Perspectives

This thesis opens several research opportunities for future works. Some of these are presented below:

- In this thesis, the problem of actuator faults, exogenous bounded inputs, and communication faults in homogeneous multi-agent systems have been considered, however, in more realistic applications, the multi-agent systems are heterogeneous. Currently, several works can be found in the context of heterogeneous multi-agent systems, however, it is a broad topic to be explored [35].
- The problem of communication faults is a problem that concerns researchers working in large-scale and interconnected systems due to its complexity. In this thesis, only deterministic faults have been considered in the information exchange, but in real applications the faults could be stochastic. The design of a robust strategy to stochastic communication faults for multi-agent systems is an interesting research opportunity [169].
- Resilient systems to cyber-attacks, the reduction of energy consumption, and the prognosis of faults are topics of broad scientific interest. Most of the current research works are focused in the design of resilient [123], event-based [83], and fault prognosis controllers in order to guarantee the efficiency and safety of the systems. As systems become increasingly complex, an alternative tool is the use of learning algorithms to save the systems, not only in multi-agent systems, but also in other systems [170].

1 Lemmas

Lemma 2 ([62]). *The Laplacian matrix $\mathcal{L}$ associated with an undirected graph $\mathcal{G}$ has at least one zero eigenvalue and all of the nonzero eigenvalues are real positive, $0 = \lambda_1(\mathcal{L}) \leq \lambda_2(\mathcal{L}) \leq \cdots \leq \lambda_N(\mathcal{L})$. The Laplacian matrix $\mathcal{L}$ has exactly one zero eigenvalue if and only if $\mathcal{G}$ is connected and the eigenvector associated with the eigenvalue zero is 1_N.*

Lemma 3 ([86]). *The matrix $\bar{\mathcal{L}}$ has nonnegative eigenvalues. The matrix $\bar{\mathcal{L}}$ is positive definite if and only if the graph $\mathcal{G}$ is connected.*

Lemma 4 (Young's relation [171]). *For given matrices A and B with appropriate dimensions, for any invertible matrix S and scalar $\mu > 0$, the following holds:*

$$\mathrm{He}\left\{A^T B\right\} \leq \mu A^T S A + \mu^{-1} B^T S^{-1} B.$$

Lemma 5 (Schur complements [172]). *For a given symmetric matrix $\begin{bmatrix} A & B \\ B^T & C \end{bmatrix} < 0$ the following statements are equivalent:*

1) $A < 0, C - B^T A^{-1} B < 0$;

2) $C < 0, A - B^T C^{-1} B < 0$.

Lemma 6 ([97]). *Suppose $A = A^T \geq 0$, $B = B^T \geq 0$, and $C = C^T > 0$. Then,*

$$x^T C x - 2\left(x^T B x\right)^{1/2} > 0, \text{ for } x^T A x > 1$$

if and only if there exists $\beta > 0$ such that:

$$C - \beta A - \beta^{-1} B \geq 0.$$

2 Notation and graph theory

Given a matrix X, X^T denotes its transpose, $X > 0 (< 0)$ denotes a symmetric positive (negative) definite matrix. The symbol $\mathbb{R}$ denotes the set of real numbers. $\|.\|$ denotes the Euclidean norm. For simplicity, the symbol $*$ within a symmetric matrix represents the symmetric entries. The Hermitian part of a square matrix X is denoted by $\mathrm{He}\{X\} := X + X^T$. The symbol $\otimes$ denotes the Kronecker product, which for real matrices A, B, C, and D with appropriate dimensions, satisfies the following properties [173]:

1) $(A + B) \otimes C = A \otimes C + B \otimes C$;

2) $(A \otimes B)^T = A^T \otimes B^T$;

3) $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$.

A directed graph $\mathcal{G}$ is a pair $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ is a non-empty finite node set and $\mathcal{E} = \{(i,j) : i, j \in \mathcal{V}\} \subseteq \mathcal{V} \times \mathcal{V}$ is an edge set of ordered pairs of N nodes (total agents). The neighbors of node i are denoted as $j \in \mathcal{N}_i$. The adjacency matrix $\mathcal{A} = \left[a_{ij}\right] \in \mathbb{R}^{N \times N}$ associated with the graph $\mathcal{G}$ is defined such that $a_{ii} = 0$, $a_{ij} > 0$ if and only if $(i,j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. The Laplacian matrix $\mathcal{L} = \left[\mathcal{L}_{ij}\right] \in \mathbb{R}^{N \times N}$ of the graph $\mathcal{G}$ is defined as $\mathcal{L}_{ii} = \sum_{j \neq i} a_{ij}$ and $\mathcal{L}_{ij} = -a_{ij}, i \neq j$.

3 Unmanned aerial vehicle general model

Let us consider the basic conceptual motions of a quadrotor shown in Figure 1.

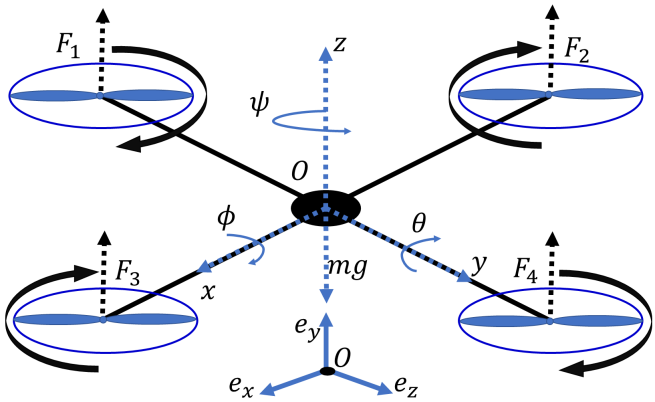


Fig. 1: Quadrotor aircraft conceptual scheme.

Four rotors produce the control actions on each agent. Each rotor produces vertical forces. The body fixed frame is assumed to be at the quadrotor gravity center. This body axis is related to the inertial frame by a position vector $(\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$ and three Euler angles $(\phi_i, \theta_i, \psi_i)$. The dynamic model can be obtained via a Lagrange approach as reported in [174], [175]. Using the force and moment balance, the motion equations can be written as follows:

$$
\begin{aligned}
\ddot{\mathbf{x}}_i &= \frac{T_i(\cos\psi_i \sin\theta_i \cos\phi_i + \sin\psi_i \sin\phi_i)}{m_s} \\
\ddot{\mathbf{y}}_i &= \frac{T_i(\sin\psi_i \sin\theta_i \cos\phi_i - \cos\psi_i \sin\phi_i)}{m_s} \\
\ddot{\mathbf{z}}_i &= \frac{T_i(\cos\theta_i \cos\phi_i) - m_s g}{m_s} \\
\ddot{\phi}_i &= \frac{\dot{\theta}_i \dot{\psi}_i (J_y - J_z) + \tau_{\phi_i}}{J_x} \\
\ddot{\theta}_i &= \frac{\dot{\phi}_i \dot{\psi}_i (J_z - J_x) + \tau_{\theta_i}}{J_y} \\
\ddot{\psi}_i &= \frac{\dot{\phi}_i \dot{\theta}_i (J_x - J_y) + \tau_{\psi_i}}{J_z}
\end{aligned}
\tag{1}
$$

where m_s is the quadrotor mass, g is the gravitational acceleration, T_i is the upward thrust force, τ_{ϕ_i} is the rolling moment, τ_{θ_i} is the pitching moment, τ_{ψ_i} is the yawing moment, J_x, J_y, J_z are the inertial of the quadrotor, Ω_r is the total sum of motor velocities, and J_r is the moment of rotational inertia around the axis of the propeller. In [176], Ω_r is considered as a disturbance because of the absence of sensors in the motors, thus, it is ignored for the design of controllers. Double integrator models are used to describe the agent dynamics when accelerations rather than velocities are directly manipulated. According to [28], a fleet of UAVs can be represented as a double integrator multi-agent system manipulating the angles of the quadrotor with the

following references:

$$\begin{aligned}
\psi_{d_i} &= 0, \\
\theta_{d_i} &= \arctan\left(\frac{u_{x_i}}{u_{z_i}+g}\right), \\
\phi_{d_i} &= \arcsin\left(\frac{-u_{y_i}}{\sqrt{u_{x_i}^2+u_{y_i}^2+(u_{z_i}+g)^2}}\right), \\
T_i &= m_s\sqrt{u_{x_i}^2+u_{y_i}^2+(u_{z_i}+g)^2},
\end{aligned} \tag{2}$$

where $u_i(t) = \left[u_{x_i}(t), u_{y_i}(t), u_{z_i}(t)\right]^T$ are to be designed based on some feedback mechanism as the consensus algorithm.

4 Controllability, observability, stabilizability, and detectability

The definitions of controllability, stabilizability, observability, and detectability are provided in this Appendix for the following system:

$$\begin{aligned}
\dot{x}(t) &= Ax(t) + Bu(t), \\
y(t) &= Cx(t) + Du(t),
\end{aligned} \tag{3}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the input vector, and $y(t) \in \mathbb{R}^p$ is the output vector. The matrices A, B, C, and D are coefficients of appropriate dimensions. These definitions can be found in [177].

4.1 Controllability

A system is said to be controllable at time t_0 if it is possible by means of an unconstrained control vector to transfer the system from any initial state $x(t_0)$ to any other state in a finite interval of time. The following theorem gives a controllability criterion.

Theorem 7 ([178]). *For the system (3) to be completely state controllable, it is necessary and sufficient that the following matrix has full rank:*

$$\begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}. \tag{4}$$

4.2 Observability

A system is said to be observable at time t_0 if, with the systems in state $x(t_0)$, it is possible to determine this state from the observation of the output over a finite time interval. The following theorem provides a observability criterion.

Theorem 8 ([178]). *For the system (3) to be completely observable, it is necessary and sufficient that the following matrix has full rank:*

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}. \tag{5}$$

4.3 Stabilizability

For a partially controllable system, if the uncontrollable modes are stable and the unstable modes are controllable, the system is said to be stabilizable.

4.4 Detectability

For a partially observable system, if the unobservable modes are stable and the observable modes are unstable, the system is said to be detectable.

5 Separation principle

According to [179], under some assumptions the problem of designing a feedback controller can be solved by designing an observer for the state of the system, which is used into the controller. Thus, the problem can be solved by separating parts, which facilitates the design. Consider a deterministic linear time-invariant (LTI) system:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned}$$

where $x(t) \in \mathbb{R}^{n_x}$, $u(t) \in \mathbb{R}^{n_u}$, and $y(t) \in \mathbb{R}^{n_y}$ represent the state vector, the input vector, and the measurement vector, respectively. The matrices A, B, and C have appropriated dimensions. Let us design the following observer-based feedback control and observer:

$$\begin{aligned} \dot{\hat{x}}(t) &= (A - LC)\hat{x}(t) + Bu(t) + Ly(t), \\ u(t) &= -K\hat{x}(t), \end{aligned}$$

where $\hat{x}(t) \in \mathbb{R}^{n_x}$ is the estimated state vector. Let us define the estimation error $e(t) = x(t) - \hat{x}(t)$, then, the following is obtained:

$$\begin{aligned} \dot{e}(t) &= (A - LC)e(t), \\ u(t) &= -K\left(x(t) - e(t)\right). \end{aligned}$$

Thus, the closed-loop is obtained as follows:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x(t) \\ e(t) \end{bmatrix}.$$

The closed-loop dynamics are governed by the upper triangular matrix,

$$\begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix},$$

the eigenvalues are given by

$$\det\left(sI - \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix}\right) = \det\left(sI - A + BK\right)\det\left(sI - A + LC\right) = 0,$$

the closed-loop eigenvalues are the eigenvalues of the state feedback controller $A - BK$, and the eigenvalues of the observer $A - LC$. This is a statement of the separation principle, which permits one to design the observer and the full-state feedback control independently, with the assurance that the eigenvalues of the closed-loop dynamic system will be the eigenvalues selected for the full-state feedback system and the observer.

References

[1] Z. Li and Z. Duan, *Cooperative control of multi-agent systems: a consensus region approach*. CRC Press, 2014.

[2] G. Villarrubia, J. F. D. Paz, D. H. D. L. Iglesia, and J. Bajo, "Combining multi-agent systems and wireless sensor networks for monitoring crop irrigation," *Sensors*, vol. 17, no. 8, 2017, ISSN: 1424-8220.

[3] G. H. Merabet, M. Essaaidi, H. Talei, *et al.*, "Applications of multi-agent systems in smart grids: A survey," in *2014 International Conference on Multimedia Computing and Systems (ICMCS)*, 2014, pp. 1088–1094.

[4] C. G. Cena, P. F. Cardenas, R. S. Pazmino, L. Puglisi, and R. A. Santonja, "A cooperative multi-agent robotics system: Design and modelling," *Expert Systems with Applications*, vol. 40, no. 12, pp. 4737–4748, 2013, ISSN: 0957-4174.

[5] G Pantelimon, K. Tepe, R Carriveau, and S. Ahmed, "Survey of multi-agent communication strategies for information exchange and mission control of drone deployments," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 3, pp. 779–788, 2019.

[6] Y. Liu and Z. Geng, "Finite-time formation control for linear multi-agent systems: A motion planning approach," *Systems & Control Letters*, vol. 85, pp. 54–60, 2015, ISSN: 0167-6911.

[7] Y. Dong and J. Huang, "Leader-following rendezvous with connectivity preservation of single-integrator multi-agent systems," *IEEE/CAA Journal of Automatica Sinica*, vol. 1, no. 1, pp. 19–23, 2014, ISSN: 2329-9266.

[8] Y. Quan, W. Chen, Z. Wu, and L. Peng, "Distributed fault detection for second-order delayed multi-agent systems with adversaries," *IEEE Access*, vol. 5, pp. 16 478–16 483, 2017.

[9] Y. Hua, X. Dong, Q. Li, and Z. Ren, "Distributed fault-tolerant time-varying formation control for high-order linear multi-agent systems with actuator failures," *ISA Transactions*, vol. 71, no. Part 1, pp. 40 –50, 2017, Special issue on Distributed Coordination Control for Multi-Agent Systems in Engineering Applications, ISSN: 0019-0578.

[10] J. Ma, D. Sun, H. Ji, and G. Feng, "Leader-following consensus of multi-agent systems with limited data rate," *Journal of the Franklin Institute*, vol. 354, no. 1, pp. 184 –196, 2017, ISSN: 0016-0032.

[11] K. Li, C. Hua, X. You, and X. Guan, "Finite-time observer-based leader-following consensus for nonlinear multiagent systems with input delays," *IEEE Transactions on Cybernetics*, pp. 1–9, 2020, ISSN: 2168-2275.

[12] G. Russo and M. di Bernardo, "Solving the rendezvous problem for multi-agent systems using contraction theory," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, 2009, pp. 5821–5826.

[13] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424 –440, 2015, ISSN: 0005-1098.

[14] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006, ISSN: 0018-9286.

[15] Z. Hu, C. Ma, L. Zhang, A. Halme, T. Hayat, and B. Ahmad, "Formation control of impulsive networked autonomous underwater vehicles under fixed and switching topologies," *Neurocomputing*, vol. 147, pp. 291–298, 2015, Advances in Self-Organizing Maps Subtitle of the special issue: Selected Papers from the Workshop on Self-Organizing Maps 2012 (WSOM 2012), ISSN: 0925-2312.

[16] G. Lafferriere, A. Williams, J. Caughman, and J. Veerman, "Decentralized control of vehicle formations," *Systems & Control Letters*, vol. 54, no. 9, pp. 899 –910, 2005, ISSN: 0167-6911.

[17] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 25–34, 1987, ISSN: 0097-8930.

[18] J. Chen and R. J. Patton, *Robust model-based fault diagnosis for dynamic systems*. Springer Science & Business Media, 2012, vol. 3.

[19] R. M. G. Ferrari, T. Parisini, and M. M. Polycarpou, "Distributed fault diagnosis with overlapping decompositions: An adaptive approximation approach," *IEEE Transactions on Automatic Control*, vol. 54, no. 4, pp. 794–799, 2009.

[20] M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, and J. Schröder, *Diagnosis and fault-tolerant control*. Springer, 2006, vol. 2.

[21] C. Chen, K. Xie, F. L. Lewis, S. Xie, and R. Fierro, "Adaptive synchronization of multi-agent systems with resilience to communication link faults," *Automatica*, vol. 111, p. 108 636, 2020, ISSN: 0005-1098.

[22] S. Li, J. Yang, W.-H. Chen, and X. Chen, *Disturbance observer-based control: methods and applications*. CRC press, 2014.

[23] R. Mu, A. Wei, H. Li, and Z.-M. Wang, "Event-triggered leader-following consensus for multi-agent systems with external disturbances under fixed and switching topologies," English, *IET Control Theory & Applications*, vol. 14, 1486–1496(10), 11 2020, ISSN: 1751-8644.

[24] Q. Liu, Z. Wang, X. He, and D. Zhou, "A survey of event-based strategies on control and estimation," *Systems Science & Control Engineering*, vol. 2, no. 1, pp. 90–97, 2014.

[25] K. J. Aström, "Event-based control," in *Astolfi A., Marconi L. (eds) Analysis and Design of Nonlinear Control Systems*, Springer, Berlin, Heidelberg, 2008.

[26] G. S. Seyboth, D. V. Dimarogonas, and K. H. Johansson, "Event-based broadcasting for multi-agent average consensus," *Automatica*, vol. 49, no. 1, pp. 245 –252, 2013, ISSN: 0005-1098.

[27] J. A. Vazquez Trejo, D. Rotondo, M. Adam Medina, and D. Theilliol, "Observer-based event-triggered model reference control for multi-agent systems," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, pp. 421–428.

[28] J. F. Guerrero-Castellanos, A. Vega-Alonzo, N. Marchand, S. Durand, J. Linares-Flores, and G. Mino-Aguilar, "Real-time event-based formation control of a group of vtol-uavs," in *2017 3rd International Conference on Event-Based Control, Communication and Signal Processing (EBCCSP)*, 2017, pp. 1–8.

[29] J. A. Vazquez Trejo, D. Theilliol, M. Adam Medina, C. D. Garcia Beltran, and M. Witczak, "Leader-following formation control for networked multi-agent systems under communication faults/failures," in *Korbicz J., Patan K., Luzar M. (eds) Advances in Diagnostics of Processes and Systems*, Springer, Cham, 2021, pp. 45–57.

[30] M. G. Losada, *Contributions to networked and event-triggered control of linear systems*. Springer, 2016.

[31] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Phys. Rev. Lett.*, vol. 75, pp. 1226–1229, 6 1995.

[32] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative control of multi-agent systems: optimal and adaptive design approaches*. Springer Science & Business Media, 2013.

[33] W. Ren and R. W. Beard, *Distributed consensus in multi-vehicle cooperative control*, 2. Springer, 2008, vol. 27.

[34] H. Bai, M. Arcak, and J. Wen, *Cooperative control design: a systematic, passivity-based approach*. Springer Science & Business Media, 2011.

[35] M. Yadegar and N. Meskin, "Output feedback fault-tolerant control of heterogeneous multi-agent systems," *Asian Journal of Control*, 2020.

[36] Y. Fan and G. Hu, "Connectivity-preserving rendezvous of multi-agent systems with event-triggered controllers," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 234–239.

[37] C. L. Watson, V. D. Chakravarthy, and S. Biswas, "A multi-agent q-learning based rendezvous strategy for cognitive radios," in *2017 Cognitive Communications for Aerospace Applications Workshop (CCAA)*, 2017, pp. 1–6.

[38] Y. Di, D. Wei, Y. Wen-jun, and Z. Yan-sheng, "Finite time rendezvous control of multi-agent networks preserving topology connectivity with constrained energy," in *2016 Chinese Control and Decision Conference (CCDC)*, 2016, pp. 1315–1319.

[39] Z. Cheng, H. T. Zhang, and M. C. Fan, "Consensus and rendezvous predictive control for multi-agent systems with input constraints," in *Proceedings of the 33rd Chinese Control Conference*, 2014, pp. 1438–1443.

[40] J. Rodrigues, D. Figueira, C. Neves, and I. Ribeiro, "Leader-following graph-based distributed formation control," in *Proc. of Robotica 2008-8th Conference on Autonomous Robot Systems and Competitions*, vol. 95, 2008.

[41] Z. Li, "Dynamic consensus of linear multi-agent systems," English, *IET Control Theory & Applications*, vol. 5, 19–28(9), 1 2011, ISSN: 1751-8644.

[42] C. Ma and J. Zhang, "On formability of linear continuous-time multi-agent systems," *Journal of Systems Science and Complexity*, vol. 25, no. 1, pp. 13–29, 2012.

[43] M. Ille and T. Namerikawa, "Collision avoidance between multi-uav-systems considering formation control using MPC," in *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2017, pp. 651–656.

[44] X. Liu, S. S. Ge, and C. H. Goh, "Neural-network-based switching formation tracking control of multiagents with uncertainties in constrained space," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. PP, no. 99, pp. 1–10, 2017, ISSN: 2168-2216.

[45] X. X. Yang, G. Y. Tang, Y. Li, and P. D. Wang, "Optimal formation control for multi-agents systems with external disturbances," in *Proceedings of the 31st Chinese Control Conference*, 2012, pp. 2291–2295.

[46] Y. D. Zhao, D. E. Kim, H. N. Yoon, S. I. Han, and J. M. Lee, "Consensus formation control of multiple wheeled mobile robots," pp. 1081–1086, 2017.

[47] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004, ISSN: 0018-9286.

[48] W. Ren, "Consensus strategies for cooperative control of vehicle formations," *IET Control Theory & Applications*, vol. 1, no. 2, pp. 505–512, 2007.

[49] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 213–224, 2010, ISSN: 1549-8328.

[50] Q. Wang, Z. Chen, P. Liu, and Q. Hua, "Distributed multi-robot formation control in switching networks," *Neurocomputing*, 2017, ISSN: 0925-2312.

[51] J. Toner and Y. Tu, "Flocks, herds, and schools: A quantitative theory of flocking," *Phys. Rev. E*, vol. 58, pp. 4828–4858, 4 1998.

[52] N. Shimoyama, K. Sugawara, T. Mizuguchi, Y. Hayakawa, and M. Sano, "Collective motion in a system of motile elements," *Phys. Rev. Lett.*, vol. 76, pp. 3870–3873, 20 1996.

[53] H. Levine, W.-J. Rappel, and I. Cohen, "Self-organization in systems of self-propelled particles," *Phys. Rev. E*, vol. 63, p. 017 101, 1 2000.

[54] M. Jafari, H. Xu, and L. R. G. Carrillo, "Brain emotional learning-based intelligent controller for flocking of multi-agent systems," in *2017 American Control Conference (ACC)*, 2017, pp. 1996–2001.

[55] C. Bhowmick, L. Behera, A. Shukla, and H. Karki, "Flocking control of multi-agent system with leader-follower architecture using consensus based estimated flocking center," in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, 2016, pp. 166–171.

[56] W. Wu, B. Liu, and H. T. Zhang, "Model predictive flocking control for the cucker-smale multi-agent model," in *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2016, pp. 1–6.

[57] H. Zhang, B. Liu, Z. Cheng, and G. Chen, "Model predictive flocking control of the cucker-smale multi-agent model with input constraints," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 8, pp. 1265–1275, 2016.

[58] J. Zhou, H. Qian, and X. Lu, "Multi-agent flocking via generalized control algorithms: Existence and properties," in *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2016, pp. 1–6.

[59] S. Martin, A. Fazeli, A. Jadbabaie, and A. Girard, "Multi-agent flocking with random communication radius," in *2012 American Control Conference (ACC)*, 2012, pp. 3871–3876.

[60] A. Belkadi, Z. Liu, L. Ciarletta, Y. Zhang, and D. Theilliol, "Flocking control of a fleet of unmanned aerial vehicles," *Control Theory and Technology*, vol. 16, no. 2, pp. 82–92, 2018.

[61] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007, ISSN: 0018-9219.

[62] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004, ISSN: 0018-9286.

[63] V. Borkar and P. Varaiya, "Asymptotic agreement in distributed estimation," *IEEE Transactions on Automatic Control*, vol. 27, no. 3, pp. 650–655, 1982.

[64] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Transactions on Automatic Control*, 1986.

[65] A. Jadbabaie, Jie Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[66] X. S. H. Z. Zhenhua Wang Juanjuan Xu, "Consensus problem in multi-agent systems under delayed information," *Neurocomputing*, vol. 316, pp. 277 –283, 2018, ISSN: 0925-2312.

[67] J. Zhang, H. Zhang, and T. Feng, "Distributed optimal consensus control for nonlinear multiagent system with unknown dynamic," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, no. 99, pp. 1–10, 2017, ISSN: 2162-237X.

[68] X. G. Guo, J. L. Wang, F. Liao, and D. Wang, "Quantized $\mathcal{H}_\infty$ consensus of multiagent systems with quantization mismatch under switching weighted topologies," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 2, pp. 202–212, 2017.

[69] G. Jing, Y. Zheng, and L. Wang, "Consensus of multiagent systems with distance-dependent communication networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, no. 99, pp. 1–15, 2017, ISSN: 2162-237X.

[70] M. He, J. Mu, and X. Mu, "$\mathcal{H}_\infty$ Leader-following consensus of nonlinear multi-agent systems under semi-Markovian switching topologies with partially unknown transition rates," *Information Sciences*, vol. 513, pp. 168 –179, 2020, ISSN: 0020-0255.

[71] J. Jiang and Y. Jiang, "Leader-following consensus of linear time-varying multi-agent systems under fixed and switching topologies," *Automatica*, vol. 113, p. 108 804, 2020, ISSN: 0005-1098.

[72] B. Cui, C. Zhao, T. Ma, and C. Feng, "Leaderless and leader-following consensus of multi-agent chaotic systems with unknown time delays and switching topologies," *Nonlinear Analysis: Hybrid Systems*, vol. 24, pp. 115 –131, 2017, ISSN: 1751-570X.

[73] L. Ding and G. Guo, "Sampled-data leader-following consensus for nonlinear multi-agent systems with markovian switching topologies and communication delay," *Journal of the Franklin Institute*, vol. 352, no. 1, pp. 369 –383, 2015, ISSN: 0016-0032.

[74] S. A. Ajwad, T. Ménard, E. Moulay, M. Defoort, and P. Coirault, "Observer based leader-following consensus of second-order multi-agent systems with nonuniform sampled position data," *Journal of the Franklin Institute*, vol. 356, no. 16, pp. 10031 –10 057, 2019, ISSN: 0016-0032.

[75] Y. Zhang, R. Li, W. Zhao, and X. Huo, "Stochastic leader-following consensus of multi-agent systems with measurement noises and communication time-delays," *Neurocomputing*, vol. 282, pp. 136 –145, 2018, ISSN: 0925-2312.

[76] H. Ren and F. Deng, "Mean square consensus of leader-following multi-agent systems with measurement noises and time delays," *ISA Transactions*, vol. 71, pp. 76 –83, 2017, Special issue on Distributed Coordination Control for Multi-Agent Systems in Engineering Applications, ISSN: 0019-0578.

[77] A Belkadi, L Ciarletta, and D Theilliol, "Particle swarm optimization method for the control of a fleet of unmanned aerial vehicles," *Journal of Physics: Conference Series*, vol. 659, p. 012 015, 2015.

[78] M. Lu and L. Liu, "Leader-following consensus of multiple uncertain euler–lagrange systems subject to communication delays and switching networks," *IEEE Transactions on Automatic Control*, vol. 63, no. 8, pp. 2604–2611, 2017, ISSN: 2334-3303.

[79] W. He, G. Chen, Q.-L. Han, and F. Qian, "Network-based leader-following consensus of nonlinear multi-agent systems via distributed impulsive control," *Information Sciences*, vol. 380, pp. 145 –158, 2017, ISSN: 0020-0255.

[80] M.-C. Fan and Y. Wu, "Global leader-following consensus of nonlinear multi-agent systems with unknown control directions and unknown external disturbances," *Applied Mathematics and Computation*, vol. 331, pp. 274 –286, 2018, ISSN: 0096-3003.

[81] X. You, C. Hua, and X. Guan, "Event-triggered leader-following consensus for nonlinear multiagent systems subject to actuator saturation using dynamic output feedback method," *IEEE Transactions on Automatic Control*, vol. 63, no. 12, pp. 4391–4396, 2018.

[82] Z. G. Wu, Y. Xu, R. Lu, Y. Wu, and T. Huang, "Event-triggered control for consensus of multiagent systems with fixed/switching topologies," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. PP, no. 99, pp. 1–11, 2017, ISSN: 2168-2216.

[83] W. Zou and Z. Xiang, "Event-triggered leader–following consensus of non-linear multi-agent systems with switched dynamics," *IET Control Theory & Applications*, vol. 13, 1222–1228(6), 9 2019, ISSN: 1751-8644.

[84] C. Deng and G.-H. Yang, "Leaderless and leader-following consensus of linear multi-agent systems with distributed event-triggered estimators," *Journal of the Franklin Institute*, vol. 356, no. 1, pp. 309 –333, 2019, ISSN: 0016-0032.

[85] T. Xiong, Z. Pu, and J. Yi, "Time-varying formation finite-time tracking control for multi-UAV systems under jointly connected topologies," *International Journal of Intelligent Computing and Cybernetics*, 2017.

[86] W. Ni and D. Cheng, "Leader-following consensus of multi-agent systems under fixed and switching topologies," *Systems & Control Letters*, vol. 59, no. 3, pp. 209 –217, 2010, ISSN: 0167-6911.

[87] X. Jiang, G. Xia, Z. Feng, and Z. Jiang, "Non-fragile guaranteed-performance $\mathcal{H}_\infty$ leader-following consensus of lipschitz nonlinear multi-agent systems with switching topologies," *Nonlinear Analysis: Hybrid Systems*, vol. 38, p. 100 913, 2020, ISSN: 1751-570X.

[88] T. Ménard, S. A. Ajwad, E. Moulay, P. Coirault, and M. Defoort, "Leader-following consensus for multi-agent systems with nonlinear dynamics subject to additive bounded disturbances and asynchronously sampled outputs," *Automatica*, vol. 121, p. 109 176, 2020, ISSN: 0005-1098.

[89] J. Han, H. Zhang, and H. Jiang, "Event-based $\mathcal{H}_\infty$ consensus control for second-order leader-following multi-agent systems," *Journal of the Franklin Institute*, vol. 353, no. 18, pp. 5081 –5098, 2016, ISSN: 0016-0032.

[90] F. Han, Q. He, H. Gao, J. Song, and J. Li, "Decouple design of leader-following consensus control for nonlinear multi-agent systems," *Systems Science & Control Engineering*, vol. 8, no. 1, pp. 288–296, 2020.

[91] S. Liang, Z. Liu, and Z. Chen, "Leader-following $\mathcal{H}_\infty$ consensus of discrete-time nonlinear multi-agent systems based upon output feedback control," *Transactions of the Institute of Measurement and Control*, vol. 42, no. 7, pp. 1323–1333, 2020.

[92] J.-N. Li, X. Liu, X.-F. Ru, and X. Xu, "Disturbance rejection adaptive fault-tolerant constrained consensus for multi-agent systems with failures," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 12, pp. 3302–3306, 2020.

[93] Y. Wang, H. Li, X. Qiu, and X. Xie, "Consensus tracking for nonlinear multi-agent systems with unknown disturbance by using model free adaptive iterative learning control," *Applied Mathematics and Computation*, vol. 365, p. 124 701, 2020, ISSN: 0096-3003.

[94] Y. Zhang, H. Liang, H. Ma, Q. Zhou, and Z. Yu, "Distributed adaptive consensus tracking control for nonlinear multi-agent systems with state constraints," *Applied Mathematics and Computation*, vol. 326, pp. 16–32, 2018, ISSN: 0096-3003.

[95] J. Sun, Z. Geng, Y. Lv, Z. Li, and Z. Ding, "Distributed adaptive consensus disturbance rejection for multi-agent systems on directed graphs," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 629–639, 2018.

[96] Y. Liu, "Adaptive leader-following consensus control of multi-agent systems using model reference adaptive control approach," *IET Control Theory & Applications*, vol. 6, 2002–2008(6), 13 2012, ISSN: 1751-8644.

[97] M. L. Brockman and M. Corless, "Quadratic boundedness of nominally linear systems," *International Journal of Control*, vol. 71, no. 6, pp. 1105–1117, 1998.

[98] M. Buciakowski, M. Witczak, M. Mrugalski, and D. Theilliol, "A quadratic boundedness approach to robust dc motor fault estimation," *Control Engineering Practice*, vol. 66, pp. 181–194, 2017, ISSN: 0967-0661.

[99] J. Cayero, D. Rotondo, B. Morcego, and V. Puig, "Optimal state observation using quadratic boundedness: Application to uav disturbance estimation," *International Journal of Applied Mathematics and Computer Science*, vol. 29, no. 1, pp. 99 –109, 2019.

[100] B. Ding, "Quadratic boundedness via dynamic output feedback for constrained nonlinear systems in takagi–sugeno's form," *Automatica*, vol. 45, no. 9, pp. 2093–2098, 2009, ISSN: 0005-1098.

[101] J. Lunze, *Control theory of digitally networked dynamic systems*. Springer, 2014, vol. 1.

[102] H. Yu and F. Hao, "The existence of zeno behavior and its application to finite-time event-triggered control," *Science China Information Sciences*, vol. 63, no. 1, pp. 1–3, 2020.

[103] H. Yu and T. Chen, "On zeno behavior in event-triggered finite-time consensus of multi-agent systems," *IEEE Transactions on Automatic Control*, pp. 1–1, 2020.

[104] W. Li, Y. Liu, and H. Sun, "A survey of event-based consensus for multi-agent systems," in *2017 Chinese Automation Congress (CAC)*, 2017, pp. 6606–6611.

[105] J. Liu, Y. Zhang, Y. Yu, and C. Sun, "Fixed-time leader-follower consensus of networked nonlinear systems via event/self-triggered control," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–9, 2020, ISSN: 2162-2388.

[106] J. Sun, J. Liu, Y. Wang, Y. Yu, and C. Sun, "Fixed-time event-triggered synchronization of a multilayer Kuramoto-oscillator network," *Neurocomputing*, vol. 379, pp. 214 –226, 2020, ISSN: 0925-2312.

[107] D. Liu and G. Yang, "A dynamic event-triggered control approach to leader-following consensus for linear multiagent systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–9, 2020, ISSN: 2168-2232.

[108] D. Yao, H. Li, R. Lu, and Y. Shi, "Distributed sliding-mode tracking control of second-order nonlinear multiagent systems: An event-triggered approach," *IEEE Transactions on Cybernetics*, pp. 1–11, 2020, ISSN: 2168-2275.

[109] S. J. Yoo and B. S. Park, "Guaranteed-connectivity-based distributed robust event-triggered tracking of multiple underactuated surface vessels with uncertain nonlinear dynamics," *Nonlinear Dynamics*, 2020, ISSN: 1573-269X.

[110] L. Zhang, J. Sun, and Q. Yang, "Distributed model-based event-triggered leader-follower consensus control for linear continuous-time multiagent systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–9, 2020, ISSN: 2168-2232.

[111] J. Savaliya, K. Patel, and A. Mehta, "Distributed event-triggered sliding mode control for voltage synchronization of DC microgrid using leader–follower consensus protocol," in *Advances in Control Systems and its Infrastructure*, A. Mehta, A. Rawat, and P. Chauhan, Eds., Singapore: Springer Singapore, 2020, pp. 25–35, ISBN: 978-981-15-0226-2.

[112] S. Wang, Y. Cao, T. Huang, Y. Chen, and S. Wen, "Event-triggered distributed control for synchronization of multiple memristive neural networks under cyber-physical attacks," *Information Sciences*, vol. 518, pp. 361 –375, 2020, ISSN: 0020-0255.

[113] J. Yang, F. Xiao, and T. Chen, "Event-triggered formation tracking control of nonholonomic mobile robots without velocity measurements," *Automatica*, vol. 112, p. 108 671, 2020, ISSN: 0005-1098.

[114] Y. Ye and H. Su, "Leader-following consensus of general linear fractional-order multiagent systems with input delay via event-triggered control," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 18, pp. 5717–5729, 2018.

[115] X. Tan, "Leader-following mean square consensus of stochastic multi-agent systems with input delay via event-triggered control," English, *IET Control Theory and Applications*, vol. 12, 299–309(10), 2 2018, ISSN: 1751-8644.

[116] M. Duan, C. Liu, and F. Liu, "Event-triggered consensus seeking of heterogeneous first-order agents with input delay," *IEEE Access*, vol. 5, pp. 5215–5223, 2017, ISSN: 2169-3536.

[117] D. Zhao and T. Dong, "Reduced-order observer-based consensus for multi-agent systems with time delay and event trigger strategy," *IEEE Access*, vol. 5, pp. 1263–1271, 2017, ISSN: 2169-3536.

[118] G. Miao, J. Cao, A. Alsaedi, and F. E. Alsaadi, "Event-triggered containment control for multi-agent systems with constant time delays," *Journal of the Franklin Institute*, vol. 354, no. 15, pp. 6956 –6977, 2017, ISSN: 0016-0032.

[119] H. Noura, D. Theilliol, J.-C. Ponsart, and A. Chamseddine, *Fault-tolerant control systems: Design and practical applications*. Springer Science & Business Media, 2009.

[120] Z. Youmin and J. Jin, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual Reviews in Control*, vol. 32, no. 2, pp. 229 –252, 2008, ISSN: 1367-5788.

[121] M. Siavash, V. J. Majd, and M. Tahmasebi, "Fault-tolerant formation control of stochastic nonlinear multi-agent systems with time-varying weighted topology," *Transactions of the Institute of Measurement and Control*, 2020.

[122] W. Guan, T. Bian, and Z. Zhao, "Fault-tolerant control of multi-agent systems with saturation and $\mathcal{L}_2$-disturbances," *IEEE Access*, vol. 8, pp. 556–564, 2020.

[123] A. Mustafa and H. Modares, "Attack analysis and resilient control design for discrete-time distributed multi-agent systems," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 369–376, 2020.

[124] Y. Xu and Z. Wu, "Distributed adaptive event-triggered fault-tolerant synchronization for multi-agent systems," *IEEE Transactions on Industrial Electronics*, pp. 1–1, 2020.

[125] L. Zhao and G.-H. Yang, "Cooperative adaptive fault-tolerant control for multi-agent systems with deception attacks," *Journal of the Franklin Institute*, 2020, ISSN: 0016-0032.

[126] J. Liu, C. Wang, and Y. Xu, "Distributed adaptive output consensus tracking for high-order nonlinear time-varying multi-agent systems with output constraints and actuator faults," *Journal of the Franklin Institute*, vol. 357, no. 2, pp. 1090 –1117, 2020, ISSN: 0016-0032.

[127] D. Liu, Z. Liu, C. P. Chen, and Y. Zhang, "Distributed adaptive neural control for uncertain multi-agent systems with unknown actuator failures and unknown dead zones," *Nonlinear Dynamics*, vol. 99, no. 2, pp. 1001–1017, 2020.

[128] H. Sunan and F. Lin, "Model-based fault accommodation control of multi-agent systems," *Advanced Control for Applications: Engineering and Industrial Systems*, 2019.

[129] C. Chen, F. L. Lewis, S. Xie, *et al.*, "Resilient adaptive and $\mathcal{H}_\infty$ controls of multi-agent systems under sensor and actuator faults," *Automatica*, vol. 102, pp. 19 –26, 2019, ISSN: 0005-1098.

[130] X.-F. Zhao and X.-Z. Jin, "Adaptive FTC consensus control for nonlinear multi-agent systems based on neural networks," in *Proceedings of the 2019 The 2nd International Conference on Robotics, Control and Automation Engineering*, ser. RCAE 2019, Lanzhou, China: Association for Computing Machinery, 2019, pp. 73–77, ISBN: 9781450376228.

[131] C. Deng, "Distributed resilient control for cyber-physical systems under denial-of-service attacks," in *2019 23rd International Conference on Mechatronics Technology (ICMT)*, 2019, pp. 1–5.

[132] C. Deng and W. Che, "Fault-tolerant fuzzy formation control for a class of nonlinear multiagent systems under directed and switching topology," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–10, 2019.

[133] D. Zhao, Z. Wang, D. W. C. Ho, and G. Wei, "Observer-based PID security control for discrete time-delay systems under cyber-attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–13, 2019.

[134] X. Shen and X. Li, "Data-driven output-feedback LQ secure control for unknown cyber-physical systems against sparse actuator attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–13, 2019.

[135] G. Zhang, J. Qin, W. X. Zheng, and Y. Kang, "Fault-tolerant coordination control for second-order multi-agent systems with partial actuator effectiveness," *Information Sciences*, vol. 423, no. Supplement C, pp. 115 –127, 2018, ISSN: 0020-0255.

[136] J. Chen, W. Zhang, Y. Cao, and H. Chu, "Observer-based consensus control against actuator faults for linear parameter-varying multiagent systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1336–1347, 2017, ISSN: 2168-2232.

[137] D. Ye, M. Chen, and K. Li, "Observer-based distributed adaptive fault-tolerant containment control of multi-agent systems with general linear dynamics," *ISA Transactions*, vol. 71, no. Part 1, pp. 32 –39, 2017, Special issue on Distributed Coordination Control for Multi-Agent Systems in Engineering Applications, ISSN: 0019-0578.

[138] Y. Hua, X. Dong, Q. Li, L. Xie, and Z. Ren, "Fault-tolerant time-varying formation control for second-order multi-agent systems with directed topologies," in *2017 13th IEEE International Conference on Control Automation (ICCA)*, 2017, pp. 467–472.

[139] S. Yazdani and M. Haeri, "Robust adaptive fault-tolerant control for leader-follower flocking of uncertain multi-agent systems with actuator failure," *ISA Transactions*, vol. 71, no. Part 2, pp. 227 –234, 2017, ISSN: 0019-0578.

[140] S. Zheng, X. Zhang, and S. Sheng, "Simultaneous fault detection and control protocol design for coordination of multi-agent systems," in *2017 29th Chinese Control And Decision Conference (CCDC)*, 2017, pp. 6425–6430.

[141] C. Deng and G. H. Yang, "Cooperative adaptive output regulation for linear multi-agent systems with actuator faults," *IET Control Theory Applications*, vol. 11, no. 14, pp. 2396–2402, 2017, ISSN: 1751-8644.

[142] C. Deng and G.-H. Yang, "Adaptive fault-tolerant control for a class of nonlinear multi-agent systems with actuator faults," *Journal of the Franklin Institute*, vol. 354, no. 12, pp. 4784 –4800, 2017, ISSN: 0016-0032.

[143] Y. Yang and D. Yue, "Distributed adaptive fault-tolerant control of pure-feedback nonlinear multi-agent systems with actuator failures," *Neurocomputing*, vol. 221, no. Supplement C, pp. 72 –84, 2017, ISSN: 0925-2312.

[144] X. Li, Q. Wang, and J. Wang, "Fault-tolerant consensus for non-linear leader-follower multi-agent systems," in *2017 36th Chinese Control Conference (CCC)*, 2017, pp. 8601–8606.

[145] M. Eslami, M. B. Menhaj, and H. Danaee, "Fault tolerant control in multi-agent systems subject to actuator fault," in *2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, 2016, pp. 81–86.

[146] Q. Wang, W. Li, X. Cao, and M. Yu, "Distributed flocking with biconnected topology for multi-agent systems," in *2016 9th International Conference on Human System Interactions (HSI)*, 2016, pp. 183–189.

[147] N. Zhou, R. Chen, Y. Xia, and J. Huang, "Adaptive fuzzy control of leader-follower high-order nonlinear multi-agent systems with actuator faults," in *2016 15th International Symposium on Parallel and Distributed Computing (ISPDC)*, 2016, pp. 334–339.

[148] C.-H. Xie and G.-H. Yang, "Cooperative guaranteed cost fault-tolerant control for multi-agent systems with time-varying actuator faults," *Neurocomputing*, vol. 214, no. Supplement C, pp. 382 –390, 2016, ISSN: 0925-2312.

[149] S. Chen, D. W. C. Ho, L. Li, and M. Liu, "Fault-tolerant consensus of multi-agent system with distributed adaptive protocol," *IEEE Transactions on Cybernetics*, vol. 45, no. 10, pp. 2142–2155, 2015, ISSN: 2168-2267.

[150] B. Zhou, W. Wang, and H. Ye, "Cooperative control for consensus of multi-agent systems with actuator faults," *Computers and Electrical Engineering*, vol. 40, no. 7, pp. 2154 –2166, 2014, ISSN: 0045-7906.

[151] H. Yang, M. Staroswiecki, B. Jiang, and J. Liu, "Fault tolerant cooperative control for a class of nonlinear multi-agent systems," *Systems and Control Letters*, vol. 60, no. 4, pp. 271 –277, 2011, ISSN: 0167-6911.

[152] D. Yue, E. Tian, and Q. L. Han, "A delay system method for designing event-triggered controllers of networked control systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 475–481, 2013, ISSN: 0018-9286.

[153] Z. Wang, F. Yang, D. W. C. Ho, and X. Liu, "Robust $\mathcal{H}_\infty$ control for networked systems with random packet losses," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 4, pp. 916–924, 2007.

[154] X. Gao, X. Liu, and J. Han, "Reduced order unknown input observer based distributed fault detection for multi-agent systems," *Journal of the Franklin Institute*, vol. 354, no. 3, pp. 1464 –1483, 2017, ISSN: 0016-0032.

[155] Z. Zhang, F. Hao, L. Zhang, and L. Wang, "Consensus of linear multi-agent systems via event-triggered control," *International Journal of Control*, vol. 87, no. 6, pp. 1243–1251, 2014.

[156] D. Rotondo, F. Nejjari, V. Puig, and J. Blesa, "Model reference FTC for LPV systems using virtual actuators and set-membership fault estimation," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 5, pp. 735–760, 2015.

[157] S. Khodabandeh, H. Kharrati, and F. Hashemzadeh, "Control for leader–follower consensus of multi-agent systems with actuator faults using decentralized robust fault-tolerant control," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, pp. 1–13, 2020.

[158] R. Obermaisser, *Event-triggered and time-triggered control paradigms*. Springer Science & Business Media, 2004, vol. 22.

[159] M. Guinaldo, D. V. Dimarogonas, D. Lehmann, and K. H. Johansson, "Distributed event-based control for interconnected linear systems," in *Asynchronous Control for Networked Systems*, Springer, 2015, pp. 149–179.

[160] H. Li, X. Liao, T. Huang, and W. Zhu, "Event-triggering sampling based leader-following consensus in second-order multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 60, no. 7, pp. 1998–2003, 2015, ISSN: 0018-9286.

[161] D. Rotondo, H Sánchez, F Nejjari, and V Puig, "Análisis y diseño de sistemas lineales con parámetros variantes utilizando LMIs," *Revista Iberoamericana de Automática e Informática industrial*, vol. 16, no. 1, pp. 1–14, 2019, ISSN: 1697-7920.

[162] X. Wang and H. Su, "Self-triggered leader-following consensus of multi-agent systems with input time delay," *Neurocomputing*, vol. 330, pp. 70 –77, 2019, ISSN: 0925-2312.

[163] J.-P. Georges, D. Theilliol, V. Cocquempot, J.-C. Ponsart, and C. Aubrun, "Fault tolerance in networked control systems under intermittent observations," *International Journal of Applied Mathematics and Computer Science*, vol. 21, no. 4, 2011.

[164] J. H. Kim and H. B. Park, "$\mathcal{H}_\infty$ State feedback control for generalized continuous/discrete time-delay system," *Automatica*, vol. 35, no. 8, pp. 1443 –1451, 1999, ISSN: 0005-1098.

[165] J. W. Langelaan, N. Alley, and J. Neidhoefer, "Wind field estimation for small unmanned aerial vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 4, pp. 1016–1030, 2011.

[166] A. Rodríguez-Mata, G Flores, A. Martínez-Vásquez, Z. Mora-Felix, R Castro-Linares, and L. Amabilis-Sosa, "Discontinuous high-gain observer in a robust control uav quadrotor: Real-time application for watershed monitoring," *Mathematical Problems in Engineering*, vol. 2018, 2018.

[167] Z. Youmin and J. Jin, "Fault tolerant control system design with explicit consideration of performance degradation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 3, pp. 838–848, 2003.

[168] J. A. Vazquez Trejo, M. Adam Medina, J. A. Vazquez Trejo, C. D. García Beltrán, and J. García Morales, "A robust observer-based leader-following consensus for multi-agent systems.," *IEEE Latin America Transactions*, vol. 19, no. 11, pp. 1949–1958, 2021.

[169] Y. Tang, H. Gao, W. Zhang, and J. Kurths, "Leader-following consensus of a class of stochastic delayed multi-agent systems with partial mixed impulses," *Automatica*, vol. 53, pp. 346–354, 2015, ISSN: 0005-1098.

[170] M. Kishida, M. Ogura, Y. Yoshida, and T. Wadayama, "Deep learning-based average consensus," *IEEE Access*, vol. 8, pp. 142 404–142 412, 2020.

[171] H. Kheloufi, A. Zemouche, F. Bedouhene, and M. Boutayeb, "On LMI conditions to design observer-based controllers for linear systems with parameter uncertainties," *Automatica*, vol. 49, no. 12, pp. 3700 –3704, 2013, ISSN: 0005-1098.

[172] K. Zhou and J. C. Doyle, *Essentials of robust control*. Prentice hall Upper Saddle River, NJ, 1998, vol. 104.

[173] A. N. Langville and W. J. Stewart, "The Kronecker product and stochastic automata networks," *Journal of Computational and Applied Mathematics*, vol. 167, no. 2, pp. 429 –447, 2004, ISSN: 0377-0427.

[174] E. Altug, J. P. Ostrowski, and R. Mahony, "Control of a quadrotor helicopter using visual feedback," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 1, 2002, 72–77 vol.1.

[175] A. L. Salih, M. Moghavvemi, H. A. F. Mohamed, and K. S. Gaeid, "Modelling and pid controller design for a quadrotor unmanned air vehicle," in *2010 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, vol. 1, 2010, pp. 1–5.

[176] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 2247–2252.

[177] K. Ogata, *Modern control engineering*. Prentice hall, 2010.

[178] B. C. Kuo, *Automatic control systems*. Prentice Hall PTR, 1987.

[179] W. S. Levine, *The control systems handbook: control system advanced methods*. CRC press, 2018.