Cooperative Control of Quadrotors and Mobile Robots: Controller Design and Experiments

by

Bingxian Mu
B.Eng., Northwestern Polytechnical University, 2009
M.A.Sc., University of Victoria, 2013

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Mechanical Engineering

© Bingxian Mu, 2017
University of Victoria

All rights reserved. This proposal may not be reproduced in whole or in part, by photocopying or other means, without the permission of the author.

Cooperative Control of Quadrotors and Mobile Robots: Controller Design and Experiments

by

Bingxian Mu
B.Eng., Northwestern Polytechnical University, 2009
M.A.Sc., University of Victoria, 2013

Supervisory Committee

Dr. Yang Shi, Supervisor
(Department of Mechanical Engineering)

Dr. Daniela Constantinescu, Department Member
(Department of Mechanical Engineering)

Dr. Jane Ye, Outside Member
(Department of Mathematics and Statistics)

Supervisory Committee

Dr. Yang Shi, Supervisor
(Department of Mechanical Engineering)

Dr. Daniela Constantinescu, Department Member
(Department of Mechanical Engineering)

Dr. Jane Ye, Outside Member
(Department of Mathematics and Statistics)

ABSTRACT

Cooperative control of multi-agent systems (MASs) has been intensively investigated in the past decade. The task is always complicated for an individual agent, but can be achieved by collectively operating a group of agents in a reliable, economic and efficient way. Although a lot of efforts are being spent on improving MAS performances, much progress has yet to be developed on different aspects. This thesis aims to solve problems in the consensus control of multiple quadrotors and/or mobile robots considering irregular sampling controls, heterogeneous agent dynamics and the presence of model uncertainties and disturbances.

The thesis proceeds with Chapter 1 by providing the literature review of the state-of-the-art development in the consensus control of MASs. Chapter 2 introduces experimental setups of the laboratory involving two-wheeled mobile robots (2WMRs), quadrotors, positioning systems and inter-vehicle communications. All of the developed theoretical results in Chapters 3-6 are experimentally verified on the platform. Then it is followed by two main parts: Irregular sampling consensus control methods (Chapter 3 and 4) and cooperative control of heterogeneous MASs (Chapter 5 and 6). Chapter 3 focuses on the non-uniform sampling consensus control for a group of 2WMRs, and Chapter 4 studies the event-based rendezvous control for a group of asynchronous robots with time-varying communication delays. Chapter 5

concentrates on cooperative control methods for a heterogeneous MAS consisting of quadrotors and 2WMRs. Chapter 6 focuses on the design of a quadrotor flight controller which is robust to various adverse factors such as model uncertainties and external disturbances. The developed controller is further applied to the consensus control of the heterogeneous MAS.

Specifically, Chapter 3 studies synchronized and non-periodical sampling consensus control methods for a group of 2WMRs. The directed and switching communication topologies among the network are considered in the controller design. The 2WMR is an underactuated system, which implies that it can not generate independent x and y accelerations in the two-dimensional plane. The rendezvous control methods are proposed for 2WMRs. The algebraic graph theory and stochastic matrix analysis are employed to conduct the convergence analysis.

Although the samplings in the work of Chapter 3 are aperiodic, one feature is that local clocks of agents are required to be synchronized. Challenges arise in the practical control of distributed MASs, especially in the scenario that the global clock is lacking. Moreover, frequent samplings can result in redundant information transmissions when the communication bandwidth is limited. To address these problems, Chapter 4 investigates an event-based rendezvous control method for a group of asynchronous MAS with time-varying communication delays. Integral-type triggering conditions for each robot are adopted to be checked periodically. If the triggering condition is satisfied at one checking instant, the agent samples and broadcasts the state to the neighbors with a bounded communication delay. Then an algorithm is provided for driving 2WMRs to asymptotically reach rendezvous. The convergence analysis is conducted through Lyapunov approaches.

Most of the theoretical works on cooperative control are focused on controlling agents with identical dynamics. However, in certain realistic scenarios, some complex missions require the cooperation of different types of agent dynamics such as surveillance, search and rescue, etc. Tasks can be carried out with higher efficiency by employing both the autonomous ground vehicles and unmanned aerial vehicles. To achieve better performance for MASs, in Chapter 5, distributed cooperative control methods for a heterogeneous MAS consisting of quadrotors and 2WMRs are developed. Consensus conditions are provided, and the theoretical results are experimentally verified.

Many existing quadrotor control methods need exact model parameters of the quadrotor. In reality, when a quadrotor is conducting some tasks with extra payloads or with unexpected damages to the model structure, errors in parameters could result in the failure of the flight. External disturbances also inevitably affect the flight performance. To move a step further towards practical applications, in Chapter 6, a robust quadrotor flight controller using Integral Sliding Mode Control (ISMC) technique is investigated. In experiments, an extra payload with the position and mass unknown, is attached to destroy the accuracy of the model and to add disturbances. The designed controller significantly rejects negative effects caused by the payload during the flight. This controller is also successfully applied to an MAS consisting of a quadrotor and 2WMRs.

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

First of all, I would like to give my sincerest thanks to my supervisor, Dr. Yang Shi for all his guidance and training during my Master and PhD studies in the past six years. Without his constant help and encouragement, my doctoral milestone would have been impossible to reach. He gave me insightful suggestions at every step on finishing my thesis in countless discussions. His patience, enthusiasm, diligence, professionalism and immense knowledge in research have stimulated and will continuously inspire me to move forward fearlessly. Besides, he selflessly provided me lots of great opportunities to promote my academic abilities. I could not imagine having a better supervisor for my PhD study. I am proud of being his student and our relation remains very special in my entire life.

I am also grateful to the rest of my thesis committee, Dr. Daniela Constantinescu and Dr. Jane Ye for their constructive comments and encouragements, also for their questions which motivated me to improve my research from various standpoints. I would like to thank Dr. Constantinescu's precious support since the day I became a graduate student at University of Victoria, and Dr. Ye's valuable discussions and inspirations when I was in low moment. My sincere thanks also go to the External Member Dr. Yajun Pan for her important suggestions in improving the thesis.

I would like to express my deep appreciation to Dr. Fran Gebhard for her empathetic, professional and encouraging speech training when I was participating the UVic three minute thesis (3MT). She is amazing and is one of my favorite professors in my PhD study. I give my special thanks to Dr. Zuomin Dong for his generous help in enhancing my public speaking skills and his valuable suggestions on my future career planning. Moreover, I would like to express my very profound gratitude to UVic Career Educator Mr. John R. Fagan for his support of my job hunting. I benefited a lot from Patrick Chang during my TA and teaching experiences for undergraduate courses. He is always knowledgeable, energetic and ready to offer help to the students and colleagues.

I thank all my labmates from Applied Control and Information Processing Lab for enlightening discussions, countless sleepless nights we were working together, and all the fun time we have experienced in the past years. Previous senior students, Dr. Hui Zhang, Dr. Ji Huang, Dr. Jian Wu, Dr. Huiping Li, Dr. Xiaotao Liu and Dr. Mingxi Liu will always be my good examples in my future endeavors. I also thank Fuqiang Liu, Wenbai Li, Wei Chen, Lei Zuo, Qianyan Shen, Zhuo Zhang, Ying Li, Chao Yang, Tingting Yu, Jicheng Chen, Kunwu Zhang, Chao Shen, Yuanye Chen, Qi Sun, Qian Zhang, Haoqiang Ji, Xiang Sheng, Huaiyuan Sheng, Chen Ma, Zhang Zhang, and Yiming Zhao for their priceless friendships. Thanks also go to my reliable and respectable friends Thomas Gilmour, Karolina Papera Valente, Mario Bras, Robin Thomas and Sherwin Fernando. In particular, I would like to thank Rebecca Chung and Henry Yeh for their selfless help.

I gratefully acknowledge the financial support from Chinese Scholarship Council (CSC), Department of Mechanical Engineering and Faculty of Graduate Studies (FGS) at the University of Victoria, and Mr. Alfred Smith and Mrs. Mary Anderson Smith Scholarship. Besides, I am grateful to receive funds through research grants like those from Natural Sciences and Engineering Research Council (NSERC) of Canada, the Canada Foundation for Innovation (CFI) and the British Columbia Knowledge Development Fund (BCKDF).

I must express my very profound gratitude to my wife Qifei Wang and daughter Abigail Jiayang Mu for providing my unfailing support. Most importantly, I would like to thank my parents throughout my life.

To my parents

To my wife Qifei Wang

To my daughter Abigail Jiayang Mu

Acronyms

MAS multi-agent system

PID proportional-integral-derivative

LQR linear quadratic regulator

SMC sliding mode control

MPC model predictive control

DOF degrees-of-freedom

AUV autonomous underwater vehicle

2WMR two-wheeled mobile robot

Chapter 1

Introduction

This chapter gives some introductory knowledge about cooperative control of multi-agent systems (MASs). A review on consensus control of MASs is presented. The motivations of my research are presented at the end of this chapter.

1.1 An Overview on Cooperative Control of Multi-agent Systems

In recent years, cooperative control of multi-agent systems (MASs) has drawn great attention. The word "agent" represents a simple system dynamics, and it can be a wheeled mobile robot, a quadrotor or a manipulator. Traditionally, the controllers for coordinating the MAS's behaviors are designed in a centralized structure, which means that a centralized computer is employed to collect information from networks, to schedule tasks and to send orders to each agent. Without a doubt, the coordination of the MAS will be easily ruined if the number of agents grows large, or there exist unanticipated constraints in communication channels, i.e., time delays, data losses or disturbances. Alternatively, a more reliable strategy called distributed control is proposed. With the equipped microprocessor, sensors and actuators, each agent is able to collect data from the networks, to plan its own tasks, and to conduct scheduled actions. A substantial amount of work has been carried out on cooperative control of MASs to accomplish the tasks that are beyond the capability of a single agent, such as rescuing, unmanned aerial vehicles surveillance and deep sea exploration. The main objective of this research is to employ a group of simple agents collectively to conduct complex tasks with high reliability and efficiency.

Cooperative control of MASs has become a highly active research area, with many novel control methods proposed for diverse system dynamics ranging from multi-vehicle systems [4–6] to smart grids [7–9] to sensor networks [10, 11] and security for industrial cyber-physical systems [12, 13]. Additionally, some leading journals publish special issues on related topics. *IEEE Transactions on Industrial Electronics* Special Issue on Distributed Coordination Control and Industrial Applications (Volume 64, Issue 6, 2016) discusses theories and applications of distributed coordination control of multi-robot systems, sensor cooperative control and electric transportation systems. *IEEE/ASME Transactions on Mechatronics* Special Issue on Advanced Control and Navigation for Marine Mechatronic Systems (Volume 22, Issue 3, 2017) studies cooperative control of surface and underwater robotic vehicles. *ASME Journal of Dynamic Systems, Measurement, and Control* Special Issue on Analysis and Control of Multi-agent Dynamic Systems (Volume 129, Issue 5, 2007) includes topics on path planning, task assignment and formation control of MASs.

As shown in Figure 1.1, three prime issues to be addressed in cooperative control of MASs are: (i) system dynamics, which mathematically describe behaviors of agents, and stresses the fundamental importance of the topic; (ii) theoretical study, which rigorously offers stability conditions for control protocols; (iii) applications.

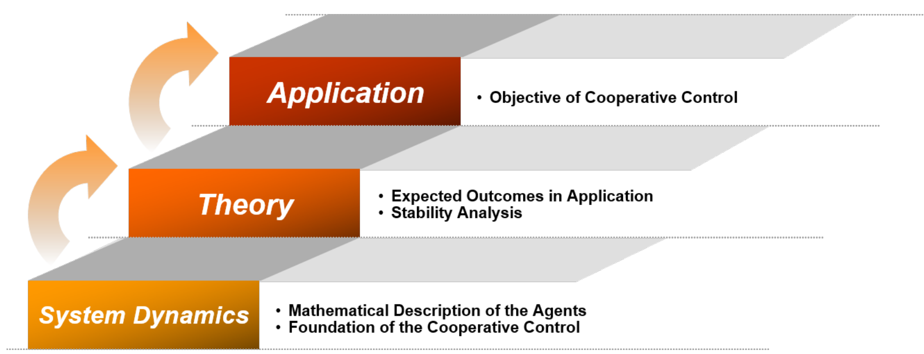


Figure 1.1: Three vital issues in the research of cooperative control of MASs.

As an important concern in cooperative control of MASs, the consensus problem has experienced a surge of research interests, aiming at forcing a group of agents' states to reach an agreement on a quantity of interest such as the rendezvous position, velocity and heading direction. Consensus can also be applied to solve problems for multi-vehicle systems, such as formation control [14–16], flocking [4, 17, 18], tracking [19–21], containment control [22], and so on. The distributed control strategy also acts as the mainstream in the study of consensus problems. The agent shares its information with partial of the networked agents. The interaction topology plays a vital role in the controller design and is usually described by a graph. In the following sections, we briefly illustrate a consensus problem and review the recent progresses on solving consensus problems.

1.2 What Is the Consensus Problem?

The consensus problem has been extensively studied as a key issue in the field of cooperative control of MASs. An MAS is usually consisting of a number of autonomous agents, and each agent has an embedded microprocessor to plan its own tasks. Simultaneously, built-in sensors and network are employed for the agent to measure the states of itself and to communicate with other agents respectively, such that the MAS will work in a collective way. Consensus is achieved if all agents reach an agreement on certain common feature such as position, velocity and heading direction. It is crucial to design appropriate control protocols for agents with information interactions over the network.

In Figure 1.2, we show an MAS consisting of five quadrotors labeled from 1 to 5. The information transmission can be either unidirectional or bidirectional as indicated by the arrow directions. The information flow between agent 3 and agent 2 implies that agent 3 receives information from agent 2, but the information of agent 3 can not be transmitted to agent 2. The bidirectional communication channel between agent 2 and 4 represents that two agents can receive information from each other.

Here we use a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ to mathematically describe the communication topology among the agents, where $\mathcal{V} = \{v_1, v_2, v_3, v_4, v_5\}$ represents the node set. $\mathcal{E} = \{(v_1, v_2), (v_1, v_5), (v_2, v_1), \ldots, (v_4, v_2)\} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges, which indicates all existing information flows, i.e., if there is an information flow from v_i to v_j, then $(v_i, v_j) \in \mathcal{E}$. $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{5\times 5}, i, j = 1, 2, \ldots, 5$, is the adjacency matrix, with $a_{ij} > 0$ if $(v_j, v_i) \in \mathcal{E}$; otherwise $a_{ij} = 0$. It is assumed that the agent does not transmit the information to itself, which implies that $a_{ii} = 0, i = 1, 2, \ldots, 5$. The communication topology can be either fixed or time-varying. More details on the communication topology can be found in [23]. Next we review the three issues mentioned in Section 1.1 associated with the consensus problem.

- **System Dynamics**

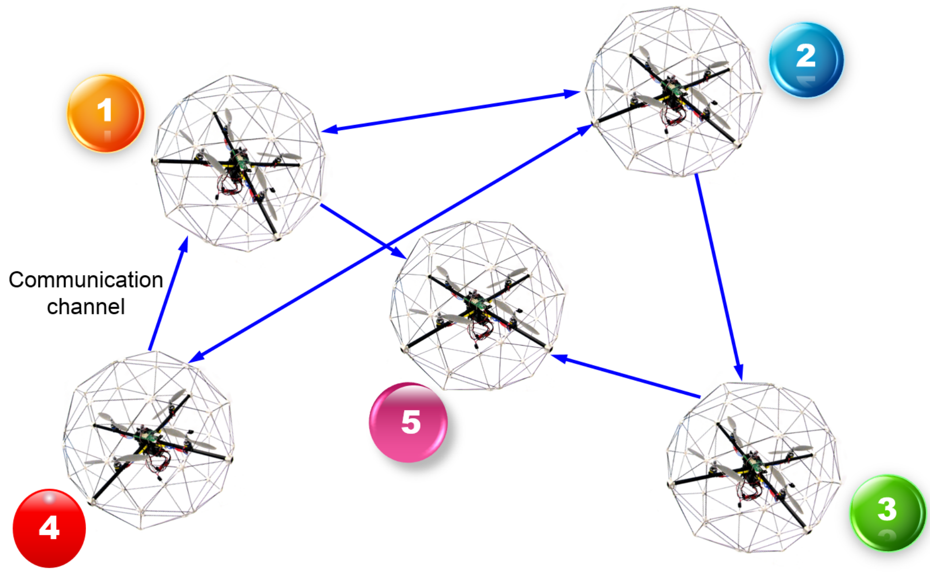


Figure 1.2: Illustration of an MAS: A group of five quadrotors.

Without loss of generality, the dynamics of an agent in the MAS can be described using the following differential equation:

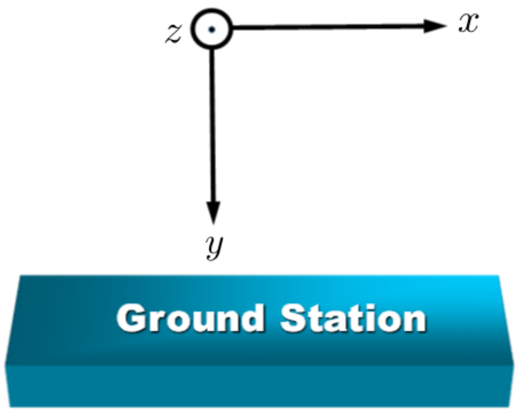
$$\dot{x}_i = f(x_i, u_i), i = 1, 2, \ldots, N,$$

where $x_i \in \mathbb{R}^n$ and $u_i \in \mathbb{R}^m$ are the state and input of agent i. We say that consensus is reached if $\lim_{t\to\infty} \|x_i(t) - x_j(t)\| = 0, \forall i, j = 1, 2, \ldots, N, i \neq j$.

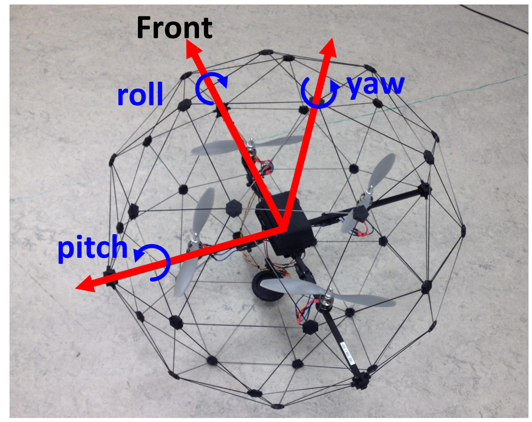
In this example, we use the earth-fixed coordinate system [1] to describe the motions of the quadrotor in the three-dimensional space, as shown in Figure 1.3(a). x and y axes are in the horizontal plane, and z-axis vertically points up. The roll, pitch and yaw angles $\phi_i(t), \psi_i(t)$ and $\theta_i(t)$ of the quadrotor is shown in 1.3(b).

Assume that the dynamics of the quadrotor in x, y, z-axes are decoupled, indicating that the motions of the quadrotor along three axes can be controlled independently. The dynamics of agent i in x-axis is given by [1]:

$$\dot{X}_i(t) = AX_i(t) + Bu_i(t), \tag{1.1}$$



(a) Positioning system frame.



(b) Roll, pitch and yaw axes of the quadrotor.

Figure 1.3: Positioning system and the quadrotor.

where $X_i(t) = \left[x_i(t), \dot{x}_i(t), \phi_i(t), \dot{\phi}_i(t), p_i(t)\right]^{\mathrm{T}}$, $A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{4K}{M} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{2KL}{J} \\ 0 & 0 & 0 & 0 & -\omega \end{bmatrix}$, and $B = [0, 0, 0, 0, \omega]^{\mathrm{T}}$. $x_i(t)$ is the position of agent i along x-axis, $p_i(t)$ is the actuator dynamics. $u_i(t)$ is the control protocol to be designed. ω is the actuator bandwidth, M represents the total mass of the quadrotor, L denotes the distance between the propeller and the center of gravity. J is the rotational inertia of the quadrotor in roll axis and K is a positive constant gain.

- **Theory**
  The next step is to design control protocols $u_i(t)$ for agent $i, i = 1, 2, \ldots, 5$, such that $\lim_{t\to\infty} \|x_i(t) - x_j(t)\| = 0, \forall i, j = 1, 2, \ldots, 5$. The convergence analysis of consensus should be rigorously conducted, meaning that the feasibility of proposed control methods and the stability of systems should be mathematically guaranteed before practical applications. For more literature review of the theoretical approaches in solving the consensus problem of MASs, see Section 1.3.

- **Application**
  The practical applications help to verify the effectiveness of the proposed control methods, and then results of the applications also reciprocally improve the design of controllers.

1.3 Literature Review on the Consensus Problem

This section provides a literature review of the earlier works on consensus problems.

The consensus problem has been earlier studied in the field of data management systems and computer science. The author in [24] describes a commit problem for distributed databases: Each agent has an initial opinion to commit or abort a transaction, and then this opinion is transmitted to all other agents directly or in several hops via other agents. An agent prefers committing the transaction if all agents in his/her connection choose "committing", and otherwise "aborting" is his/her preference. All agents are assumed to communicate with others and finally get to a common decision "commit" or "abort". In [25], the consensus problem is studied for the parallel and distributed optimization algorithm in the signal processing. Later, intensive theoretical investigations on consensus problems start surging. Jadbabaie *et al.* [26] analytically study the heading convergence condition for the Vicsek's model [27]: The discrete-time agents move with the same speed in the plane, and each agent updates the heading based on the information of itself and its neighbors. Without the centralized controller, all agents can eventually move in the same direction. Specially, the graph theory technique is employed to analyze biologically inspired models and later becomes one of the main approaches for the stability analysis of MASs. [28] discusses the consensus problem for a variety of cases: Fixed or switching network topology; presence of time delays in the communication channels; directed or undirected information flow, and so on. The consensus performance is determined by the algebraic connectivity of the topology, and the maximum time delay that an MAS can tolerate is also calculated. In [29], the condition to guarantee consensus behaviors of an MAS is that a spanning tree exists frequently enough in the directed changing interaction graphs. The authors in the above works build the theoretical framework for solving the consensus problem based on algebraic theory, matrix theory and graph theory.

1.3.1 Consensus Problems from Different Perspectives

The study of consensus problems has been developed along different directions. Some representative topics are shown in Figure 1.4.

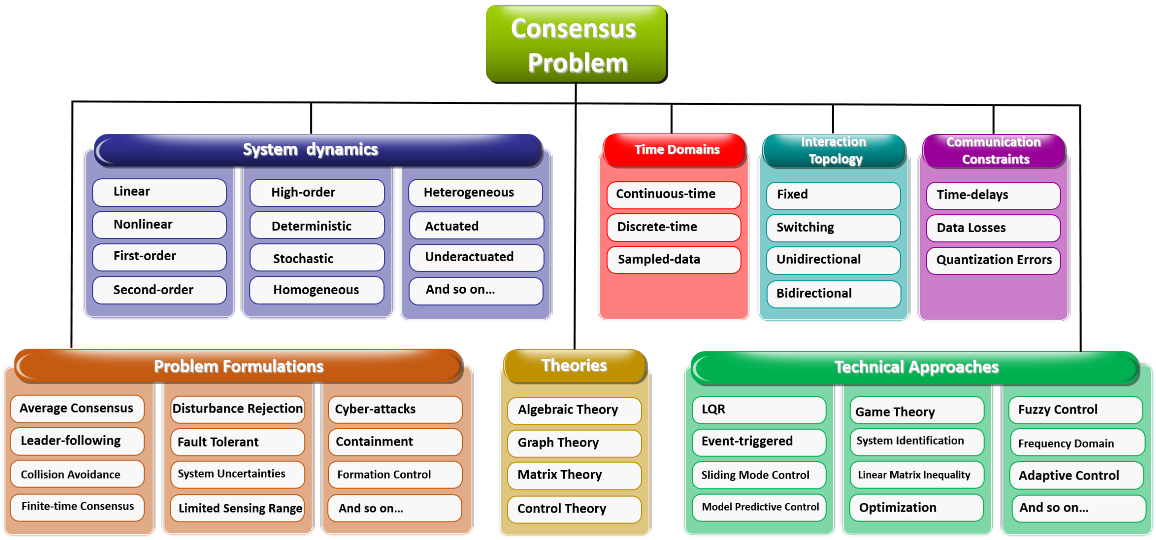


Figure 1.4: Categories of theoretical studies of consensus problems.

System Dynamics

In general, the dynamics of agents can be broadly classified into two types: *Linear* and *nonlinear systems*. In the early stage, the linear system receives major attention [26, 28, 29] due to its simplicity from both theoretical study and implementation standpoints. However, it has become apparent that in order to apply the linear methodological framework to real world problems, sometimes we have to pay attention to the inherent nonlinear characteristics of the dynamics. The nonlinearity is investigated as an intrinsic feature of the dynamics in certain scenarios. In [30–32], the authors design consensus controllers with the consideration of the nonlinear terms in MASs. Some consensus problems are associated with the nonlinear systems, e.g., multi-pendulum synchronization problem is studied in [33], and the consensus protocols for Euler-Lagrange dynamics such as manipulators are discussed in [34, 35].

The system dynamics can also be characterized by the order of the differential equations. The *first-order dynamics* involving the position information of the agent is relatively simple and is studied in the pioneer work [26, 28, 29]. Later on, more researchers have been working on first-order consensus problems. Wu and Shi study the first-order consensus problem for an MAS under the sampled-data setting. They investigate the communication constraints such as uniform time delays, time-varying delays, packet losses and nonuniform sampling control strategy [36–38]. The stability conditions are provided. It is found that spanning trees in the communication topologies and appropriately chosen control gains are important issues to guarantee the consensus behavior. Xiao *et al.* discuss the first-order consensus problem considering finite-time formation control [39], asynchronous rendezvous analysis [40], and so on. Besides first-order dynamics, the research on the consensus problem of MASs with more complicated dynamics is also widely conducted. *The second-order* consensus problem has received growing attention because it is more realistic to characterize MASs with double-integrator dynamics. Mu *et al.* [6] study the second-order consensus problem of multiple unmanned aerial vehicles with time delays governed by a Markov chain. The authors in [41] investigate the second-order consensus problem with arbitrary sampling periods. In addition, more topics on second-order consensus problems such as partial state consensus [42], leader-following consensus [43], finite-time consensus [44], communication link failure [45], limited interaction ranges [46] have attracted wide interests. Moreover, in realistic situations, there are many systems whose dynamics must be described by high-order models, for example, the quadrotor dynamics in (1.1). Consensus of the MAS with *high-order dynamics* is studied in [47–50]. The work in [49] generalizes first-order and second-order consensus algorithms to a high-order consensus algorithm and demonstrates the sufficient conditions to ensure consensus. In [51], the authors extend Ren and Beard's results [29] to the consensus control of an MAS with high-order dynamics. The consensus controller is designed for controllable linear systems, and it is assumed that the interaction connectivity condition remains the same as in [29]: The union of the directed graphs has a spanning tree frequently enough. Compared with existing results, a more general hypothesis on nth order agent dynamics is considered.

Regarding noises in the state measurements, the consensus problem can be studied with *deterministic* and *stochastic dynamics*. In the above discussion, most of the control protocols are studied with noise-free states, indicating that the exact data is measured and broadcast among the MAS. In fact, the data measurement and transmission processes involve using sensors, quantization techniques and wireless networks, which are inevitably affected by intrinsic uncertainties in the environment. Accordingly, it is very important to consider the stochastic features when studying the consensus problem. In order to minimize the error in the consensus result, a least square optimization approach is proposed in [52] by choosing appropriate coefficients in averagely estimating the additive noises. Huang *et al.* [53] study stochastic algorithms to solve the consensus seeking problem for the MAS with noises in the measurement of the neighbors' states. Further, it is proved that the existence of a spanning tree in the interaction topology is a critical need to guarantee the mean square and almost sure convergence in [53]. Random communication link failure is also considered as a stochastic feature in this work. The authors in [6, 36] formulate communication constraints such as time delays into Markov jump linear systems. Stochastic characteristics of time delays are illustrated in the probabilistic distribution, which helps to reduce the conservativeness.

In the above literature review, MASs are usually assumed to be the so-called *homogeneous systems*, meaning that all agents have same dynamics with identical model structures and parameters. However, in some situations, it is difficult or impossible to employ homogeneous MAS to cooperatively accomplish the mission, e.g., the task of rescuing which requires the coordination of the ground vehicles and the unnamed aerial vehicles. The study of heterogeneous MASs allows further breadth of military and civilian applications. Zheng *et al.* [54] study the consensus problem of a heterogeneous MAS consisting of first-order and second-order dynamics. The authors in [55, 56] design consensus protocols for linear heterogeneous MASs. In [56], agents can be a general nth order dynamics, and there might exist uncertain parameters in model structures. Output regulation theory is employed to analyze the convergence of consensus.

If a system has less number of actuators compared to its degrees-of-freedom (DOF), this system is the so-called *underactuated dynamics*. Many research studies have been carried out on the consensus problem for underactuated MASs. In [57], an adaptive control consensus controller is designed for a group of underactuated thrust-propelled vehicles. The consensus in multiple underactuated Euler-Lagrange systems is investigated in [58]. The proportional plus damping controllers are proposed for the MAS, and the synchronization behavior is studied under the fixed communication topology. Additionally, consensus problems of underactuated MASs with a variety of dynamics such as spacecrafts [58], planar rigid bodies [59] and autonomous underwater vehicles (AUVs) [60] are investigated in the literature.

The aforementioned review on consensus problems categorized by different system dynamics are summarized in Table 1.1.

Time Domains

Consensus problems can be studied in different time domains: *Continuous-time*, *discrete-time* and *sampled-data* frameworks. It should be noted that MASs will intuitively be characterized by using the continuous-time methodological framework

Table 1.1: Selected papers on consensus problems classified by system dynamics.

System dynamics	Feature	Related work
Linear	Superposition property.	[26, 28, 29]
Nonlinear	Superposition property does not hold.	[30–32]
First-order	Single-integrator dynamics.	[36–38]
Second-order	Double-integrator dynamics.	[6, 41]
High-order	nth ($n > 2$) order dynamics	[47–51]
Stochastic	Noises in sensing, quantization or transmission.	[36, 52, 53]
Heterogeneous	Different model structures for agents.	[54–56]
Underactuated	less number of actuators than DOF.	[57–60]

because the states of most dynamics evolve continuously in real operation; e.g., temperature changes in a thermodynamic system, motions of the human body, trajectories of a flying quadrotor, and so on. As reported in [28, 29, 40, 61–63], many research studies are conducted on continuous-time consensus protocols. On the other hand, it becomes more convenient and popular to study MASs under the discrete-time framework due to the development of the digital signal processing and communication technologies. In [26], the agent updates the states by averaging its neighbors' states. Much attention on discrete-time consensus can be found in [64–66], etc. Although much effort has been made to the above two types of time domain frameworks, it is still far from completion because usually MASs are operated in the analog world and microcontrollers embedded in agents process digital signals. The sampled-data control system involves continuous-time system dynamics and discrete-time controllers. The inter-sample behaviors of agents are not obtained. The sampling is usually assumed periodic and synchronized for all agents [6, 36]. However, it is always difficult for one to sample periodically in reality due to the communication constraints, e.g., time delays, data losses, and so on. In this case, the study of irregular sampling for the control protocol is of practical significance. The consensus control for an MAS with double-integrator dynamics and non-uniform sampling is investigated in [38]. In [67], an asynchronous consensus protocol for an MAS with arbitrary sampling intervals is developed.

Table 1.2: Selected papers on consensus problems classified by time domains.

Time domains	Feature	Related work
Continuous-time	Differential equations to describe dynamics.	[28, 29, 40, 61]
Discrete-time	Difference equations to describe dynamics.	[26, 64–66]
Sampled-data	Continuous-time dynamics to discrete-time ones.	[6, 36, 38, 67]

Interaction Topologies

The interaction topologies can be generally categorized into the fixed topology and switching topologies. The fixed topology is time-invariant, and considerable attention has been paid to the study [28, 53, 68]. The algebraic connectivity among the agents builds up the key link between information interaction situations and the convergence analysis of consensus. In fact, communication constraints in unreliable channels have an impact on the fixed interaction topology, such as time delays caused by different data transmission rates, the limited communication range, data losses, and malicious cyber-attacks. To keep consensus protocols implementable, detailed studies on switching topologies are necessary. Consensus protocols under dynamically changing topologies are investigated in [29]. If a stochastic matrix's infinite self-products have the identical rows, this matrix is called stochastic indecomposable and aperiodic (SIA) [29]. By using the knowledge of SIA and incorporating switching topologies with MAS dynamics, the authors study the consensus condition: A spanning tree appears frequently enough in the union of changing interaction topologies. The model of the switching topologies is formulated as a Markov process by Wu *et al.* in [36], indicating that changes of the topology is probabilistically determined by current communication links. More related work on the consensus problem with changing interaction topologies can be found in [69–71].

Table 1.3: Selected papers on consensus problems classified by interaction topologies.

Interaction topologies	Feature	Related work
Fixed topology	Time-invariant topology.	[28, 53, 68]
Switching topologies	Time-varying topology.	[36, 69–71]

Communication Constraints

The communication constraint is an essential issue that has been mentioned in the above literature review. It exists ubiquitously in practical situations and may deteriorate the MAS performance. Here a short review on the communication constraints is provided with special regards to *time delays* and *data losses*. Much research has been carried out on the consensus problem with time delays. The case that a constant time delay exists in all links is studied for the average consensus protocol in [28]. It shows that the upper bound of the time delay is inversely proportional to the largest eigenvalue of the *Laplacian matrix* of the fixed interaction topology. One limit in this work is that the agent who sends information to others also suffers the same time delay as the agent who receives the information. Later, in [72], the situation that time delays only affect data receivers is studied. The convergence analysis of consensus is conducted by using Lyapunov-based approach and the sufficient condition in terms of Linear Matrix Inequality (LMI) is provided. Consensus in MASs with time-varying delays also receives intensive studies from different perspectives: Unbounded time delays [73], stochastic process governed time delays [36], finite-time consensus [74], asynchronous sampling consensus [40] and so on. In addition, the data loss is another important concern of communication constraints when studying the consensus problem, which is usually caused by the long time delay or the failure of the communication link. The work in [37, 75, 76] characterize data losses by using Bernoulli processes. In [77], the designed control protocol can solve the mean-square consensus problem if the data loss probability is within a calculated bound. In [78], three approaches dealing with data losses are discussed: (1) The missing data is set to be zero, meaning that the failed link will not affect the control input of the agent that should have received the data. (2) The previous received information is used again if a data loss happens. (3) A predictor is designed for the data receiver to estimate the missing information. Moreover, some other communication constraints such as quantization errors, noisy measurements, cyber-attacks are also intensively investigated in the literature.

Problem Formulations

Spurred by the pioneer works in [26, 28, 29], a broad class of consensus problems have been formulated in the literature, as shown in Figure 1.4. Here we present a discussion on how researchers describe the consensus problems from a variety of standpoints.

Table 1.4: Selected papers on consensus problems classified by communication constraints.

Communication constraints	Feature	Related work
Time delays	Time lag in the interactions.	[36, 72–74]
data losses	Failure of the interactions.	[37, 75–78]

The *average consensus* is studied in [37, 49, 79, 80]. All agents in the MAS will converge to the exact average value of their initial states. When an agent moves, the average value of the states can remain constant by changing another agent's states with the same magnitude in the opposite direction [37, 79]. More complicated situations such as switching topologies, time-varying delays in the average consensus problem are discussed in [81].

Another interesting perspective to formulate the consensus problem is that whether there exist one or more leaders in the MAS. In the *leader-following consensus* problem, the leader can be either static or dynamic, and then accordingly the problem is formulated as consensus regulation problem with static leaders or consensus tracking problem with dynamic leaders. It is a more complicated case that only a portion of agents in the MAS receive the information from the leaders in a consensus tracking problem. An important concern on the graph analysis for the leader-following consensus is that the followers' information can not affect the leaders because that the leaders are always the root nodes in the directed spanning trees. Some theoretical works are given in this field: Sliding mode controllers and the uncertain MAS are studied for the leader-following consensus in [43] and [62], respectively. It is also worthwhile to mention that experimental studies for the leader-following flocking are conducted, such as flocking control for a group of robotic fish in [4] and for the multiple four-wheeled robots in [17].

Obstacle or collision avoidance is also a vital issue in developing the consensus protocols for MASs. The artificial potential field approach is a commonly used method [82, 83] to solve cooperative control problems with obstacle or collision avoidance. The concept "safety region" is proposed for an agent, and then attractive potential fields from the target and repulsive potential fields from the obstacles or from other vehicles are assigned to the agent. The repulsive force will increase rapidly if there exist obstacles or other vehicles approaching the "safety region" of the agent. Finally, the resultant force acting on the agent controls the agent to finish the tasks safely.

An alternative approach to deal with cooperative control of MASs with obstacle or collision avoidance is optimization-based method. In [84], the trajectory of the agent is planned by using Model Predictive Control (MPC), and collision-avoidance is considered as coupled constraints.

When investigating consensus problems, the *convergence rate* is a critical index to evaluate the proposed control methods. In [28], the convergence rate can be enhanced by maximizing the algebraic connectivity of the communication topology. In [36], the properly chosen feedback control gains in the control protocol can also improve the convergence speed. However, it is analyzed that the works in the above efforts are on asymptotic consensus, meaning that consensus can not be reached in finite time. By proposing the finite-time Lyapunov stability analysis and using time-varying weighted directed graphs, the authors in [85] provide the finite-time consensus protocol for an MAS. A consensus tracking algorithm using terminal sliding-mode control is proposed in [86]. The finite-time stability is proved based on Lyapunov theory, and the proposed control protocol is robust to input disturbances and model uncertainties. Besides, some other interesting challenges are discussed on the topics of finite-time consensus for MASs, see [87, 88] and references therein.

Consensus problems have been studied in presence of *input disturbances*. To reject deterministic disturbances such as time-invariant or sinusoidal disturbances, one approach is to design the controller for the task while suppressing the effects caused by disturbances, which is referred to as the internal model principle [89]. The deterministic disturbances can also be coped with the output regulation approach in [90, 91]. From the state or output measurements, the disturbances are firstly estimated, and then the estimated disturbances will be used in the controller design to compensate the effects of disturbances. In [92], disturbances are modeled as a linear exogenous system and a disturbance observer is proposed. The stability is analyzed by using the LMI approach, and accordingly control gains are calculated by solving the LMIs. Later, exogenous disturbance systems and disturbance observer techniques are applied to nonlinear MASs, by using Input-to-State Stability approach to analyze the convergence of consensus.

Above we list some perspectives from which consensus problems are formulated and studied. However, only limited number of points are presented. A considerable amount of interesting consensus problems have been formulated from other different standpoints, such as limited sensing ranges [93], system uncertainties [64], containments [22] and more.

Table 1.5: Selected papers on consensus problems classified by problem formulations.

Problem formulations	Feature	Related work
Average consensus	Unchanging average value of the states.	[37, 49, 79, 80]
Leader-following	One or more leaders in MASs.	[43, 43, 62]
Collision avoidances	Collision free with obstacles and vehicles.	[82–84]
Finite-time consensus	Convergence reached in finite time.	[85, 87, 88]
Disturbances	Disturbances in the control inputs.	[89–92]
Others	...	...

1.3.2 Theoretical Approaches for Solving Consensus Problems

Graph Theory

We use a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ to model the communication topology among agents. $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ is the vertex set representing N agents. $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set. $e_{ij} = (v_j, v_i) \in \mathcal{E}$ denotes that the information of agent j can be transmitted to agent i, and agent j is called the neighbor of agent i. $\mathcal{N}_i = \{v_j \in \mathcal{V} : e_{ij} \in \mathcal{E}\}$ denotes the neighbor set of agent i. $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ is the adjacency matrix where $a_{ij} = 1$ if $e_{ij} \in \mathcal{E}$, otherwise $a_{ij} = 0$. $a_{ii} = 0, \forall i = 1, 2 \dots, N$. The graph has a spanning tree rooted at v_i if there exists an ordered sequence of edges that starts from v_i and reaches any other node v_j $(j = 1, 2, \dots, N, j \neq i)$ in the graph, e.g., (v_i, v_{m_1}), (v_{m_1}, v_{m_2}),..., $(v_{m_p}, v_j) \in \mathcal{E}$. The *graph Laplacian* $L = [l_{ij}] \in \mathbb{R}^{N \times N}$ is defined as: $l_{ij} = -a_{ij}$, $\forall i \neq j$; $l_{ii} = \sum_{j=1, j \neq i}^{N} a_{ij}$. More details on the knowledge of graph theory can be found in [23] and references therein.

Control Theories and Methods

Various control methods are applied to study consensus problems. The frequency domain technique such as Nyquist criterion is used to provide sufficient conditions on consensus in [28]. Lyapunov stability analysis is one of the prevalent approaches, which has been intensively used in proving the convergence of consensus, [6, 36]. Input-to-State Stability (ISS) is a suitable tool to be used for the stability analysis of nonlinear dynamics [94].

As mentioned above, the convergence rate is an index to evaluate the system performance. The quadratic cost function consisting of the state convergence and the control effort is employed as another important index to estimate the performance of the systems. Accordingly, *Linear Quadratic Regulator*-based (LQR-based) optimal consensus control protocols are proposed. By minimizing the quadratic cost function, the closed-loop control gain matrix is obtained. Convergence of consensus is also analyzed, see details in [95, 96].

Model Predictive Control (MPC) solution for the consensus problem also attracts some attention. Agents solve a set of constrained finite-time optimal control problems involving neighbors' information at each step and obtain a sequence of control inputs. The first control signal is then applied to the MAS at the current time instant. One of the major advantages of the MPC approach is that the MAS cooperative control will be conducted from an optimization point of view. Some physical limits such as input bounds, safety regions for the collision avoidance can be formulated as constraints in the optimization problem. In [97], the authors propose MPC consensus control schemes for discrete-time first- and second-order dynamics with time-varying interaction topologies. For more works on solving the consensus problem using MPC-based approaches, refer to [84, 98] and so on.

Periodical sampling can be easily implemented to coordinate MASs under the sampled-data control strategy. However, redundant information transmissions or computations will be conducted if time-scheduled sampling intervals are too small. For the *event-triggered control* strategy, the controller updates when the predefined event-triggering condition is satisfied. It becomes obvious that by using event-triggered control methods, not only the energy consumption is reduced, but also less controller updates and data transmissions are needed so that the lifespans of devices can be increased. Early work on event-triggered consensus can be traced back to [99], and further studies are conducted addressing various points of view. Examples are found in first-order consensus [100], second-order consensus [101], consensus in general linear MASs [102], nonlinear MASs [103], observer-based consensus [104], self-triggered consensus [105], etc.

Model uncertainties and disturbances exist ubiquitously and tend to deteriorate the performance of MASs. *Sliding mode control* (SMC) is usually an effective technique to deal with uncertainties and disturbances in the controller design for agents to reach the desired formation in finite time. Normally, a function called sliding surface is designed, and then the control input enforces the system state to reach and to slide along the designed surface. In [106], the Laplacian matrix dependent sliding surface is proposed for MASs to conduct consensus behaviors. The authors in [88] investigate the discontinuous integral sliding mode consensus control considering the relative information among high-order agents.

More theoretical methods are applied to solve consensus problems, such as adaptive control [107], LMI [36], game theory [108] and more.

1.3.3 Application-oriented Research on Cooperative Control of MASs

The goals of the research on MASs are to provide stability conditions for cooperative control, and then to utilize proposed control methods on existing systems. In most of the literature, theoretical results are only verified by simulations, and relatively few studies can be found from the application point of view.

In [109], a leader-following formation control protocol is investigated. The leader is supposed to move at a constant velocity and the follower only measures the relative positions between itself and other agents. The formation control of the MAS is conducted by using the graph theory and nonlinear adaptive control theory. A group of quadrotors *Arducopter*[1] are employed to verify the effectiveness of proposed control methods. An artificial potential field method is studied in [110] for the formation control of the specific shape for an MAS, and the collision avoidance is also considered in the control method. Experimental studies are provided for this work by using a collection of mobile robots which are subject to nonholonomic constraints. In [111], robots equipped with the monocular cameras produced by Pioneer 3Dx Inc.[2] are used to carry out orientation consensus studies. The image processing algorithm is incorporated into the consensus algorithm by considering switching interaction topologies, and only the visual information is transmitted among agents. An interesting collaborative task "Cleanup" is described in [112]: There are some small colored boxes scattered in the room and a group of *Pioneer DX* robots are controlled to localize the boxes from build-in cameras, and then to push the boxes towards the wall. By using the sonar senor, trajectories of robots are planned considering collision avoidance among agents. Ranjbar-Sahraei *et al.* [113] explore the formation control using the artificial potential field method with robust control techniques. The adaptive fuzzy

[1] [Online]. Available: http://dev.ardupilot.com/

[2] [Online]. Available: http://www.mobilerobots.com/ResearchRobots/PioneerP3DX.aspx

logic algorithm is proposed to estimate unknown parameters in system models. The designed control methods are applied to a swarm of fully actuated mobile robots *Palm Pilot Robot Kit*[3]. The consensus controller for multiple 3-DOF helicopters are studied in [19]. The decentralized nonlinear controller and disturbance estimation term are used to compensate vehicle model uncertainties. The theoretical result is tested on an experimental platform involving 4 helicopters provided by Quanser Consulting Inc.[4] Flocking and formation control of multiple robotic fish is presented in [4]. Robotic fish are swimming in the surface of a pool and are localized by an overhead camera. Leader-following formation control is developed for swarming fish with information interactions among followers.

1.4 Motivations and Contributions

In the aforementioned review, it is shown that agent dynamics, control protocol designs, stability analyses and practical applications are the concerns to be addressed for solving a consensus problem. In the following, motivations and objectives of each chapter are summarized.

In a sampled-data scheme dealing with consensus problem, continuous-time states of agents are sampled periodically, and discrete-time controllers are designed [114]. Communication constraints such as time delays and data losses result in difficulties for applying the periodic sampling to MASs. Many research studies have been carried out on MAS irregular samplings [6, 67]. Particularly, a consensus protocol with non-uniform samplings is proposed for the MAS under the fixed interaction topology in [38]. However, the application of the control method considering the fixed communication topology is relatively limited. Chapter 3 studies non-uniform sampling consensus protocols for a group of 2WMRs under switching communication topologies. Control methods for underactuated 2WMR dynamics are explicitly developed. Based on algebraic graph theory and stochastic matrix, a sufficient condition to ensure consensus is given by choosing appropriate control gains. Control method implementations are conducted on a practical MAS of 2WMRs.

Consensus problems with special regard to asynchronous agent behaviors attract wide concerns because it is challenging to synchronize local clocks for a distributed

[3] [Online]. Available: http://www.cs.cmu.edu/ pprk/
[4] [Online]. Available: http://www.quanser.com/products/3dof_helicopter

MAS, i.e., in a GPS-denied environment. On the other hand, frequent and periodical samplings can result in a waste of interactions and computation resources when agents in an MAS are equipped with resource-limited microcontrollers, or the communication bandwidth is limited. Moreover, inherently existing time delays in communication channels can also deteriorate the controller performance. Chapter 4 focuses on an event-based consensus control for an asynchronous MAS considering time-varying delays. We design integral-type triggering conditions for each agent to check periodically, such that the average performance of the agent is comprehensively considered from the most recent controller update instant to the event-checking instant. We also provide a rendezvous algorithm for an MAS consisting of 2WMRs.

Cooperative control of heterogeneous MASs is of significance in some specific scenarios, e.g., the task of rescuing which requires the coordination of ground vehicles and unnamed aerial vehicles. In Chapter 5, we deal with a rendezvous problem of a heterogeneous MAS with 2WMRs and quadrotors. The LQR-based control methods for underactuated 2WMR dynamics and for quadrotors are proposed respectively. The state convergence of the heterogeneous MAS is guaranteed if switching interaction topologies always have a spanning tree. The experimental tests are also presented.

In practical applications, the flight performance of a quadrotor can be affected by extra payloads, unexpected variations to the model structure and parameter errors. It is important to design a flight controller which is robust against model uncertainties and external disturbances. Chapter 6 investigates the inner-outer loop structured ISMC-based flight controller for a quadrotor. We prove that the waypoint tracking task for a quadrotor can be conducted in finite time if model uncertainties and disturbances are upper bounded. In experiments, an extra payload with unknown mass is attached to the random position on a quadrotor. By using the designed controller, the flight performance is significantly improved compared with using the traditional LQR-based flight controller. We also implement the flight controller on the heterogeneous MAS involving a quadrotor and 2WMRs.

Chapter 2

Experimental Setup

2.1 An Overview on Quanser Multiple Unmanned Vehicle Systems (UVS) Lab

For the convenience of readers, this chapter briefly introduces the experimental platform in the Distributed Optimization and Control for Multi-Agent Systems (DOCMAS) Lab, in the Department of Mechanical Engineering, University of Victoria. The platform will be used for MAS experimental studies throughout the whole thesis. Detailed information can be found in lab manuals [1–3] provided by Quanser Inc. Figure 2.1 shows the layout of the experimental platform. The platform, consisting of quadrotors, 2WMRs and third-party built vehicles, is an open-architecture platform. In this process, control algorithms are programed in Matlab®/Simulink®, and then the Quanser real-time software QUARC® compiles the designed controller into ARM executable files. The ARM executable files are downloaded to the target vehicles through wireless network.

The remainder of the chapter is organized as follows. Section 2.2 and 2.3 describe mechanical and electrical components of the quadrotor Qball-X4 and 2WMR Qbot, respectively. Section 2.4-2.6 introduce the control software QUARC®, the communication module and the indoor positioning system setup.

2.2 Quanser Qball-X4

Over the past years, unmanned aerial vehicles (UAVs) have drawn considerable attention in the field of robotics. In particular, many research studies have been

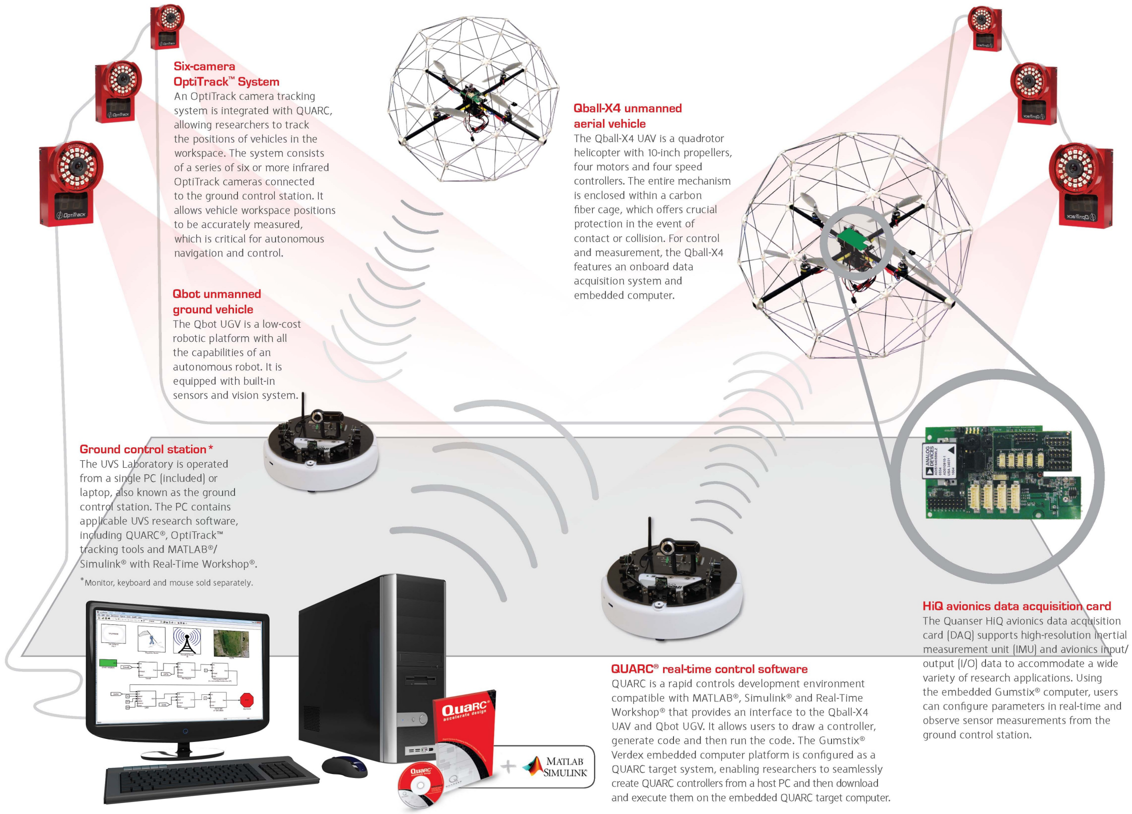


Figure 2.1: Layout of the Quanser Unmanned Vehicle Systems Lab [3].

carried out on the *quadrotor* UAV due to its great interest on both industry and academia [115,116]. A quadrotor has the ability to hover, take off and land vertically. Compared with other rotary-wing UAVs, the quadrotor is capable of having high angular acceleration since the pair of opposing motors is at the ends of the relatively long lever arms, which can generate large torques along rotation axes. The agile mobility of the quadrotor makes it suitable for conducting some complex tasks.

As shown in Figure 2.1, a Quanser Qball-X4 quadrotor is propelled by 4 brushless motors with 10-inch propellers. The carbon fiber cage encloses the crossbeam-structured quadrotor components. The cage can prevent the propellers to contact with obstacles, other vehicles and human operators, which guarantees the operation safety in the indoor environment. The motors are symmetrically mounted on the crossbeam and other components are placed at the center of the quadrotor; i.e., an embedded Gumstix microcontroller and wireless module, HiQ aerial vehicle data acquisition (DAQ) card, and batteries.

2.2.1 HiQ DAQ and Gumstix Microcontroller

HiQ DAQ, as shown in Figure 2.2, is provided by Quanser Inc. It consists of the high-resolution inertial measurement unit (IMU) and avionics input/output (I/O) card. This card is used to collect on-board sensor data and to output motor commands. By using the data from IMU including the sonar sensor, gyroscopes, accelerometers and magnetometers, the Gumstix microcontroller calculates pulse-width modulation (PWM) servo outputs for actuators according to the flight algorithms designed by users. Each motor is connected to and receives commands from a specific servo output channel integrated on HiQ DAQ. Parts of HiQ DAQ I/O components are listed in Table 2.1.

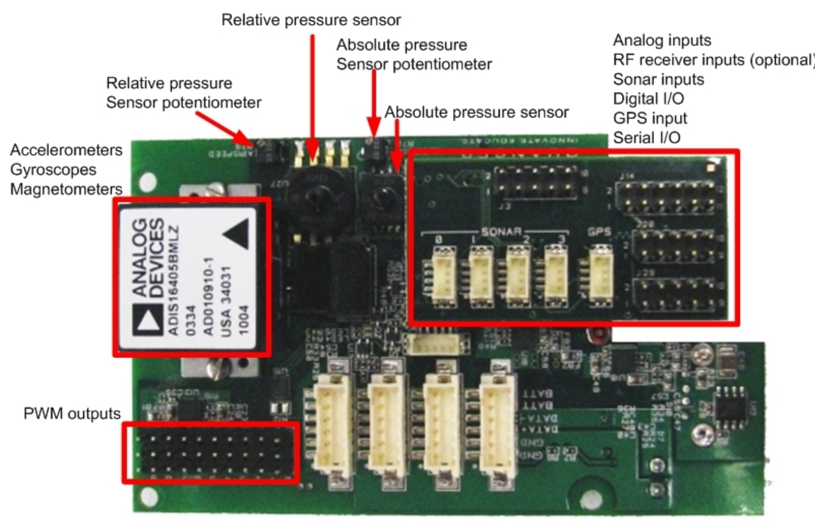


Figure 2.2: HiQ DAQ card [1].

Table 2.1: Parts of HiQ DAQ I/O [1].

Component	Description
Power input	10-20 V, 400 mA.
Gyroscope	3-axis, range configurable for ±75°/s, ±150°/s, ±300°/s, resolution 0.0125°/s/LSB at a range setting of ±75°/s.
Accelerometer	3-axis, resolution 3.33 mg/LSB.
Magnetometer	3-axis, 0.5 mGa/LSB.
Sonar input	connected to Maxbotix XL-Maxsonar-EZ3, range from 20 cm to 765 cm, resolution 1 cm.
Analog input	connected to 6 channels, 12-bit, ±3.3 V.
PWM output	10 servo motor outputs.

HiQ DAQ has a daughter-board with some general purpose I/O channels; i.e., receiver inputs, sonar inputs, and a TTL serial input used for a GPS receiver. Indeed, the daughter-board provides a convenient way for researchers to interface additional sensors and to apply the developed flight algorithms on the quadrotor.

2.2.2 Motors and Propellers

The motor, propeller and electronic speed controller (ESC) components are shown in Figure 2.3. The Quanser Qball-X4 uses E-Flite Park 480 (1020 Kv) motors [117] with paired reverse rotating propellers [118]. Two pairs of APC 10x4.7SFP propellers are employed to generate the force to lift the quadrotor. The front and rear propeller pair spins clockwise, while the left and right propeller pair spins counter-clockwise such that the net aerodynamics torque is balanced, and the torque around yaw axis is small.

Each motor is controlled by a Hobbywing Flyfun-30A ESC [119]. ESC receives PWM commands from HiQ DAQ, and then generates appropriate motor throttles. The PWM output from HiQ DAQ ranges from 1 ms (0 throttle) to 2 ms (full throttle), and thus the duty cycle is set from 5% to 10% of a 20 ms cycle.

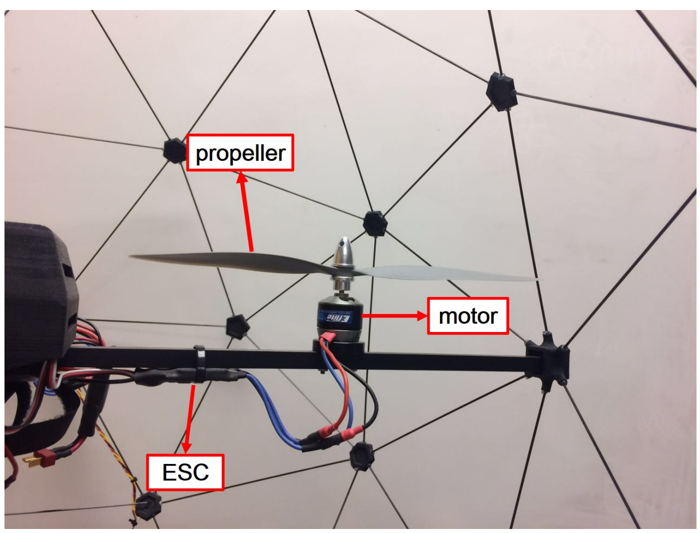


Figure 2.3: Motor, propeller and ESC.

2.2.3 Batteries

The Quanser Qball-X4 uses two 3-cell, 2500 mAh Lithium-Polymer (LiPo) batteries to power motors and HiQ DAQ card, as shown in Figure 2.4. The front and back motor pair uses one battery, and the left and right motor pair uses the other. The batteries are placed in a battery compartment beneath the crossbeam center. Due to the safety reason, it is of importance to fix the batteries firmly during the flight, and they are secured by velcro straps and battery connectors.

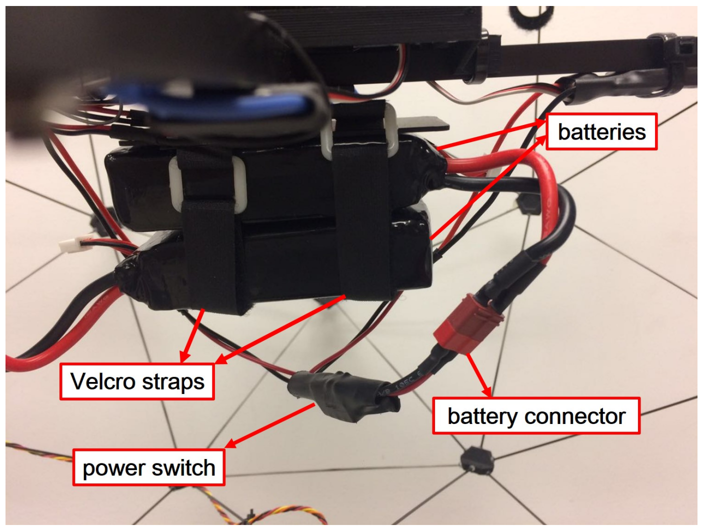


Figure 2.4: Two 2500mAh LiPo batteries.

LiPo batteries should be charged before the voltage is lower than 10 V. The HiQ DAQ card has an battery voltage input channel, and a module is designed to monitor the battery level when operating the quadrotor. Once the voltage is lower than 10.8 V, a low battery warning will be displayed such that the operator can stop the quadrotor and charge batteries. Batteries should always be charged and used in pairs. A Li-Ion/Polymer Battery Charger/Balancer is used to charge the battery with the setup of LiPo Balance/11.1 V(3S)/2.5 A. Note that when the LiPo battery is not been used for a long time, it tends to discharge itself. If the voltage drops below 3.0 V per cell, there exists a risk that the battery will not be able to be charged. In case that the low voltage is displayed on the charger, and the LiPo Balance charging mode can not be used, one can try to fix the battery by charging it with the setup of NiMH/5.0 A until the battery voltage reaches 9.1 V. Then the battery can be charged with the normal setup of LiPo Balance. To avoid battery damage, we can discharge the battery with the setup of LiPo/Storage/1.0 A/11.1 V(3S) and preserve the battery in a dry environment.

2.3 Quanser Qbot

The Quanser Qbot is a 2WMR designed by Quanser Inc. Two wheels are symmetrically mounted on an axis through the geometry center of the robot. The left and right wheels are controlled independently to have the forward or backward speed such that the motion of the robot can be controlled. Each Qbot is equipped with the iRobot Create® robotic platform, infrared sensors, sonar sensors, a Logitech Quickcam Pro 9000 USB camera, a Gumstix microcontroller and a data acquisition board. Qbot is open-architecture and suitable for researchers to add the off-the-shell sensor and to realize the designed 2WMR control algorithms. In Chapter 3 and 4, a group of Qbots are employed to verify the 2WMR rendezvous algorithms considering irregular samplings, and in Chapter 5 and 6, together with the quadrotor Qball-X4, Qbots are used for cooperative control of heterogeneous MASs. Table 2.2 illustrates some hardware configurations of the Qbot.

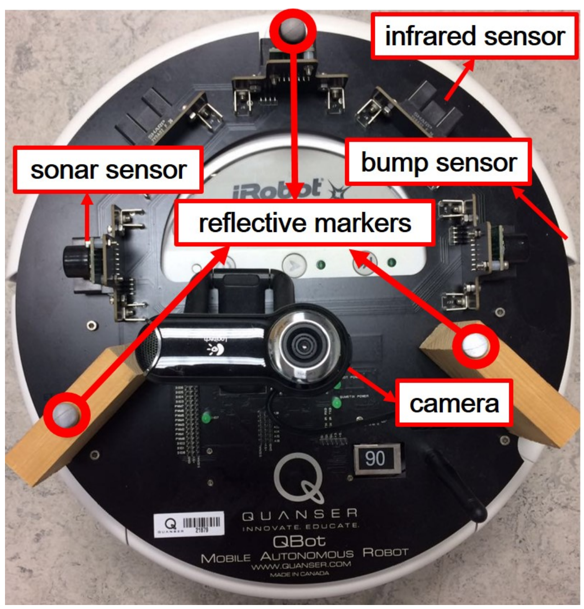


Figure 2.5: Top view of the Quanser Qbot and sensors.

Table 2.2: Parts of the Qbot components [2].

Component	Description
INT/EXT Jumper	INT, the internal iRobot Create battery; EXT, the external battery power supply.
SW/nSW jumper	SW, iRobot Create® must be switched on to receive power; nSW, iRobot Create® always draws power.
Camera	Logitech Quickcam Pro 9000 USB camera, specs in [120].
Infrared sensor	5 SHARP 2Y0A02 sensors, range from 20 cm to 150 cm.
Sonar sensor	3 MaxSonar-EZ0 sensors, range from 0 m to 6.45 m, resolution 2.54 cm.
Battery	APS 3000Ni-MH Battery/14.4 V, 3000 mAh.
DIO Pins	Digital Input/Output Pins, need to be configured as input (or output) channels.
Gumstix IR serial	ground (GND), receive (GUMSTIX IR RXD), transmit (GUMSTIX IR TXD), power (+3.3V or +5.0V).

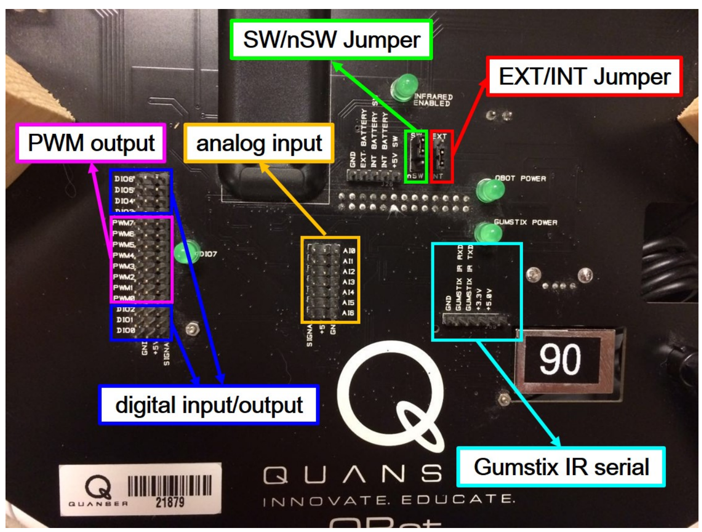


Figure 2.6: Printed circuit board of the Quanser Qbot.

Figures 2.5 and 2.6 show the available Qbot sensors and the printed circuit board respectively. There are five infrared sensors and three sonar senors placed on the top surface of the Qbot. These sensors are connected to analog input pins of the Qbot and are used to detect distances from the robot to obstacles. Bump sensors are employed to navigate the robot when a collision occurs.

2.4 QUARC®

QUARC® is a real-time control software developed by Quanser Inc., which provides a Matlab®/Simulink® interface for users to program their designed controllers for the actual hardware. QUARC® can also compile the Simulink® controller and generate the executable files for the on-board microcontrollers. When executing controllers on the Gumstix, the controller parameters can be tuned, and sensor measurements can be observed in real-time. These features make the application convenient from the user's standpoint by avoiding the complicated code writings. Some important characteristics of QUARC® are list as follows:

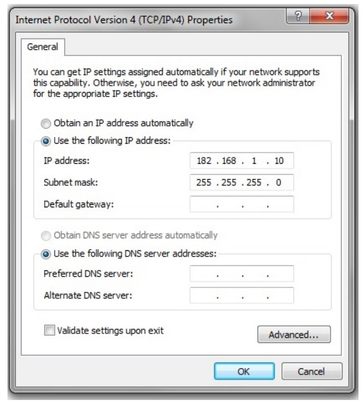
- Simple and flexible programming environment and hardware interface.
- Online parameter tuning, plotting, and other display capabilities in Simulink®.
- Code generation for multiple targets from one Simulink® block.
- Run multiple modules simultaneously on a single target.
- Support of multi-rate, multi-threaded, and asynchronous models.
- Internal communication capabilities among the multiple targets.
- Data archiving and more.

QUARC® supports various types of operating systems including Windows, QNX x86, and Linux Verdex. Controllers, communication modules are edited and complied at Host PC in Simulink® environment, and then downloaded to remote targets such as quadrotors or 2WMRs through wireless network.

2.5 Communication

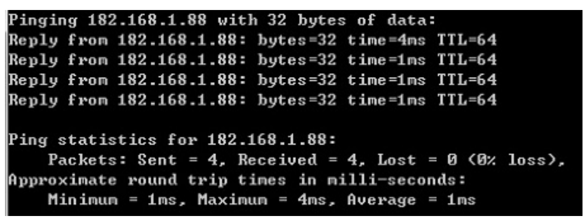
Communication among the vehicles plays an important role in the successful coordination of a multi-agent system. QUARC® provides the ad-hoc peer-to-peer wireless TCP/IP protocol for multi-vehicle information interactions. Each vehicle is configured with a predefined unique IP address. A USB wireless adapter is used on the Host PC for setting up the connection. When the power of the Qball-X4 or Qbot is turned on, the Host PC can detect a network called GSAH, as shown in Figure 2.7(a). The IP address of the Host PC is set as shown in Figure 2.7(b) and the connection between an individual vehicle and the Host PC can be checked by the command "ping {IP address of the Quanser vehicle}" in the Window Run box. Figure 2.8(a) shows an example of the successful connection with the corresponding time delays, and Figure 2.8(b) demonstrates a failed connection.

(a) Wireless ad-hoc network GSAH created by the vehicle.

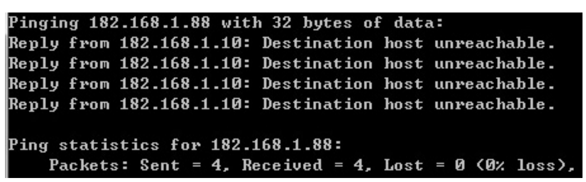


(b) Host PC IP address setup.

Figure 2.7: Wireless network setup on the Host PC.



(a) A succussful connection between the Host PC and the vehicle.



(b) A failed connection between the Host PC and the vehicle.

Figure 2.8: Communication checking results.

To download the compiled file into the Quanser vehicles, IP addresses of the targets are specified as "all linux-verdex targets" in the QUARC® setup, and the default model URI is replace by the IP address of the target vehicles, i.e., tcpip://182.168.1.88:17001 for a Quanser Qbot.

QUARC® provides communication blocksets under Simulink® environment to realize data interactions among the agents. Stream Client blocks on the Host PC and on each vehicle act as hosts to send the data, and Stream Server blocks accept data from the local host or remote host. More details of communication block configurations can be found in [1–3].

2.6 Positioning System

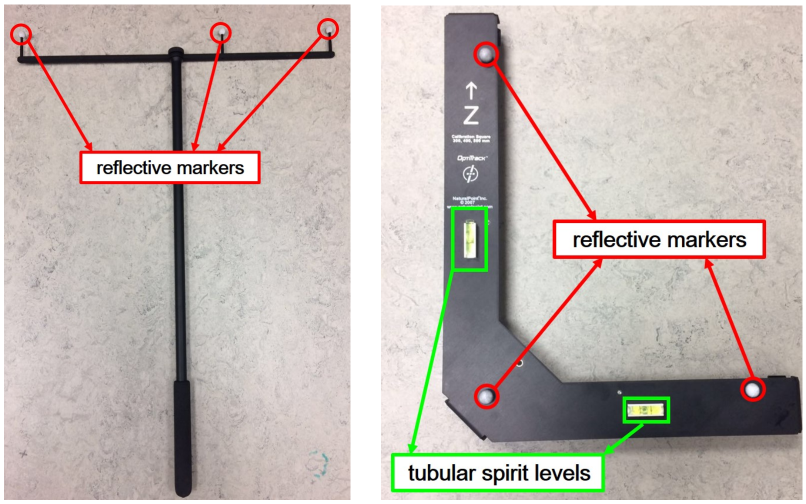
The Quanser UVS is operated in an indoor GPS-denied environment. Alternatively, the OptiTrack™ cameras developed by NaturalPoint Inc. are incorporated with the on-board IMU to provide the precise real-time position and orientation information of each vehicle, as shown in Figure 2.1. Each high-speed OptiTrack™ Flex 3 camera is with 640×480 VGA resolution and is capable to process 100 frames per second. The Precision Grayscale Mode is chosen for the camera, and then camera is only sensitive to the infrared light. The minimum number of cameras used in the Quanser UVS Lab is 3. More cameras are needed for the precise position information of the vehicles in a larger workspace. The cameras are mounted higher up along the wall such that the optimal overlapping workspace viewpoint can be achieved. In order to apply the positioning system, following steps are to be taken in sequence with the Motive software.

- Remove all objects that can reflect infrared light in the scene, and perform the camera calibration by waving the wand (Figure 2.9(a)) with three reflective markers through the workspace. The cameras capture the wand motions. When each camera captures a sufficient number of the samples, stop waving the wand, and the Motive software calculates the spatial information of the workspace.

- Position the L-shape OptiTrack™ calibration square (Figure 2.9(b)) horizontally on the ground and set the ground plane in the Motive software. The reflective marker between the two tubular spirit levels are set as the origin of the earth-fixed frame (0,0,0). x and y-axes are also defined.

- Fix a cluster reflective markers on each vehicle with the unique configuration (an example is shown in Figure 2.5), and the vehicle can be recognized and tracked as a rigid body by Flex 3 cameras in the calibrated workspace.

- Load the workspace calibration file and the rigid body definition file generated by the Motive software into the corresponding QUARC® block. The position and orientation information of the vehicle can now be obtained in the Simulink® environment.



(a) The 3-marker wand. (b) The OptiTrack™ calibration square.

Figure 2.9: Wand and L-shape OptiTrack™ calibration square.

Figure 2.10 illustrates the model receiving position information from the Host PC. 2WMRs' headings, positions of each 2WMR and quadrotor are measured from the OptiTrack™ cameras.

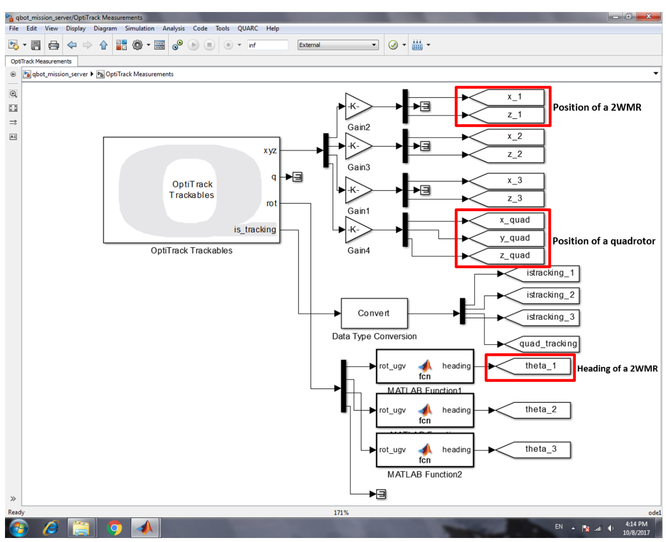


Figure 2.10: Simulink model of the experimental platform: Positioning systems.

A first-order high-pass filter that estimating the rate of change of the input is employed to estimate the quadrotor's velocity, as shown in Figure 2.11. Figure 2.12 shows the approach calculating the real-time angular velocities of a quadrotor.

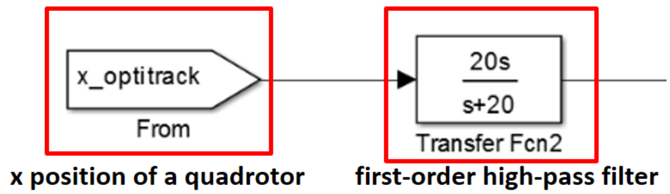


Figure 2.11: Simulink model of the experimental platform: Velocity estimation of a quadrotor.

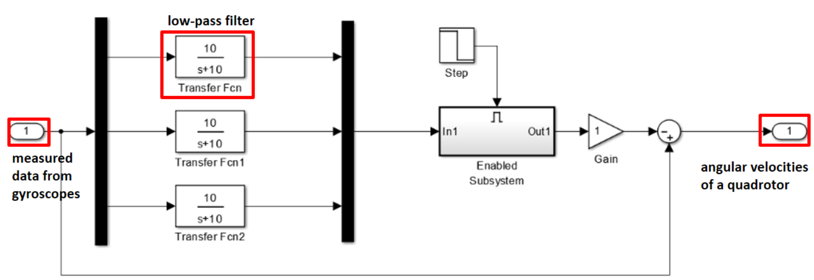


Figure 2.12: Simulink model of the experimental platform: Angular velocity calculation of a quadrotor.

Real-time roll, pitch and yaw angles of the quadrotor are calculated using the data measured from accelerometers, gyroscopes and magnetometers, as shown in Figures 2.13–2.15. The low-pass filter G_i and band-pass filter G_g are verified through experimental tests.

$$G_i = \frac{100s}{100s^2 + 20s + 1},$$
$$G_g = \frac{20s + 1}{100s^2 + 20s + 1}.$$

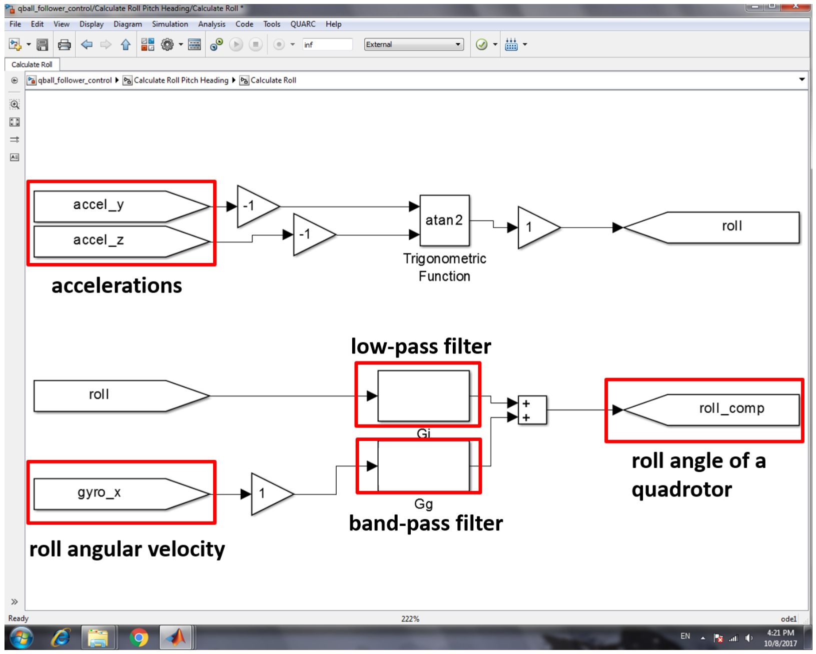


Figure 2.13: Simulink model of the experimental platform: Roll angle calculation of a quadrotor.

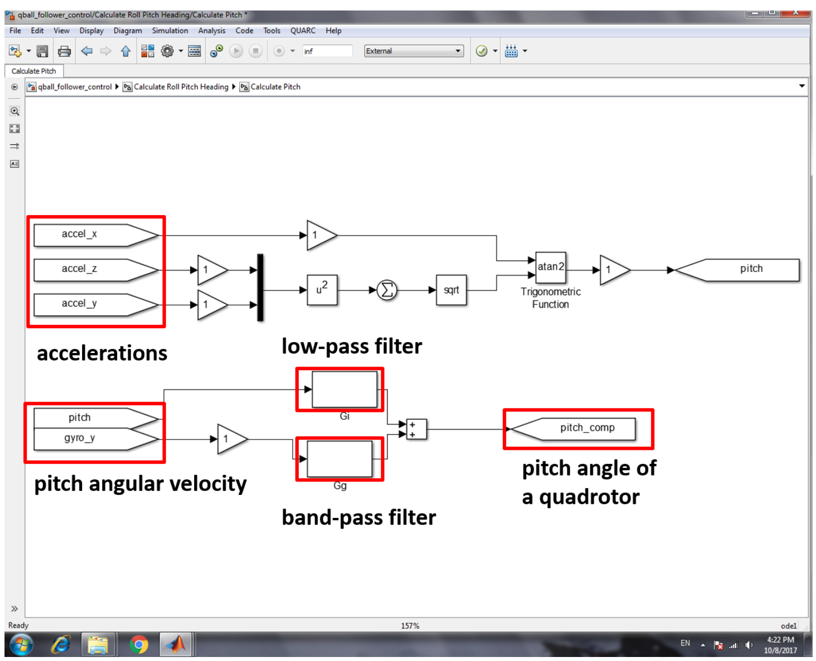


Figure 2.14: Simulink model of the experimental platform: Pitch angle calculation of a quadrotor.

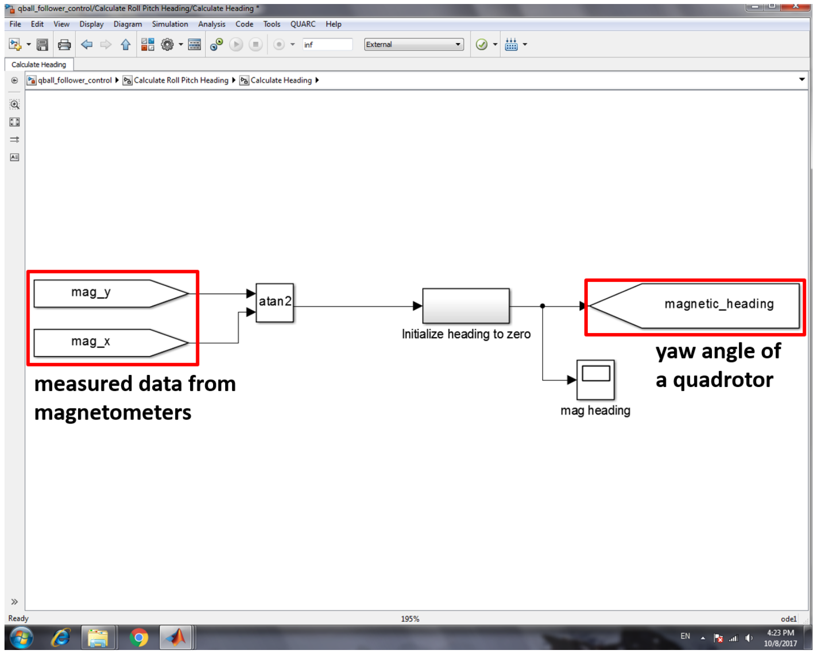


Figure 2.15: Simulink model of the experimental platform: Yaw angle calculation of a quadrotor.

2.7 Conclusion

QUARC® supports running multiple modules on a single target independently and simultaneously. Figure 2.16 shows an example of the Quanser Qbot control Simulink® modules, which consists of 6 parts: Module 1 receives the data such as the real-time position and orientation information of the robot itself from the Host PC; using the Stream Client and Server blocks, module 2 broadcasts the data to other agents, and module 3 receives the information from the networked robots. Module 4 is the black box for the data saving. In module 5, the designed control algorithm for the 2WMR is programmed in the M-file. Outputs of module 5 are the real-time wheel velocities, which are sent to module 6, the Roomba blockset to drive the vehicle.

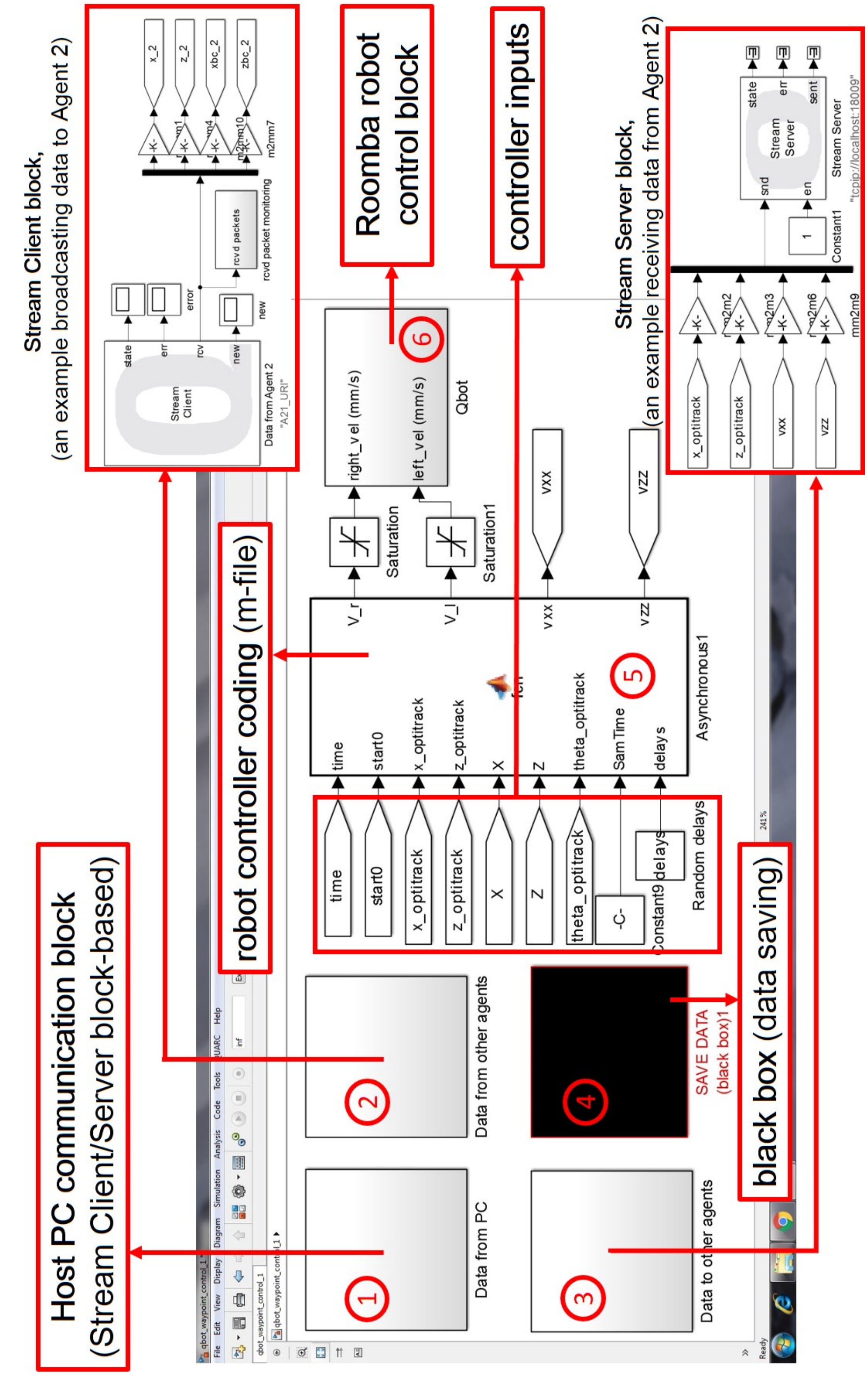


Figure 2.16: One Quanser Qbot programming in the Simulink® environment.

Chapter 3

Non-uniform Sampling Cooperative Control on a Group of Two-wheeled Mobile Robots

3.1 Introduction

Periodical samplings are employed to study the consensus problem for MASs under the sampled-data setting [114]. However, due to the existence of practical communication constraints, such as time delays and data losses, it is difficult to sample MAS states periodically. The study of the irregular sampling for cooperative control is of practical importance. Consensus protocols with non-uniform samplings are investigated in [38, 121]. In particular, the authors in [38] propose the consensus protocol for an MAS with double-integrator dynamics and non-uniform samplings. Consensus can be reached if the fixed communication topology has a spanning tree and control gains are appropriately chosen.

Most of existing works on cooperative control focus on the theoretical analysis, and the effectiveness of proposed control protocols are only verified by simulations. The application-oriented research on MASs is still at the early stage. In recent years, the control of a 2WMR attracts much attention [122, 123]. For instance, the Qbot developed by Quanser Consulting Inc. is an innovative 2WMR [2]. This mobile robot consists of two wheels in parallel, and is equipped with wireless embedded computer Gumstix and built-in sensors. With the accurate indoor global positioning system, OptiTrack$^{\mathrm{TM}}$ cameras from NaturalPoint Inc., the group of 2WMRs is ideally suited for MAS experimental studies.

Motived by the aforementioned discussion, our objective is to design and implement the consensus protocol for a group of 2WMRs. In this work, both first-order and second-order system dynamics are considered in the consensus problem with non-uniform sampling. Communication topologies in this work are assumed to be directed and switching. Since two wheels of a 2WMR are mounted on a common axis, it is impossible for a 2WMR to accelerate in the direction along the wheels' connecting axis. Such a system that has a less number of actuators than the degrees-of-freedom (DOF) is the underactuated system. We develop a *Rotate&Run Scheme* for cooperative control of a group of 2WMRs and prove that consensus can be reached with non-uniform sampling if directed graphs satisfy certain conditions.

The main contributions of this work are three-fold:

- A new *Rotate&Run Scheme* is developed to solve the consensus problem for a group of underactuated 2WMRs with non-uniform sampling. Physical constraints of the wheel velocities are considered in the design. The results on stochastic matrices are employed to study the stability of the MAS.

- A more general framework is provided compared with the existing work. The communication topology investigated in [38] is fixed and directed. However, interaction topologies are always complicated and may change dynamically in reality. In this work we consider the scenario of cooperative control of MASs under switching topologies.

- The designed algorithms are efficiently implemented on the experimental platform for solving the consensus problem.

The remaining part of the work are organized as follows. Section 3.2 and 3.3 introduce some preliminaries and formulate the problems. The main theoretical results on the consensus analysis are presented in Section 3.4. In Section 3.5, experimental results on the real platform are provided to verify the effectiveness of the proposed methods. Section 3.6 concludes this work.

Notation: The superscript 'T' represents the matrix transpose. $\mathbb{R}$ represents the space of the real numbers. $\mathbf{1}_N$ and $\mathbf{0}_N$ denote vectors $[1, 1, \dots, 1]^{\mathrm{T}} \in \mathbb{R}^N$ and $[0, 0, \dots, 0]^{\mathrm{T}} \in \mathbb{R}^N$ respectively. $\mathbf{0}_{N\times N}$ represents a zero matrix with the dimension N by N. $I_N \in \mathbb{R}^{N\times N}$ is the identity matrix. $\mathrm{diag}\{y_1, y_2, \dots, y_n\}$ is an $n \times n$ matrix with diagonal entries $y_1, y_2, \dots, y_n$. The notations will be used throughout the thesis.

3.2 Preliminaries

We first present some basic knowledge in algebraic graph theory and nonnegative matrices.

A directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ with N nodes is used to model the communication topology over the network, where $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ represents the vertex set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the edge set and $A = [a_{ij}] \in \mathbb{R}^{N \times N}$, $(a_{ij} \geq 0, \forall i, j = 1, 2, \ldots, N)$ is the adjacency matrix. Each agent is represented by a node in $\mathcal{V}$. An edge $e_{ij} = (v_j, v_i)$ denotes that agent i receives information from agent j. $a_{ij} > 0$ if the corresponding edge $e_{ij} \in \mathcal{E}$, and agent j is called the neighbor of agent i. $a_{ij} = 0$ if $e_{ij} \notin \mathcal{E}$. It is assumed that there is no information interaction from an agent to itself, meaning that $a_{ii} = 0$. $\mathcal{N}_i = \{v_j \in \mathcal{V} : (v_j, v_i) \in \mathcal{E}\}$ denotes the set of neighbors of agent i. A path that from a vertex v_i to the other vertex v_j is an ordered sequence of edges (v_i, v_{m_1}), (v_{m_1}, v_{m_2}),..., (v_{m_p}, v_j), where each of the edges is in $\mathcal{E}$. If there exists a node v_i in the graph that a directed path from that node to any other node exists, the graph has a directed spanning tree rooted at v_i. For more details, see [23] and references therein. The *graph Laplacian* $L = [l_{ij}] \in \mathbb{R}^{N \times N}$ is defined as:

$$l_{ij} = -a_{ij}, \ \forall i \neq j; \ l_{ii} = \sum_{j=1, j \neq i}^{N} a_{ij}.$$

Matrix $A \in \mathbb{R}^{n \times n}$ is nonnegative if all the entries of A are nonnegative, i.e., $A \geqslant 0$. We denote $A \geqslant B$ if $A - B \geqslant 0$. A stochastic matrix is defined as a nonnegative square matrix with each row sum equal to 1. A matrix A is called stochastic indecomposable and aperiodic (SIA) if $\lim_{k \to \infty} A^k = \mathbf{1}_n c^{\mathrm{T}}$, where c is an $n \times 1$ vector of constants.

The knowledge of directed graph and stochastic matrix will be used throughout the works in Chapter 3, Chapter 5 and Chapter 6.

3.3 Problem Statement and System Description

3.3.1 Consensus of the MAS with First-order Dynamics

When operating the 2WMR in a two-dimensional plane, axes of the workspace are arranged as given in [2]; see the top view in Figure 3.1. This is the workspace frame used by the 2WMR controller. The heading θ is the angle between the direction in which the 2WMR's nose is pointing and the positive direction of x-axis. In this work, the heading is measured anti-clockwise from the positive direction of x-axis, in degrees from $-180°$ to $180°$. The coordinate system of the 2WMR will be used throughout the thesis.

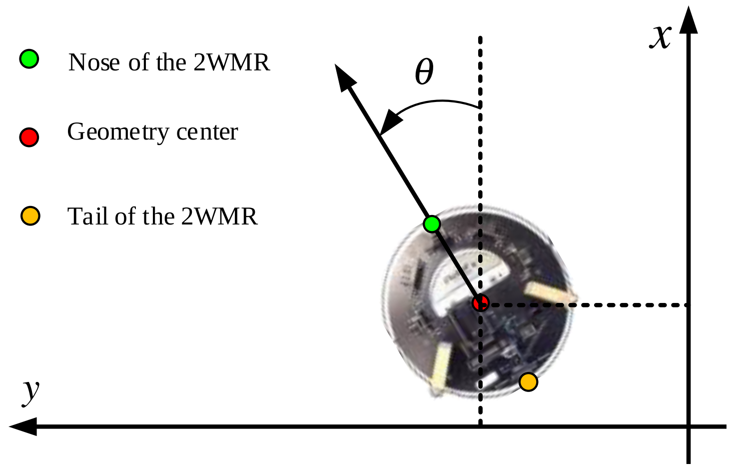


Figure 3.1: Top view of the coordinate systems for the 2WMR.

If we assume that motions along x and y-axes are decoupled, the system dynamics in x-axis and y-axis can be modeled as

$$\begin{cases} \dot{\hat{x}}_i(t) = u_{\hat{x}_i}(t), \\ \dot{\hat{y}}_i(t) = u_{\hat{y}_i}(t), \;\; i = 1, 2, \ldots, N, \end{cases} \tag{3.1}$$

where $\hat{x}_i(t), \hat{y}_i(t) \in \mathbb{R}$ are the position information, and $u_{\hat{x}_i}(t), u_{\hat{y}_i}(t) \in \mathbb{R}$ are the control inputs along x-axis and y-axis, respectively.

In this work, all agents sample states simultaneously, and the sampling is not periodical. Let $\{t_k, k = 0, 1, \ldots\}$ be a sequence consisting of sampling time instants. It is supposed that sampling periods are integer multiple of the smallest sampling period h and are from a finite set, i.e., $m_k h = t_{k+1} - t_k \in \Gamma = \{n_1, n_2, \ldots, n_\tau\}h, \; \forall k \geqslant 0$, where n_i is an integer.

Discretizing the dynamics in (3.1) by using the sampled-data setup and the zero-order hold, we have

$$\begin{cases} \hat{x}_i(k+1) = \hat{x}_i(k) + m_k h u_{\hat{x}_i}(k), \\ \hat{y}_i(k+1) = \hat{y}_i(k) + m_k h u_{\hat{y}_i}(k), \end{cases}$$

where $\hat{x}_i(k), \; \hat{y}_i(k) \in \mathbb{R}$, $u_{\hat{x}_i}(k), \; u_{\hat{y}_i}(k) \in \mathbb{R}$ are the position and control input of agent

i at time instant k. The system in (3.1) is then rewritten as

$$\begin{cases} \hat{x}(k+1) = \hat{x}(k) + m_k h u_{\hat{x}}(k), \\ \hat{y}(k+1) = \hat{y}(k) + m_k h u_{\hat{y}}(k) \end{cases}$$

with $\hat{x}(k) = [\hat{x}_1(k), \hat{x}_2(k), \ldots, \hat{x}_N(k)]^{\mathrm{T}}$,
$\hat{y}(k) = [\hat{y}_1(k), \hat{y}_2(k), \ldots, \hat{y}_N(k)]^{\mathrm{T}}$,
$u_{\hat{x}}(k) = [u_{\hat{x}_1}(k), u_{\hat{x}_2}(k), \ldots, u_{\hat{x}_N}(k)]^{\mathrm{T}}$,
$u_{\hat{y}}(k) = [u_{\hat{y}_1}(k), u_{\hat{y}_2}(k), \ldots, u_{\hat{y}_N}(k)]^{\mathrm{T}}$.
If we apply the following control protocol to the decoupled system dynamics in (3.1):

$$\begin{cases} u_{\hat{x}_i}(t) = -\beta_k \sum_{j=1}^{N} a_{ij}(t_k)[\hat{x}_i(t_k) - \hat{x}_j(t_k)], \\ u_{\hat{y}_i}(t) = -\beta_k \sum_{j=1}^{N} a_{ij}(t_k)[\hat{y}_i(t_k) - \hat{y}_j(t_k)], \ t_k \leqslant t < t_{k+1}. \end{cases} \tag{3.2}$$

The state evolution of the system in (3.1) is further written as

$$\begin{cases} \hat{x}(t_{k+1}) = (I_N - m_k h \beta_k L(t_k))\hat{x}(t_k), \\ \hat{y}(t_{k+1}) = (I_N - m_k h \beta_k L(t_k))\hat{y}(t_k). \end{cases} \tag{3.3}$$

Note that the trajectory control of the 2WMR can be realized by varying the velocity of each wheel. As shown in Figure 3.2, the body frame axes consist of x'-axis and y'-axis which are pointing in the forward direction of the 2WMR and to the left along the common axis of two wheels. Define v_l and v_r as the left and right wheel velocities parallel to x'-axis. By realizing the fact that as an underactuated system, the 2WMR cannot generate the independent velocity along y'-axis, and the design of the control protocol in (3.2) needs to be modified. Instead of controlling the decoupled system dynamics in (3.1), we propose a *Rotate&Run Scheme* based on the control protocol in (3.2) to control the 2WMR in the two-dimensional plane.

The body frame axes divide the plane into four infinite regions according to the signs of the two coordinates $(x',\ y')$: Quadrant I $(+,\ -)$; II $(+,\ +)$; III $(-,\ +)$ and IV $(-,\ -)$; see Figure 3.2, and the shaded region represents Quadrant I. The *Rotate&Run Algorithm* for each agent to update the states is illustrated in **Algorithm 3.1**. At the sampling time instant t_k, we calculate the goal heading for agent i, and determine the rotation direction. After rotating in place until it aims at the calculated direction, agent i then moves in a straight line toward $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$ at the velocity $v = \sqrt{(\hat{u}_{x_i}(t_k))^2 + (\hat{u}_{y_i}(t_k))^2}$ if $v < v_{\max}$ otherwise $v = v_{\max}$ until the next sampling time instant.

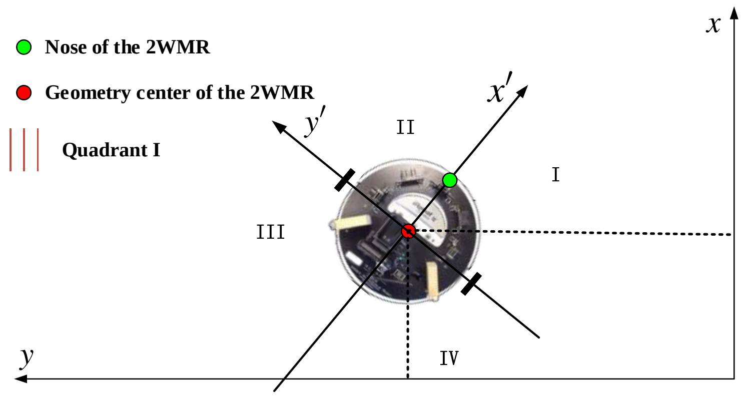


Figure 3.2: Body frame axes of the 2WMR.

Note that if we apply **Algorithm 3.1** to a group of 2WMRs, the time interval $[t_k, t_{k+1})$ includes $[t_k, t_k + \Delta T)$ for the 2WMR to rotate around the midpoint of the wheel axis and $[t_k + \Delta T, t_{k+1})$ for the 2WMR to move toward $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$. By adopting this algorithm and considering the physical constraints of the wheel velocities $\|v_l\| \leqslant v_{max}$ and $\|v_r\| \leqslant v_{max}$, it is readily shown that the maximum rotation angle for the 2WMR is $\pm 90°$ to ensure that either the nose or the tail points at $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$. With the assumption that $\Delta T \leqslant t_{k+1} - t_k, \forall k = 0, 1, 2, \ldots,$ x and y position evolutions of agent i, $(x_i(t_{k+1}),\ y_i(t_{k+1}))$ at the time instant t_{k+1} is supposed to be on the line segment from $(x_i(t_k),\ y_i(t_k))$ to $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$. It is noticed that $\|\frac{x_i(t_{k+1}) - x_i(t_k)}{\hat{x}_i(t_{k+1}) - x_i(t_k)}\| = \|\frac{y_i(t_{k+1}) - y_i(t_k)}{\hat{y}_i(t_{k+1}) - y_i(t_k)}\| = \lambda_i(t_k), \forall i = 1, 2, \ldots, N$, where $\lambda_i(t_k) \in (0, 1]$ is a constant. Let $x(t_k) = [x_1(t_k), x_2(t_k), \ldots, x_N(t_k)]^{\mathrm{T}}$ and $y(t_k) = [y_1(t_k), y_2(t_k), \ldots, y_N(t_k)]^{\mathrm{T}}$, after some algebraic manipulation, we get

$$\begin{cases} x(t_{k+1}) = \Psi_k x(t_k), \\ y(t_{k+1}) = \Psi_k y(t_k), \end{cases} \tag{3.4}$$

where $\Psi_k = (I_N - \lambda_k m_k h \beta_k L(t_k))$ and $\lambda_k = \mathrm{diag}\{\lambda_1(t_k), \lambda_2(t_k), \ldots, \lambda_N(t_k)\}$. The consensus property of (3.4) will be analyzed in Section 3.4.

Algorithm 3.1 The *Rotate&Run Algorithm* for agent i with the first-order dynamics ($i = 1, 2 \dots N$.)

Input: $x_i(t_k),\ y_i(t_k),\ \theta_i(t_k),\ x_j(t_k),\ y_j(t_k),\ \forall j \in \mathcal{N}_i(t_k)$.
Output: $v_l(t),\ v_r(t),\ t_k \leqslant t < t_{k+1}$.

if $t = t_k$ **then**
  Choose the value of β_k satisfying the condition in **Lemma 3.1**.
  Set $\hat{x}_i(t_k) = x_i(t_k),\ \hat{y}_i(t_k) = y_i(t_k)$, and calculate $\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1})$ using (3.3), $\theta_{iT} = \mathrm{atan2}((\hat{y}_i(t_{k+1}) - y_i(t_k)),\ (\hat{x}_i(t_{k+1}) - x_i(t_k)))$, $v_i = \sqrt{(u_{\hat{x}_i}(t_k))^2 + (u_{\hat{y}_i}(t_k))^2}$, for the case $\sqrt{(u_{\hat{x}_i}(t_k))^2 + (u_{\hat{y}_i}(t_k))^2} < v_{\max}$, otherwise $v_i = v_{\max}$.
  if $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$ is in Quadrant I, **then**
    Rd= 1, Md= 1, $\theta_{id}(t_k) = \theta_{iT}$;
  else if $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$ is in Quadrant II, **then**
    Rd= -1, Md=1, $\theta_{id}(t_k) = \theta_{iT}$;
  else if $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$ is in Quadrant III, **then**
    Rd= 1, Md= -1;
    $\theta_{id}(t_k) = \theta_{iT} \pm 180°,\ \theta_{id}(t_k) \in [-180°, 180°)$;
  else
    Rd= -1, Md= -1;
    $\theta_{id}(t_k) = \theta_{iT} \pm 180°,\ \theta_{id}(t_k) \in [-180°, 180°)$.
  end if
end if
while $t_k \leqslant t < t_{k+1}$ **do**
  if $|\theta_i(t) - \theta_{id}(t_k)| > \varepsilon$ **then**
    $v_l = \bar{v}$, $v_r = -\bar{v}$ for the case Rd= 1 and
    $v_l = -\bar{v}$, $v_r = \bar{v}$ for the case Rd= -1.
  else
    $v_l = v_i$, $v_r = v_i$ for the case Md= 1 and
    $v_l = -v_i$, $v_r = -v_i$ for the case Md= -1.
  end if
end while

Rd= 1: Rotating clockwise,
Rd= -1: Rotating counterclockwise,
Md= 1: Moving forward when the nose of the 2WMR is pointing at $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$,
Md= -1: Moving backward when the tail of the 2WMR is pointing at $(\hat{x}_i(t_{k+1}),\ \hat{y}_i(t_{k+1}))$,
$x_i(t_k),\ y_i(t_k),\ \theta_i(t_k)$ are x, y positions and heading of 2WMR i at instant t_k,
ε is the tolerance for the heading,
v_l and v_r are the left and right wheel velocities,
$\bar{v}$ satisfying $0 \leqslant \bar{v} \leqslant v_{max}$ is the constant wheel velocity for the 2WMR to rotate in place.

3.3.2 Consensus of the MAS with Second-order Dynamics

In this subsection, we propose a second-order consensus protocol that incorporates both the position and velocity feedback into the controller design for a group of 2WMRs. First we consider the following decoupled system dynamics

$$\begin{cases} \dot{\hat{x}}_i(t) = v_{\hat{x}_i}(t),\ \dot{v}_{\hat{x}_i}(t) = u_{\hat{x}_i}(t), \\ \dot{\hat{y}}_i(t) = v_{\hat{y}_i}(t),\ \dot{v}_{\hat{y}_i}(t) = u_{\hat{y}_i}(t), \end{cases} \tag{3.5}$$

where $\hat{x}_i(t),\ \hat{y}_i(t),\ v_{\hat{x}_i}(t),\ v_{\hat{y}_i}(t),\ u_{\hat{x}_i}(t),\ u_{\hat{y}_i}(t) \in \mathbb{R}$ are the position, velocity information and control inputs along x-axis and y-axis. With the non-uniform and synchronous sampling, we apply a zero-order hold for the control inputs $u_{\hat{x}_i}(t)$ and $u_{\hat{y}_i}(t)$. Eq. (3.5) is then discretized as

$$\begin{cases} \hat{x}_i(k+1) = \dfrac{1}{2} m_k^2 h^2 u_{\hat{x}_i}(k) + m_k h v_{\hat{x}_i}(k) + \hat{x}_i(k), \\ v_{\hat{x}_i}(k+1) = v_{\hat{x}_i}(k) + m_k h u_{\hat{x}_i}(k), \\ \hat{y}_i(k+1) = \dfrac{1}{2} m_k^2 h^2 u_{\hat{y}_i}(k) + m_k h v_{\hat{y}_i}(k) + \hat{y}_i(k), \\ v_{\hat{y}_i}(k+1) = v_{\hat{y}_i}(k) + m_k h u_{\hat{y}_i}(k). \end{cases} \tag{3.6}$$

Define

$$\begin{aligned} \hat{x}(k) &= [\hat{x}_1(k), \hat{x}_2(k), \ldots, \hat{x}_N(k)]^{\mathrm{T}}, \\ \hat{y}(k) &= [\hat{y}_1(k), \hat{y}_2(k), \ldots, \hat{y}_N(k)]^{\mathrm{T}}, \\ v_{\hat{x}}(k) &= [u_{\hat{x}_1}(k), v_{\hat{x}_2}(k), \ldots, v_{\hat{x}_N}(k)]^{\mathrm{T}}, \\ v_{\hat{y}}(k) &= [u_{\hat{y}_1}(k), v_{\hat{y}_2}(k), \ldots, v_{\hat{y}_N}(k)]^{\mathrm{T}}. \\ u_{\hat{x}}(k) &= [u_{\hat{x}_1}(k), u_{\hat{x}_2}(k), \ldots, u_{\hat{x}_N}(k)]^{\mathrm{T}}, \\ u_{\hat{y}}(k) &= [u_{\hat{y}_1}(k), u_{\hat{y}_2}(k), \ldots, u_{\hat{y}_N}(k)]^{\mathrm{T}}. \end{aligned}$$

Then

$$\begin{cases} \begin{bmatrix} \hat{x}(k+1) \\ v_{\hat{x}}(k+1) \end{bmatrix} = \begin{bmatrix} I_N & m_k h I_N \\ 0 & I_N \end{bmatrix} \begin{bmatrix} \hat{x}(k) \\ v_{\hat{x}}(k) \end{bmatrix} + \begin{bmatrix} \frac{1}{2} m_k^2 h^2 I_N \\ m_k h I_N \end{bmatrix} u_{\hat{x}}(k), \\ \begin{bmatrix} \hat{y}(k+1) \\ v_{\hat{y}}(k+1) \end{bmatrix} = \begin{bmatrix} I_N & m_k h I_N \\ 0 & I_N \end{bmatrix} \begin{bmatrix} \hat{y}(k) \\ v_{\hat{y}}(k) \end{bmatrix} + \begin{bmatrix} \frac{1}{2} m_k^2 h^2 I_N \\ m_k h I_N \end{bmatrix} u_{\hat{y}}(k). \end{cases} \tag{3.7}$$

We propose the following consensus protocol for the second-order decoupled system dynamics:

$$\begin{cases} u_{\hat{x}_i}(t) = -\alpha_k v_{\hat{x}_i}(t_k) - \beta_k \sum_{j=1}^{N} a_{ij}(t_k)[\hat{x}_i(t_k) - \hat{x}_j(t_k)], \\ u_{\hat{y}_i}(t) = -\alpha_k v_{\hat{y}_i}(t_k) - \beta_k \sum_{j=1}^{N} a_{ij}(t_k)[\hat{y}_i(t_k) - \hat{y}_j(t_k)], \\ \quad t_k \leqslant t < t_{k+1}. \end{cases} \tag{3.8}$$

The system in (3.7) can be rewritten as

$$\begin{cases} \begin{bmatrix} \hat{x}(t_{k+1}) \\ v_{\hat{x}}(t_{k+1}) \end{bmatrix} = \hat{\Pi}_k \begin{bmatrix} \hat{x}(t_k) \\ v_{\hat{x}}(t_k) \end{bmatrix}, \\ \begin{bmatrix} \hat{y}(t_{k+1}) \\ v_{\hat{y}}(t_{k+1}) \end{bmatrix} = \hat{\Pi}_k \begin{bmatrix} \hat{y}(t_k) \\ v_{\hat{y}}(t_k) \end{bmatrix}, \end{cases} \tag{3.9}$$

where

$$\hat{\Pi}_k = \begin{bmatrix} I_N - \frac{1}{2}m_k^2 h^2 \beta_k L(t_k) & (m_k h - \frac{1}{2}\alpha_k m_k^2 h^2) I_N \\ -\beta_k m_k h L(t_k) & (1 - \alpha_k m_k h) I_N \end{bmatrix}.$$

The control protocol in (3.8) can not be applied to the 2WMR because the underactuation characteristic of the system dynamics may result in the failure of the control. Let $x_i(t_k), v_{x_i}(t_k), y_i(t_k), v_{y_i}(t_k), \theta_i(t_k)$ be x position, x velocity, y position, y velocity and heading of 2WMR i at t_k. We propose **Algorithm 3.2** for each 2WMR with the second-order dynamics to update the states: 1) Set $\hat{x}_i(t_k) = x_i(t_k),\ \hat{y}_i(t_k) = y_i(t_k)$, $v_{\hat{x}_i}(t_k) = v_{x_i}(t_k)$ and $v_{\hat{y}_i}(t_k) = v_{y_i}(t_k)$; choose the values of control gains α_k and β_k satisfying the conditions in **Lemma 3.5**, and then calculate $v_{\hat{x}_i}(t_{k+1}),\ v_{\hat{y}_i}(t_{k+1})$ using (3.9); 2) the 2WMR rotates in place until the nose or the tail aims at the goal orientation $\theta_{iT} = \text{atan2}(v_{\hat{y}_i}(t_{k+1}), v_{\hat{x}_i}(t_{k+1}))$; 3) the vehicle moves along the orientation at the velocity $v = \sqrt{(v_{\hat{x}_i}(t_{k+1}))^2 + (v_{\hat{y}_i}(t_{k+1}))^2}$ if $v < v_{\max}$ otherwise $v = v_{\max}$ until the next sampling time instant.

The sampling period at the time instant t_k is expected to be $m_k h = t_{k+1} - t_k$ for

Algorithm 3.2 The Rotate&Run algorithm for a group of Qots with the second-order and two-dimensional dynamics

Input: $x_i(t_k), v_{x_i}(t_k), y_i(t_k), v_{y_i}(t_k), \theta_i(t_k),\ x_j(t_k), v_{x_j}(t_k), y_j(t_k), v_{y_j}(t_k),\ \forall j \in \mathcal{N}_i(t_k)$.
Output: $v_l(t), v_r(t),\ t_k \leqslant t < t_{k+1}$.

if $t = t_k$ **then**
    Set $\hat{x}_i(t_k) = x_i(t_k),\ \hat{y}_i(t_k) = y_i(t_k),\ v_{\hat{x}_i}(t_k) = v_{x_i}(t_k)$ and $v_{\hat{y}_i}(t_k) = v_{y_i}(t_k)$; choose the value of α_k and β_k satisfying the conditions in **Lemma 3.5**.
    Calculate $v_{\hat{x}_i}(t_{k+1}),\ v_{\hat{y}_i}(t_{k+1})$ using (3.9),
    $\theta_{iT} = \mathrm{atan2}(v_{\hat{y}_i}(t_{k+1}), v_{\hat{x}_i}(t_{k+1}))$,
    $v_i = \sqrt{(v_{\hat{x}_i}(t_{k+1}))^2 + (v_{\hat{y}_i}(t_{k+1}))^2}$,
    for the case $v = \sqrt{(v_{\hat{x}_i}(t_{k+1}))^2 + (v_{\hat{y}_i}(t_{k+1}))^2} < v_{\max}$, otherwise $v_i = v_{\max}$.
    if $(x_i(t_k) + v_{\hat{x}_i}(t_{k+1}),\ y_i(t_k) + v_{\hat{y}_i}(t_{k+1}))$ is in Quadrant I, **then**
        Rd$=1$, Md$=1$, $\theta_{id}(t_k) = \theta_{iT}$;
    else if $(x_i(t_k) + v_{\hat{x}_i}(t_{k+1}),\ y_i(t_k) + v_{\hat{y}_i}(t_{k+1}))$ is in Quadrant II, **then**
        Rd$=-1$, Md$=1$, $\theta_{id}(t_k) = \theta_{iT}$;
    else if $(x_i(t_k) + v_{\hat{x}_i}(t_{k+1}),\ y_i(t_k) + v_{\hat{y}_i}(t_{k+1}))$ is in Quadrant III, **then**
        Rd$=1$, Md$=-1$;
        $\theta_{id}(t_k) = \theta_{iT} \pm 180°,\ \theta_{id}(t_k) \in [-180°, 180°)$;
    else
        Rd$=-1$, Md$=-1$;
        $\theta_{id}(t_k) = \theta_{iT} \pm 180°,\ \theta_{id}(t_k) \in [-180°, 180°)$.
    end if
end if
while $t_k \leqslant t < t_{k+1}$ **do**
    if $|\theta_i(t) - \theta_{id}(t_k)| > \varepsilon$ **then**
        $v_l = \bar{v}$, $v_r = -\bar{v}$ for the case Rd$=1$ and
        $v_l = -\bar{v}$, $v_r = \bar{v}$ for the case Rd$=-1$.
    else
        $v_l = v_i$, $v_r = v_i$ for the case Md$=1$ and
        $v_l = -v_i$, $v_r = -v_i$ for the case Md$=-1$.
    end if
end while

Rd$=1$: Rotating clockwise,
Rd$=-1$: Rotating counterclockwise,
Md$=1$: Moving forward with the velocity v, satisfying $\|\frac{v_{x_i}(t)}{v_{y_i}(t)}\| = \|\frac{v_{\hat{x}_i}(t_{k+1})}{v_{\hat{y}_i}(t_{k+1})}\|$,
Md$=-1$: Moving backward with the velocity v, satisfying $\|\frac{v_{x_i}(t)}{v_{y_i}(t)}\| = \|\frac{v_{\hat{x}_i}(t_{k+1})}{v_{\hat{y}_i}(t_{k+1})}\|$,
ε is the tolerance for the heading,
$\bar{v}$ satisfying $0 \leqslant \bar{v} \leqslant v_{max}$ is the constant wheel velocity for the 2WMR to rotate in place.

updating the velocity. Using (3.9), we have

$$\begin{cases} v_{\hat{x}_i}(t_{k+1}) = v_{\hat{x}_i}(t_k) + m_k h u_{\hat{x}_i}(t_k), \\ v_{\hat{y}_i}(t_{k+1}) = v_{\hat{y}_i}(t_k) + m_k h u_{\hat{y}_i}(t_k). \end{cases}$$

However, since the time interval $[t_k, t_{k+1})$ consists of the time interval for the 2WMR to rotate in place to meet the required orientation and the time interval for the 2WMR to update the positions along x and y-axes, the actual velocities along x and y-axes of agent i at the time instant t_{k+1} are

$$\begin{cases} v_{x_i}(t_{k+1}) = v_{x_i}(t_k) + \rho_i(t_k) m_k h u_{\hat{x}_i}(t_k), \\ v_{y_i}(t_{k+1}) = v_{y_i}(t_k) + \rho_i(t_k) m_k h u_{\hat{y}_i}(t_k), \end{cases}$$

where $\rho_i(t_k) \in (0, 1]$. Let $x(t_k) = [x_1(t_k), x_2(t_k), \ldots, x_N(t_k)]^{\mathrm{T}}$, $v_x(t_k) = [v_{x_1}(t_k), v_{x_2}(t_k), \ldots, v_{x_N}(t_k)]^{\mathrm{T}}$, $y(t_k) = [y_1(t_k), y_2(t_k), \ldots, y_N(t_k)]^{\mathrm{T}}$ and $v_y(t_k) = [v_{y_1}(t_k), v_{y_2}(t_k), \ldots, v_{y_N}(t_k)]^{\mathrm{T}}$, we have

$$\begin{cases} \begin{bmatrix} x(t_{k+1}) \\ v_x(t_{k+1}) \end{bmatrix} = \Pi_k \begin{bmatrix} x(t_k) \\ v_x(t_k) \end{bmatrix}, \\ \begin{bmatrix} y(t_{k+1}) \\ v_y(t_{k+1}) \end{bmatrix} = \Pi_k \begin{bmatrix} y(t_k) \\ v_y(t_k) \end{bmatrix}, \end{cases} \tag{3.10}$$

where

$$\begin{aligned} \Pi_k &= \begin{bmatrix} I_N - \frac{1}{2}(\rho_k m_k h)^2 \beta_k L(t_k) & \rho_k m_k h - \frac{1}{2}\alpha_k (\rho_k m_k h)^2 \\ -\beta_k \rho_k m_k h L(t_k) & I_N - \alpha_k \rho_k m_k h \end{bmatrix}, \\ \rho_k &= \mathrm{diag}\{\rho_1(t_k), \rho_2(t_k), \ldots, \rho_N(t_k)\}. \end{aligned} \tag{3.11}$$

3.4 Main Results

In this section, we present the main results on the conditions of ensuring consensus.

Consensus Analysis for the MAS with First-order Dynamics

Consider **Algorithm 3.1** for the MAS, we have the following lemma.

Lemma 3.1. *There exist $\beta_k, k = 1, 2 \ldots$ such that the corresponding Ψ_k in (3.4) are stochastic matrices.*

Proof. If we choose β_k such that $0 < \beta_k < \min_i \frac{1}{\lambda_i(t_k) m_k h \sum_{j \in \mathcal{N}_i} a_{ij}(t_k)}$, $\forall i = 1, 2, \ldots, N$. It is readily to show that Ψ_k are nonnegative matrices and the diagonal elements are positive. □

Lemma 3.2 (Lemma 3.3, Corollary 3.5 & Lemma 3.7 in [29]). *If the stochastic matrix $M \in \mathbb{R}^{n \times n}$ has positive diagonal elements, and the associated directed graph has a spanning tree, then M is SIA.*

Lemma 3.3 ([124]). *Let $\{S_1, S_2, \ldots, S_k\}$ be a set consisting of finite SIA matrices with the same dimension $n \times n$. Any sequence of matrix product $S_{i_m} S_{i_{m-1}} \ldots S_1$ with arbitrary length is SIA. Moreover, for the product $S_{i_m} S_{i_{m-1}} \ldots$ with infinite length, there exists a column vector c such that*

$$\lim_{m \to \infty} S_{i_m} S_{i_{m-1}} \ldots S_1 = \mathbf{1}_n c^{\mathrm{T}}.$$

The following theorem presents the consensus condition for a group of agents with first-order dynamics.

Theorem 3.1. *Assume that $\beta_k, k = 1, 2 \ldots$ satisfy the condition in **Lemma 3.1**. Consensus of the group of 2WMRs with the first-order dynamics and non-uniform sampling can be reached if the switching directed graphs $\mathcal{G}(t_k), k = 1, 2 \ldots$ have a spanning tree at each sampling time instant.*

Proof. From the analysis in the previous section, it is shown that by varying the rotating velocity $\bar{v}$ of the wheels, we can change the length of the rotating time $\Delta T = \frac{|\theta_i(t_k) - \theta_{id}(t_k)| \pi d}{360 \bar{v}}$ in the time interval $[t_k, t_{k+1})$ such that $0 < \lambda_i(t_k) \leqslant 1$, where $\theta_i(t_k)$ is the heading angle of agent i at the time instant t_k. $\theta_{id}(t_k)$ is the target heading of agent i, and d is the diameter of the 2WMR. Then from **Lemma 3.1** and **Lemma 3.2**, it is seen that Ψ_k is SIA. From **Lemma 3.3**, we know that

$$\begin{cases} x(t_k) = \lim\limits_{k \to \infty} \Psi_k \Psi_{k-1} \ldots \Psi_0 x(t_0) = \mathbf{1}_N c_1^{\mathrm{T}} x(t_0), \\ y(t_k) = \lim\limits_{k \to \infty} \Psi_k \Psi_{k-1} \ldots \Psi_0 y(t_0) = \mathbf{1}_N c_1^{\mathrm{T}} y(t_0), \end{cases}$$

which implies that consensus can be reached. □

3.4.1 Consensus Analysis of the MAS with Second-order Dynamics

This section presents conditions for solving the consensus problem for a group of 2WMRs with second-order dynamics. Consider the system in (3.10), it is seen that the state convergence analyses on x-axis and y-axis are the same. For the convenience of the theoretical study, we only explore the convergence of the consensus on x-axis.

From (3.10), we have

$$\begin{bmatrix} x(t_{k+1}) \\ v_x(t_{k+1}) \end{bmatrix} = \Pi_k \Pi_{k-1} \dots \Pi_0 \begin{bmatrix} x(t_0) \\ v_x(t_0) \end{bmatrix} = \begin{bmatrix} B_k & C_k \\ D_k & E_k \end{bmatrix} \begin{bmatrix} x(t_0) \\ v_x(t_0) \end{bmatrix}.$$

In light of Lemma 3.1 in [114] and Lemma 1 in [38], we have the following lemma.

Lemma 3.4. *Assume that* $\alpha_k \rho_i(t_k) m_k h \neq 2$, $\forall k = 0, 1, \dots$ *and* $i = 1, 2, \dots, N$. *If* $\lim_{k\to\infty} B_k$ *exists and all rows of* $\lim_{k\to\infty} B_k$ *are same, then we have* $x_i(t_k) \to x_j(t_k),\ v_i(t_k) \to 0$ *for any initial conditions of* B_0 *and* D_0.

Proof. It can be proved by following the similar lines in [114]. □

In order to study the condition for reaching the consensus of the MAS in (3.10), we next derive the iteration of B_k. Denote

$$\Pi_k = \begin{bmatrix} \Pi_{k,11} & \Pi_{k,12} \\ \Pi_{k,21} & \Pi_{k,22} \end{bmatrix},$$

where $\Pi_{k,11}$, $\Pi_{k,12}$, $\Pi_{k,21}$ and $\Pi_{k,22}$ are the blocks in (3.11). It is shown that $\Pi_{k,12}$ and $\Pi_{k,22}$ are diagonal matrices. Then we have

$$\begin{aligned} B_k &= \Pi_{k,11} B_{k-1} + \Pi_{k,12} D_{k-1}, \\ B_{k-1} &= \Pi_{k-1,11} B_{k-2} + \Pi_{k-1,12} D_{k-2}, \\ D_{k-1} &= \Pi_{k-1,21} B_{k-2} + \Pi_{k-1,22} D_{k-2}. \end{aligned}$$

From the above equations, we obtain

$$B_k = \Psi_{k1} B_{k-1} + \Psi_{k2} B_{k-2},$$

where

$$\begin{cases} \Psi_{k1} = \Pi_{k,11} + \Pi_{k,12}\Pi_{k-1,22}\Pi_{k-1,12}^{-1}, \\ \Psi_{k2} = \Pi_{k,12}\Pi_{k-1,21} - \Pi_{k,12}\Pi_{k-1,22}\Pi_{k-1,12}^{-1}\Pi_{k-1,11}. \end{cases} \tag{3.12}$$

Then the equation can be further written as

$$\begin{bmatrix} B_k \\ B_{k-1} \end{bmatrix} = H_k \begin{bmatrix} B_{k-1} \\ B_{k-2} \end{bmatrix} = \begin{bmatrix} \Psi_{k1} & \Psi_{k2} \\ I_N & 0 \end{bmatrix} \begin{bmatrix} B_{k-1} \\ B_{k-2} \end{bmatrix}. \tag{3.13}$$

Lemma 3.5. *There exist control gains α_k and $\beta_k, k = 1, 2\ldots$ such that Ψ_{k1} and Ψ_{k2} are nonnegative matrices with positive diagonal entries. The matrix H_k is a stochastic matrix.*

Proof. Let $\alpha_k\rho_i(t_k)m_kh = \gamma \neq 2, \forall k$ and $i = 1, 2, \ldots, N$. The rotating time in the time interval $[t_k, t_{k+1})$ can be calculated as $\Delta T = \frac{|\theta_i(t_k)-\theta_{id}(t_k)|\pi d}{360\bar{v}}$. It follows that $\rho_i(t_k) = \frac{m_kh-\Delta T}{m_kh}$. Denote $\Phi_k = \mathrm{diag}\{\frac{\rho_1(t_k)}{\rho_1(t_{k-1})}, \frac{\rho_2(t_k)}{\rho_2(t_{k-1})}, \ldots, \frac{\rho_N(t_k)}{\rho_N(t_{k-1})}\}$.

Then we have

$$\begin{cases} \Pi_{k,12}\Pi_{k-1,22}\Pi_{k-1,12}^{-1} = \Phi_k(1-\gamma)\dfrac{m_k}{m_{k-1}}, \\ \Pi_{k,12}\Pi_{k-1,21} = -\rho_k m_k h(1-\dfrac{1}{2}\gamma)\beta_{k-1}\rho_{k-1}m_{k-1}hL(t_{k-1}), \\ \qquad = -\Phi_k\dfrac{m_k}{m_{k-1}}(1-\dfrac{1}{2}\gamma)\beta_{k-1}(\rho_{k-1}m_{k-1}h)^2L(t_{k-1}), \\ \Pi_{k,12}\Pi_{k-1,22}\Pi_{k-1,12}^{-1}\Pi_{k-1,11} = \Phi_k\dfrac{m_k}{m_{k-1}}(1-\gamma)\Pi_{k-1,11}. \end{cases} \tag{3.14}$$

Substituting (3.14) and $\Pi_{k-1,11}$ into (3.12), we obtain

$$\begin{cases} \Psi_{k1} = \Phi_k(1-\gamma)\dfrac{m_k}{m_{k-1}} + I_N - \dfrac{1}{2}(\rho_k m_k h)^2\beta_k L(t_k), \\ \Psi_{k2} = -\Phi_k\dfrac{m_k}{m_{k-1}}(1-\dfrac{1}{2}\gamma)\beta_{k-1}(\rho_{k-1}m_{k-1}h)^2L(t_{k-1}) - \Phi_k\dfrac{m_k}{m_{k-1}}(1-\gamma) \\ \qquad \times [I_N - \dfrac{1}{2}(\rho_{k-1}m_{k-1}h)^2\beta_{k-1}L(t_{k-1})] \\ \quad = -\dfrac{1}{2}\Phi_k\dfrac{m_k}{m_{k-1}}\beta_{k-1}(\rho_{k-1}m_{k-1}h)^2L(t_{k-1}) - \Phi_k(1-\gamma)\dfrac{m_k}{m_{k-1}}. \end{cases} \tag{3.15}$$

Note that Ψ_{k1} and Ψ_{k2} are all assumed to be nonnegative matrices and the diagonal entries are positive, and thus we first require that

$$1 + \frac{\rho_i(t_k)}{\rho_i(t_{k-1})}\frac{m_k}{m_{k-1}}(1-\gamma) > 0, \forall i = 1, 2, \ldots N,$$
$$(1-\gamma) < 0.$$

It gives that $1 < \gamma < 1 + \frac{\rho_i(t_{k-1})m_{k-1}}{\rho_i(t_k)m_k}$. In other words, since we have $\alpha_k \rho_i(t_k) m_k h = \gamma$, the control gain α_k should satisfy

$$\frac{1}{\rho_i(t_k)m_k h} < \alpha_k < \frac{\rho_i(t_{k-1})m_{k-1}}{(\rho_i(t_k)m_k)^2 h}. \tag{3.16}$$

Define

$$\hat{\beta}_1 = \min_i \left\{ \frac{2[\rho_i(t_k)m_k(1-\gamma) + \rho_i(t_{k-1})m_{k-1}]}{\rho_i(t_{k-1})m_{k-1}[\rho_i(t_k)m_k h]^2 l_{ii}(t_k)} \right\},$$
$$\hat{\beta}_2 = \min_i \left\{ \frac{-2(1-\gamma)}{(\rho_i(t_k)m_k h)^2 l_{ii}(t_k)} \right\}.$$

Next, if we choose β_k appropriately, i.e.,

$$0 < \beta_k < \min\left\{\hat{\beta}_1, \hat{\beta}_2\right\}, \tag{3.17}$$

then it is shown that Ψ_{k1} and Ψ_{k2} are both nonnegative matrices with positive diagonal entries. It follows that H_k is a nonnegative matrix.

Now we investigate the row sums of H_k. Considering (3.15), we have that the ith ($i = 1, 2, \ldots, N$) row sum of H_k can be equivalently written as

$$\begin{aligned} s_i(t_k) = &\frac{\rho_i(t_k)m_k}{\rho_i(t_{k-1})m_{k-1}}(1-\gamma) - \frac{1}{2}(\rho_k m_k h)^2 \beta_k \sum_{j=1}^{N} l_{ij}(t_k) \\ &- \frac{1}{2}\frac{\rho_i(t_k)m_k\beta_{k-1}}{\rho_i(t_{k-1})m_{k-1}}(\rho_{k-1}m_{k-1}h)^2 \sum_{j=1}^{N} l_{ij}(t_{k-1}) + 1 - \frac{\rho_i(t_k)m_k}{\rho_i(t_{k-1})m_{k-1}}(1-\gamma). \end{aligned}$$

Accordingly, noting that $L(t_k)\mathbf{1}_N = 0$, we obtain that $s_i(t_k) = 1, i = 1, 2, \ldots, N$. Combining the fact that $s_i(t_k) = 1, i = N+1, N+2, \ldots, 2N$, it implies that H_k is a stochastic matrix if α_k and β_k satisfy the conditions in (3.16) and (3.17). $\square$

We list two useful lemmas in drawing the main results.

Lemma 3.6 (Lemma 2 in [26]). *Let $m \geqslant 2$ be a positive integer and let $M_1, M_2, \ldots, M_m$ be nonnegative matrices with positive diagonal entries. There exists a positive number ε such that*

$$M_1 M_2 \ldots M_m \geqslant \varepsilon(M_1 + M_2 + \cdots + M_m).$$

Lemma 3.7 (Corollary 3.4 in [114]). *Suppose that the row sums of the nonnegative matrix $M \in \mathbb{R}^{n\times n}$ are same. It then follows that the directed graph associated with $\begin{bmatrix} M & M \\ M & M \end{bmatrix}$ has a directed spanning tree if the directed graph of M has a directed spanning tree.*

In the sequel, we present our main result on the consensus of a group of 2WMRs with the second-order dynamics and non-uniform sampling.

Theorem 3.2. *Let $\mathcal{G}(t_k)$ be switching directed graphs which have a spanning tree at the time instant $t_k, k = 1, 2, \ldots$. We assume that if the graph dynamically changes at the time instant t_{k+1}, then the graph remains unswitched at the next two sampling time instants, i.e., $\mathcal{G}(t_{k+1}) = \mathcal{G}(t_{k+2}) = \mathcal{G}(t_{k+3})$. Suppose that α_k and β_k satisfy the conditions in* (3.16) *and* (3.17), ***Algorithm 3.2*** *solves the second-order consensus problem for a group of 2WMRs with non-uniform sampling.*

Proof. From (3.13), it follows that

$$H_k H_{k-1} = \begin{bmatrix} \Psi_{k1}\Psi_{(k-1)1} + \Psi_{k2} & \Psi_{k1}\Psi_{(k-1)2} \\ \Psi_{(k-1)1} & \Psi_{(k-1)2} \end{bmatrix} \geqslant \begin{bmatrix} \Psi_{k2} & \Psi_{k1}\Psi_{(k-1)2} \\ \Psi_{(k-1)1} & \Psi_{(k-1)2} \end{bmatrix}. \tag{3.18}$$

From **Lemma 3.6** and **3.7**, there exists a positive number ε_1 such that

$$\Psi_{k1}\Psi_{(k-1)2} \geqslant \varepsilon_1(\Psi_{k1} + \Psi_{(k-1)2}).$$

Then

$$H_k H_{k-1} \geqslant \begin{bmatrix} \Psi_{k2} & \varepsilon_1\Psi_{k1} \\ \Psi_{(k-1)1} & \Psi_{(k-1)2} \end{bmatrix}.$$

From (3.15), we can see that the nonnegative matrices Ψ_{k1}, $\Psi_{(k-1)1}$, Ψ_{k2} and $\Psi_{(k-1)2}$ have zero and positive entries at the same positions if the associated Laplacian matrices $L(t_k)$, $L(t_{k-1})$ and $L(t_{k-2})$ have zero and non-zero entries at the same positions. Because Ψ_{k1} and Ψ_{k2} can be calculated as $\mathrm{diag}\{b_{1k}, b_{2k}, \ldots, b_{Nk}\} - \eta_k L(t_k)$ and $\mathrm{diag}\{c_{1k}, c_{2k}, \ldots, c_{Nk}\} - \eta_{k-1} L(t_{k-1})$, where $b_{ik}, c_{ik}, i = 1, 2, \ldots, N$ and η_k are scalars which can be specified from $m_{k-1}, m_k, \rho_i(t_k), \rho_i(t_{k-1}), \alpha_k, \alpha_{k-1}, \beta_k$ and β_{k-1}. Hence there exists $\varepsilon_0 > 0$, with the property that

$$H_k H_{k-1} \geqslant \varepsilon_0 \begin{bmatrix} \Psi_{k1} & \Psi_{k1} \\ \Psi_{k1} & \Psi_{k1} \end{bmatrix}.$$

It is readily to show that the directed graph associated with Ψ_{k1} has a spanning tree if the graph $\mathcal{G}(t_k)$ has a spanning tree. Using **Lemma 3.7**, we see that the directed graph of $H_k H_{k-1}$ also has a spanning tree. Therefore, from **Lemma 3.2**, $H_k H_{k-1}$ is SIA. Multiplying both sides of the equation in (3.18) by H_{k+1}, and then using an argument similar to the above process, we have

$$H_{k+1} H_k H_{k-1} \geqslant \begin{bmatrix} \varepsilon_1(\Psi_{(k+1)2} + \Psi_{(k-1)1}) & \varepsilon_2(\Psi_{(k+1)2} + \Psi_{(k-1)2}) \\ \Psi_{k2} & \varepsilon_3(\Psi_{k1} + \Psi_{(k-1)2}) \end{bmatrix}.$$

Note that the structure of the four blocks in the right-hand side of the above matrix inequality are determined by $L(t_k)$, $L(t_{k-1})$ and $L(t_{k-2})$ as well, which implies that the four blocks have zero and positive entries at the same positions. It then follows that

$$H_{k+1} H_k H_{k-1} \geqslant \bar{\varepsilon}_0 \begin{bmatrix} \Psi_{k1} & \Psi_{k1} \\ \Psi_{k1} & \Psi_{k1} \end{bmatrix}.$$

From the above statements we see that the switching interaction graph $\mathcal{G}(t_{k+1})$ can either be the same with $\mathcal{G}(t_k)$ or change dynamically. And then **Lemma 3.6** and **Lemma 3.2** explicitly show that $H_{k+1} H_k H_{k-1}$ is SIA.

Let the following sequence $H_k H_{k-1} \ldots H_2 H_1 = R_m\ R_{m-1} \ldots R_2 R_1$, where $R_i = H_{j+s} H_{j+s-1} \ldots\ H_{j+1} H_j, s \geqslant 3, i = 1, 2, \ldots, m$, with the property that $L(t_{j-1}) = L(t_j) = \cdots = L(t_{j+s-1}) \neq L(t_{j+s}) = L(t_{j+s+1}) \ldots$. It can be obtained from the similar argument that $R_i, i = 1, 2, \ldots, m$ are SIA. Then by applying **Lemma 3.3**, there exists a vector c_2 that

$$\lim_{m \to \infty} R_m R_{m-1} \ldots R_1 R_0 = \mathbf{1}_N c_2^{\mathrm{T}}.$$

B_k in (3.13) will have the identical rows as $k \to \infty$. Then, by **Lemma 3.4**, we show that consensus can be reached. □

3.5 Experiment

In Figure 3.3, the experimental setup is demonstrated. In order to maneuver vehicles with accurate positioning in a GPS-denied environment, each vehicle is defined and tracked as a unique rigid body with a cluster of reflective markers. Then OptiTrack$^{\text{TM}}$ cameras from NaturalPoint Inc. are employed to locate the marked rigid bodies. The two-wheeled ground robot Qbot is equipped with Gumstix microcontroller for running the real-time control software, QUARC$^{\circledR}$ from Quanser Consulting Inc. A ground station is employed to processes the OptiTrack$^{\text{TM}}$ data, and also to compile the QUARC$^{\circledR}$ models into the vehicles. Communications are established through the wireless network. In the following, we present the experimental results to demonstrate the effectiveness of the designed controllers.

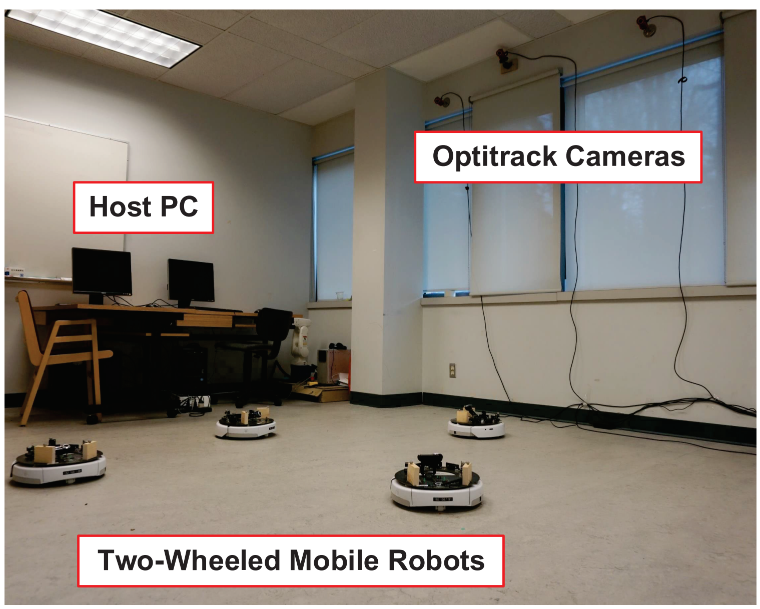


Figure 3.3: Experimental setup of the Quanser Unmanned Vehicle Systems Lab.

3.5.1 Experiment 1: Non-uniform Consensus of the MAS with First-order Dynamics

In this experiment, we examine the consensus performance of a group of 2WMRs with first-order dynamics.

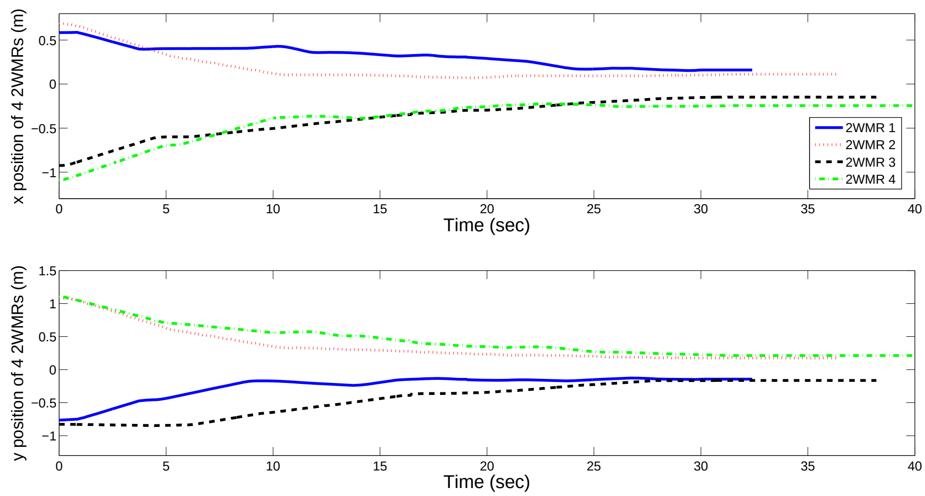


Figure 3.4: Experimental results of **Algorithm 3.1**: Time response of x and y positions of four 2WMRs.

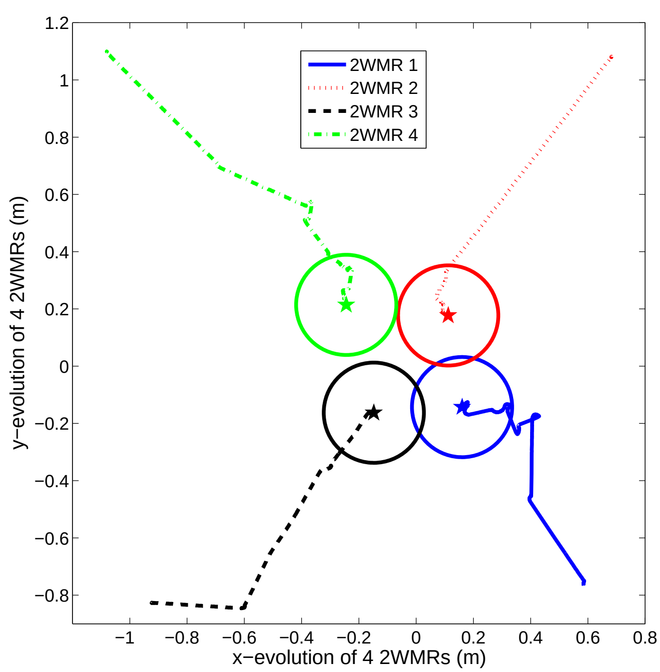


Figure 3.5: Experimental results of **Algorithm 3.1**: Trajectories of four 2WMRs.

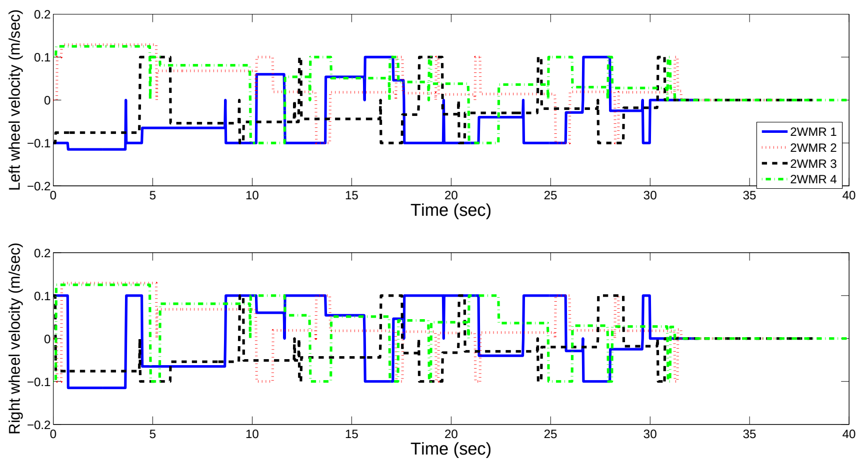


Figure 3.6: Experimental results of **Algorithm 3.1**: Wheel velocities of 2WMRs.

Figures 3.4 show the time response of x and y positions of four 2WMRs. Figure 3.5 shows the trajectories of the geometry centers of robots. The 2WMR is with a diameter 34 cm, and we say that consensus is reached if each robot's edge touches at least one edge of the other robot with their final positions. In Figure 3.5, we plot circles with diameter 34 cm representing 2WMRs' final positions. It is observed that some edges of 2WMRs are intersected. If a 2WMR's certain portion of edge mounted with a bump sensor collides with the other vehicle, its radius of this portion of edge may shrink. Figure 3.6 shows that physical constraints of wheel velocities $\|v_l\| \leqslant v_{max} = 0.5$ m/sec, $\|v_r\| \leqslant v_{max} = 0.5$ m/sec are satisfied.

3.5.2 Experiment 2: Non-uniform Consensus of the MAS with Second-order Dynamics

A testing is conducted to check the effectiveness of **Algorithm 3.2**. The experimental results are shown in Figures 3.7–3.8. It is shown that consensus can be reached. From Figures 3.6 and 3.8, we have the comparison result regarding the convergence speed between **Algorithm 1** and **Algorithm 3.2**: Consensus is reached faster by applying second-order control protocols.

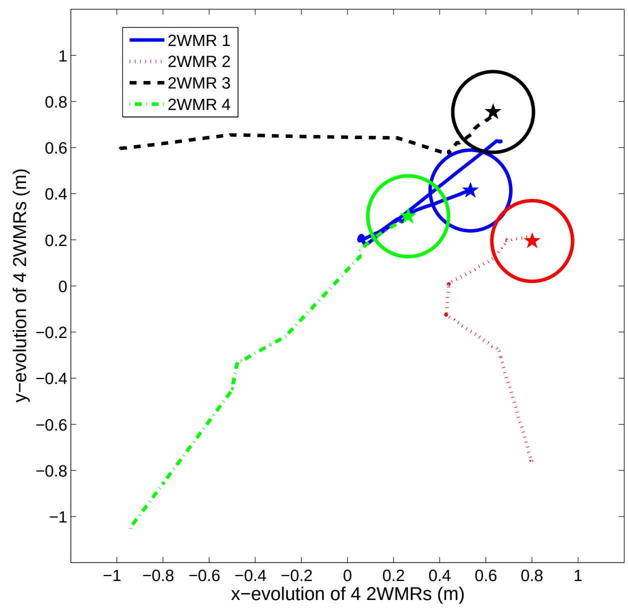


Figure 3.7: Experimental results of **Algorithm 3.2**: Trajectories of four 2WMRs.

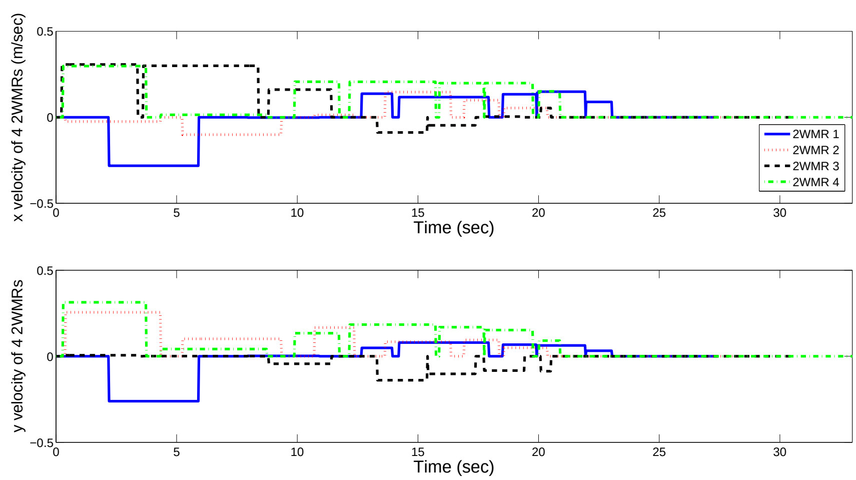


Figure 3.8: Experimental results of **Algorithm 3.2**: Time response of x and y velocities of four 2WMRs.

3.6 Conclusion

In this work, we investigated the consensus problem for a group of 2WMRs with non-uniform sampling and switching directed communication topologies. We developed consensus algorithms for first-order and second-order underactuated systems, respectively. The control gains are properly chosen and sufficient conditions are established in terms of the graph connectivity to ensure consensus. The effectiveness is validated by experimental results. Future research will be focused on the convergence speed analysis for the proposed algorithms and the study of consensus algorithms with asynchronous and non-uniform sampling.

A video of the experiment is posted on the following URL:
https://youtu.be/1O9mcvHcs-U

Chapter 4

Event-Based Rendezvous Control for a Group of Robots with Asynchronous Periodic Detection and Communication Time Delays

4.1 Introduction

In most results on cooperative control of MASs, it is usually assumed that all agents sample and broadcast data periodically, and agent controllers are updated synchronously. However, constraints such as time delays, data loss, limited communication resources need to be taken into consideration when designing the appropriate controllers. Many research studies have been carried out on the irregular sampling consensus control methods. In Chapter 3, we study the non-uniform sampling cooperative control for a group of 2WMRs. It is noted that the synchronous sampling control strategies are adopted in this work, which in essence involves synchronizing local clocks of the distributed MAS, and brings real challenges to practical applications.

Consensus problems with special regard to asynchronous agent behaviors are practical concerns. In [40], by using the nonnegative matrix theory, the authors study an asynchronous consensus control for the first-order continuous-time MAS with discrete data transmission. Cao and Wang investigate the second-order asynchronous consensus problem and analyze convergence through Lyapunov approach [125]. Later in [126], a velocity estimation based consensus method is proposed for the second-order asynchronous MAS in the sampled-data setting. More works can be found in [67, 127], etc. Most of the above mentioned works assume that agents update the controllers asynchronously but periodically. However, periodic and frequent samplings may result in redundant data transmissions in the case that time-scheduled sampling intervals are too small, especially when agents are equipped with resource-limited microcontrollers.

To address the above concern, an event-triggered control method is applied as an important technique in the study of consensus problems. The controller will only be updated when predefined event-triggering conditions are satisfied. Pioneer work on event-triggered consensus can be traced back to [99], and further researches are carried out from various standpoints, such as first-order consensus [128], second-order consensus [101], consensus in general linear MASs [102], nonlinear MASs [103], observer-based consensus [104] and more.

In our previous work [100], an event-triggered consensus control is proposed for an asynchronous MAS. Each agent checks the designed event-triggering condition periodically according to its own clock. If the event-triggering condition for an agent is satisfied, this agent samples and broadcasts its current state. The agent and its neighbors' controllers are updated right away. By using the novel integral-type event-triggering condition, the agent performance from the last event-detection instant is comprehensively considered. Consensus experiments on a group of asynchronous 2WMRs are also conducted.

Nevertheless, from the application-oriented perspective, the work in [100] can be improved in the following two aspects: 1) due to the existence of the intrinsic communication time delays, it is difficult to update MAS controllers promptly. It is of more practical importance to investigate the effects of time delays on the asynchronous controllers. There exist lots of works addressing consensus problems with time delays. For example, Zhang *et al.* [129] propose an effective consensus control method for the uncertain continuous-time MASs with time delays. More related works can be found in [6, 130–133]; 2) it is worth mentioning that the state of 2WMR in [100] is only with one degree-of-freedom (DOF), which implies that the algorithm can not be applied to solve the rendezvous problem. However, when operating the distributed 2WMRs in a two-dimensional plane, the rendezvous control can be applied in some useful applications such as recycling robots after cooperative control missions; e.g., search and rescue, mine sweeping, area exploration, etc.

Motivated by the above discussion, this chapter investigates the event-triggered rendezvous control for a group of 2WMRs with communication time delays. All agents have the same fixed event-checking period, but each agent checks triggering conditions asynchronously. Two integral-type event-triggering conditions need to be checked at the event-checking instant for each agent, i.e., one is for x and the other is for y state evolution. If the condition for x (or y) state evolution is satisfied, this agent samples current x (or y) state and broadcast it to the neighbors through networks with a time-varying delay. After an agent receives the updated data, it calculates the its control input. We design a *Rotate&Compensate&Run Rendezvous Scheme* to tune the orientation of the agent and then update the states. The sufficient condition is given to show that 2WMRs are eventually driven to reach rendezvous at the average of all agents's initial states.

The remainder of the chapter is organized as follows. Section 4.2 introduces some preliminaries. In Section 4.3, we formulate the 2WMR rendezvous problem and provide the distributed event-triggered control method. We analyze the convergence of 2WMRs with time-varying delays in Section 4.4. Experiments are given in Section 4.5 to verify the effectiveness of the designed control method. Section 4.6 presents the conclusion and future work.

4.2 Preliminaries

In this section, we first review some preliminaries in the graph theory and algebraic theory [23], which will be used throughout this chapter.

Different from the work using directed graphs in Chapter 3, an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ is employed to describe interactions among the MAS over the network in this chapter. $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ represents the vertex set with N nodes. $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set. $e_{ij} = (v_j, v_i) \in \mathcal{E}$ indicates that vertex i can receive information from vertex j. For an undirected graph, the communication between two vertices are bidirectional, and thus e_{ij} and e_{ij} indicate the same edge. The set of neighbors of vertex i is defined by $\mathcal{N}_i = \{j | e_{ij} \in \mathcal{E}\}$, for $j = 1, 2, \ldots, N, j \neq i$. For any two vertices i and j, if there exists a sequence of vertices $l_1, l_2, \ldots, l_r$, $l_1 = i$, $l_r = j$, such that $e_{l_1 l_2}, e_{l_2 l_3}, \ldots, e_{l_{r-1} l_r} \in \mathcal{E}$, we say that the undirected graph is *connected*. The nonnegative matrix $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ is the adjacency matrix. $a_{ii} = 0$ and $a_{ij} = a_{ji} > 0$ if $e_{ij} \in \mathcal{E}$; otherwise $a_{ij} = a_{ji} = 0$. The *graph Laplacian* $L = [l_{ij}] \in \mathbb{R}^{N \times N}$ is defined by $l_{ij} = -a_{ij},\ \forall i \neq j;\ l_{ii} = \sum_{j=1, j \neq i}^{N} a_{ij}$. It is shown that L is a symmetric and positive semi-definite matrix for any undirected graph. L has N real eigenvalues, and $\lambda_1 \leq \lambda_2 \leq, \ldots, \leq \lambda_N$ with $\lambda_1 = 0$. For an undirected connected graph, λ_1 of Laplacian matrix L is of algebraic multiplicity 1, and $\mathbf{1}_N$ is an associated eigenvector of λ_1. Regarding the second largest eigenvalue λ_2 of L, we have the following important lemma to be used later.

Lemma 4.1. *[23] The graph Laplacian $L \in \mathbb{R}^{N \times N}$ of an undirected and connected graph $\mathcal{G}$ satisfies the property:*

$$\frac{x^{\mathrm{T}} L x}{\|x\|_2^2} \geq \lambda_2, \quad for\ any\ x \neq 0\ and\ \mathbf{1}_N^{\mathrm{T}} x = 0.$$

4.3 Problem Formulation and Controller Design

4.3.1 Problem Formulation

The coordinate system of the 2WMR [2] in the earth-fixed frame is shown in Section 3.3.1. This frame $v_E = \{x, y\}$ is used in both controller design and real-time application. Two wheels of the 2WMR are mounted parallel on a common axis through the geometry center of the robot. Each wheel is controlled independently to move forward or backward such that the robot motion can be changed. Suppose we have N 2WMRs in the MAS, and $x_i(t)$ and $y_i(t)$ denote x and y positions of agent i, respectively, in the earth-fixed coordinate v_E. In this work, we will design the control inputs for both wheels of each 2WMR using the asynchronous and periodic event-checking approach to solve the rendezvous problem. We say that the rendezvous control for a group of 2WMRs is achieved asymptotically if $\lim_{t\to\infty} \|x_i(t) - x_j(t)\| = 0$ and $\lim_{t\to\infty} \|y_i(t) - y_j(t)\| = 0,\ \forall i, j = 1, 2, \ldots, N,\ i \neq j$.

4.3.2 Rendezvous Control Method Design

To study the rendezvous problem, we first assume that the dynamics of the agent only has one degree of freedom, and then investigate the consensus control for the following MAS consisting of N first-order dynamics:

$$\dot{x}_i(t) = u_{xi}(t), \quad i = 1, 2, \ldots, N. \tag{4.1}$$

The rendezvous control method for 2WMRs will be discussed later based on the above dynamics. Here we set the event-checking periods equivalently as h for all agents in (4.1). Agent i starts at $t_0^i \in [0, h)$, for $i = 1, 2, \ldots, N$. The event-checking instants for agent i will be $t_1^i, t_2^i, \ldots, t_k^i, \ldots$, in which $t_{k+1}^i = t_k^i + h,\ k = 0, 1, 2, \ldots$. t_0^i is defined as the first event instant. Besides, if the triggering condition is satisfied at one event-checking instant, we also call this instant as the event instant. The event instants of agent i can be written as an increasing sequence $t_{(0)}^i, t_{(1)}^i, t_{(2)}^i, \cdots$, where $t_{(0)}^i = t_0^i$. Agent i samples its state $x_i(t_{(k)}^i)$ and broadcasts it to the neighbors with a time delay $\tau_{(k)}^i$. Let τ_k^i be the communication time delay if agent i samples the state at t_k^i, for $k = 0, 1, 2, \ldots$. An illustration of the event-checking scheme is shown in Figure 4.1. Accordingly, the latest broadcast state of agent i is given as:

$$\hat{x}_i(t) = x_i(t_{(k)}^i), \quad t_{(k)}^i + \tau_{(k)}^i \leq t < t_{(k+1)}^i + \tau_{(k+1)}^i, \quad k = 0, 1, 2, \ldots.$$

The control input of agent i is

$$u_{xi}(t) = -\sum_{j \in N_i} \tilde{a}_{ij}(t)\big(\hat{x}_i(t) - \hat{x}_j(t)\big), \quad t \in [0, +\infty), \tag{4.2}$$

where $\tilde{a}_{ij}(t) = 0$ for $t < \max\{t_0^i + \tau_0^i,\ t_0^j + \tau_0^j\}$, $\tilde{a}_{ij}(t) = a_{ij}$ for $t \geq \max\{t_0^i + \tau_0^i,\ t_0^j + \tau_0^j\}$, and $\hat{x}_i(t) = 0$ for $t \in [0, t_0^i + \tau_0^i)$. The setup of $\tilde{a}_{ij}(t)$ indicates that the communication link between agents i and j is not established until both of them receive each other's information.

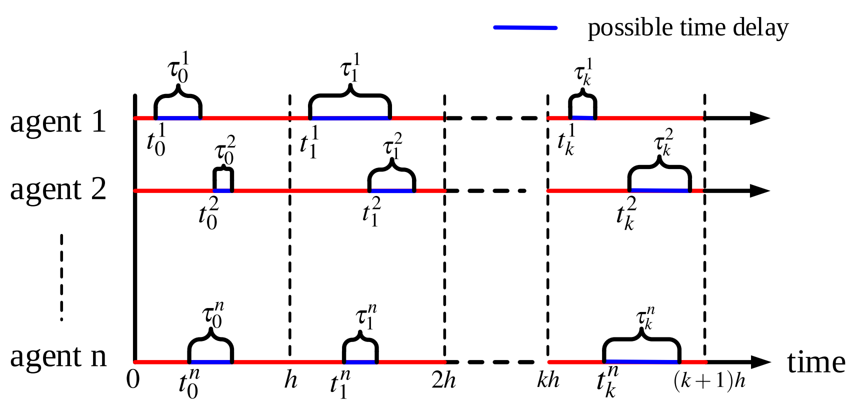


Figure 4.1: Event-checking time instants of the multi-agent system.

Remark 4.1. *Following a similar line as discussed in the previous work [100], we consider a practical application scenario that the agent starts asynchronously and checks the event-triggering conditions periodically by its own clock. On the other hand, if the inevitable communication delays are not incorporated within the controller design, as shown in [100], it can sometimes bring difficulties when we implement the* *control method on real-time systems; especially when the distributed MAS works in an environment with the limited communication bandwidth. If the time delays are small enough and neglectable, the work [100] can be considered as a special case under our proposed framework here. Moreover, $\tau_k^i = 0$ in the situation that the event-triggering condition is not satisfied at t_k^i, agent i does not sample the state, and the latest broadcast data of agent i is not changed.*

Assume that $t_k^i + \tau_k^i < (k+1)h$, which implies that time delays are bounded. The system dynamics of agent i is then written as

$$\dot{x}_i(t) = -\sum_{j \in N_i} a_{ij}\big(\hat{x}_i(t) - \hat{x}_j(t)\big), \qquad t \in [h, +\infty). \tag{4.3}$$

Define $x(t) = [x_1(t), x_2(t), \ldots, x_N(t)]^{\mathrm{T}}$ and $\hat{x}(t) = [\hat{x}_1(t), \hat{x}_2(t), \ldots, \hat{x}_N(t)]^{\mathrm{T}}$. The MAS dynamics in (4.1) is then represented by

$$\dot{x}(t) = -L\hat{x}(t), \quad t \in [h, +\infty), \tag{4.4}$$

where L is the graph Laplacian.

If the graph is undirected and $a_{ij} = a_{ji}$, it is readily shown that the average value of all agents' states is time-invariant. Let $\bar{x}(t) = \frac{1}{N}\sum_{i=1}^N x_i(t), \;\; t \in [0, +\infty)$. We have that $\dot{\bar{x}}(t) = \frac{1}{N}\sum_{i=1}^N \dot{x}_i(t) = -\frac{1}{N}\sum_{i=1}^N \sum_{j \in N_i} \tilde{a}_{ij}\big(\hat{x}_i(t) - \hat{x}_j(t)\big) = 0$. Next we design the event-triggering condition for agent i to check at $t_{(k)}^i + qh$:

$$\Big|x_i(t_{(k)}^i + qh) - x_i(t_{(k)}^i + \tau_{(k)}^i)\Big| > \sigma\sqrt{\frac{\int_{t_{(k)}^i + (q-1)h + \tau_l^i}^{t_{(k)}^i + qh} (L_i\hat{x}(w))^2 dw}{h}}, \tag{4.5}$$

where σ is a positive scalar, L_i is the i-th row of the graph Laplacian associated with $[\tilde{a}_{ij}(t)]$, and $l = \bar{k} + q - 1$, $t_{(k)}^i = t_0^i + \bar{k}h$, indicating that $t_{(k)}^i$ is the $\bar{k}$th event-checking instant in agent i's event-checking sequence. In our triggering-checking process, q takes successively the integer values $1, 2, 3, \ldots,$ until inequality (4.5) holds for the first q. We set $t_{(k+1)}^i = t_{(k)}^i + qh$.

If the dynamics of the 2WMR along x-axis and y-axis are independent, the controller of y state can be presented in the similar argument:

$$\dot{y}_i(t) = u_{yi}(t), \quad i = 1, 2, \ldots, N. \tag{4.6}$$

Let $\hat{y}_i(t)$ be the latest broadcast data of y position of agent i, we have

$$\hat{y}_i(t) = y_i(t^i_{(k)}), \quad t^i_{(k)} + \tau^i_{(k)} \leq t < t^i_{(k+1)} + \tau^i_{(k+1)}, \quad k = 0, 1, 2, \ldots.$$

The control input along y-axis is

$$u_{yi}(t) = -\sum_{j \in N_i} \tilde{a}_{ij}(t)\big(\hat{y}_i(t) - \hat{y}_j(t)\big), \quad t \in [0, +\infty), \tag{4.7}$$

and the dynamics of agent i along y-axis is

$$\dot{y}_i(t) = -\sum_{j \in N_i} a_{ij}\big(\hat{y}_i(t) - \hat{y}_j(t)\big), \quad t \in [h, +\infty). \tag{4.8}$$

Define $y(t) = [y_1(t), y_2(t), \ldots, y_N(t)]^{\mathrm{T}}$ and $\hat{y}(t) = [\hat{y}_1(t), \hat{y}_2(t), \ldots, \hat{y}_N(t)]^{\mathrm{T}}$. We have

$$\dot{y}(t) = -L\hat{y}(t), \quad t \in [h, +\infty). \tag{4.9}$$

The triggering condition for y state is given as

$$\Big|y_i(t^i_{(k)} + qh) - y_i(t^i_{(k)} + \tau^i_{(k)})\Big| > \sigma\sqrt{\frac{\int_{t^i_{(k)}+(q-1)h+\tau^i_l}^{t^i_{(k)}+qh}(L_i\hat{y}(w))^2 dw}{h}}. \tag{4.10}$$

Remark 4.2. *Note that 2WMRs are operated in the two-dimensional plane, and two wheels are mounted on a common axis through the robot geometry center. The mechanical structure of the 2WMR determines that two wheels can only be controlled to move forward or backward with the perpendicular velocity to the wheels connecting axis. Thus the proposed consensus controller in this section is not appropriate to be applied to the 2WMR dynamics. Therefore, to solve the rendezvous problem of 2WMRs, we propose a* Rotate&Compensate&Run Rendezvous Scheme *based on the consensus controllers in* (4.2) *and* (4.7).

As shown in Figure 4.2, the body-fixed axes x' (pointing from the 2WMR center to the nose) and y' (pointing from the 2WMR center to the left) divide the plane into four regions according to the sign of the coordinates (x', y'); i.e., Quadrant I (+, −); Quadrant II (+, +); Quadrant III (−, +) and Quadrant IV (−, −). Suppose at the time instant t_a, agent i's broadcast data is received by the other agent, or agent i receives the updated information. The agent then needs to calculate the

Algorithm 4.1 *Asynchronous Rotate&Compensate&Run Rendezvous Scheme* for 2WMR i

Input: $\hat{x}_i(t),\ \hat{y}_i(t),\ \hat{x}_j(t),\ \hat{y}_j(t),\ \theta_i(t), \forall j \in \mathcal{N}_i,\ t.$

Output: Left wheel and right wheel velocities $v_l(t)$ and $v_r(t)$.

1: when $t < t_0^i$, $\hat{x}_i(t) = 0$, $\hat{y}_i(t) = 0$, and $v_l(t) = 0, v_r(t) = 0$. At $t = t_0^i$, agent i samples the current states $x_i(t_0^i)$, $y_i(t_0^i)$, and broadcasts them to the neighbors; at $t_k^i = t_0^i + kh$, $k = 1, 2, 3, \ldots$, agent i checks its triggering condition in (4.5) and (4.10). If the condition in (4.5) or (4.10) is satisfied, the agent samples the current $x_i(t_k^i)$ or $y_i(t_k^i)$, and broadcasts the data to the network.

2: Suppose 2WMR m is the first agent who establishes the undirected communication link with agent i, at $t = \max\{t_0^i + \tau_0^i,\ t_0^m + \tau_0^m\}$, goto Step 3;

3: Agent i calculates the following values: Its x and y controllers $u_{xi}(t)$ and $u_{yi}(t)$ using (4.2) and (4.7); the desired orientation $\bar{\theta}_i = \text{atan2}(u_{yi}(t),\ u_{xi}(t))$, and the desired velocity $v_i = \sqrt{(u_{x_i}(t))^2 + (u_{y_i}(t))^2}$.

4: Agent i rotates in place until its tail or the nose aiming at θ_i, and the length of rotating time is recorded as ΔT_i.

5: Agent i moves forward/backward if its nose/tail aims at θ_i, at the velocity $v_i + \frac{v_i}{\alpha}$ for the time period $\alpha \Delta T_i$.

6: Agent i keeps moving forward/backward at v_i until it receives the updated data, or until its new broadcast data is received by the other agent. Goto Step 3.

$\theta_i(t)$ is heading of agent i. α is a positive scalar.

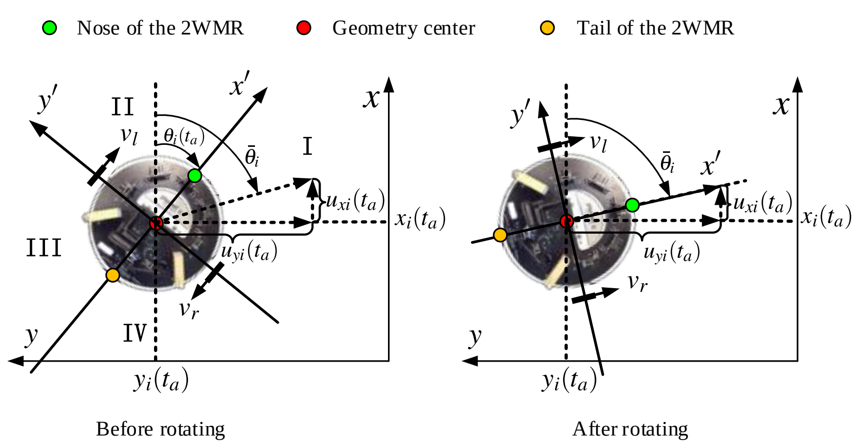


Figure 4.2: Rotating illustration of the 2WMR.

desired orientation $\bar{\theta}_i$. If $\bar{\theta}_i$ is Quadrant I/III, the 2WMR rotates in place clockwise, and moves forward/backward after rotating; otherwise if $\bar{\theta}_i$ is Quadrant II/IV, the 2WMR rotates in place anticlockwise, and moves forward/backward after rotating. By using the proposed rotating scheme, we minimize the angle that the 2WMR needs to rotate such that either its nose or tail aims at the desired orientation.

In Step 4 of **Algorithm 4.1**, ΔT_i is the rotating time for the 2WMR to tune its heading, during which, x and y velocities of the 2WMR are both 0. Then in Step 5, the velocity of the 2WMR is set as $v_i + \frac{v_i}{\alpha}$ in the following time period $\alpha\Delta T_i$. We assume that the rotating time ΔT and the state compensating time $\alpha\Delta T$ are both small such that agent i's next controller updating instant is after the state compensating time period $\alpha\Delta T_i$. The distance evolution of the 2WMR in $\Delta T_i + \alpha\Delta T_i$ can be readily calculated as $\Delta T_i \times 0 + \alpha\Delta T_i \times (v_i + \frac{v_i}{\alpha}) = v_i \times (\Delta T_i + \alpha\Delta T_i)$. Note that by using the controller in (4.2) to the one DOF MAS, after a controller updating instant, x state evolution in the time period $\Delta T_i + \alpha\Delta T_i$ is $u_{xi}(t)(\Delta T_i + \alpha\Delta T_i)$, which is equivalent to x state evolution of the 2WMR using our proposed *Rotate&Compensate&Run Rendezvous Scheme*. Therefore, the part $\frac{v_i}{\alpha}$ in Step 5 during $\alpha\Delta T_i$ is designed for compensating x and y state evolution losses during the rotating time ΔT. It is also shown that in the end of the state compensation time periods $\max\{\Delta T_i + \alpha\Delta T_i, \Delta T_j + \alpha\Delta T_j\}$ of both 2WMRs, the average x and y state values of 2WMRs do not change.

In the next section, we present a sufficient condition for 2WMRs to reach rendezvous. The terminal rendezvous x (y) state of 2WMRs is the average value of the initial x (y) states of the MAS.

4.4 Main Result

In this section, we present the main result on the rendezvous conditions. Except in the rotating time ΔT and in the state compensating time $\alpha\Delta T$, the 2WMR x state evolution is equivalent to the state evolution of the agent in (4.1) if we apply the consensus controller in (4.2) and the triggering condition in (4.5) to (4.1). Thus the convergence analysis of 2WMRs rendezvous control can be conducted by analyzing the consensus performance of the control protocol in (4.2).

Theorem 4.1. *A group of distributed MAS in* (4.4) *is driven by the control input in* (4.2) *with the triggering condition in* (4.5)*, and the communication topology is undirected and connected. Assume that the time delay is bounded, i.e.,* $t_k^i + \tau_k^i < (k+1)h$*, for any* $i = 1, 2, \ldots, N$ *and* $k = 0, 1, \ldots$*. If the event-checking period* h*, positive scalar* σ *in* (4.5) *and the largest eigenvalue* λ_N *of* L *satisfy the following condition:*

$$2h + \frac{3}{2}\sigma < \frac{1}{\lambda_N}, \tag{4.11}$$

then states of the MAS asymptotically converge to the average value of the initial states $\bar{x}(0)$.

Proof. Let $\xi(t) = [\xi_1(t)\ \xi_2(t)\ \cdots\ \xi_n(t)]^{\mathrm{T}} = x(t) - \bar{x}(0)\mathbf{1}_N$. Consider the following Lyapunov function candidate:

$$V(t) = \frac{1}{2}\xi(t)^{\mathrm{T}}\xi(t), \quad t \in [t_0, +\infty).$$

Taking the derivative of $V(t)$ with respect to time t, we have

$$\dot{V}(t) = \xi(t)^{\mathrm{T}}\dot{\xi}(t) = -(x(t) - \bar{x}(0)\mathbf{1}_N)^{\mathrm{T}}L\hat{x} = -x(t)^{\mathrm{T}}L\hat{x}(t).$$

The controller update instants of the agents are in the set $\{t_k^i + \tau_k^i\}$, for $i = 1, 2, \ldots, N$ and $k = 0, 1, \ldots$. Let $t_0 = \min\{t_1^1 + \tau_1^1, t_1^2 + \tau_1^2, \ldots, t_1^N + \tau_1^N\}$. Putting all the instants in $\{t_k^i + \tau_k^i\}$ after t_0 in the ascending order, we get an increasing sequence of time instants, i.e., $t_0, t_1, t_2, \ldots, t_k, \ldots$. Then we have

$$\begin{aligned} V(t_{k+1}) =& V(t_0) + \int_{t_0}^{t_{k+1}} \dot{V}(w)dw \\ =& V(t_0) - \int_{t_0}^{t_{k+1}} x(w)^{\mathrm{T}}L\hat{x}(w)dw \\ =& V(t_0) - \sum_{i=1}^{N}\int_{t_0}^{t_{k+1}} x_i(w)L_i\hat{x}(w)dw. \end{aligned} \tag{4.12}$$

For any k and each $i \in \{1, 2, \ldots, N\}$, we can find q_i such that $t_{q_i+1}^i + \tau_{q_i+1}^i < t_{k+1} \leq t_{q_i+1}^i + h + \tau_{q_i+2}^i$. It follows that

$$\begin{aligned} &-\int_{t_0}^{t_{k+1}} x_i(w)L_i\hat{x}(w)dw \\ &= -\Big(\int_{t_0}^{t_1^i+\tau_1^i} + \int_{t_1^i+\tau_1^i}^{t_2^i+\tau_2^i} + \cdots + \int_{t_p^i+\tau_p^i}^{t_{p+1}^i+\tau_{p+1}^i} + \cdots + \int_{t_{q_i}^i+\tau_{q_i}^i}^{t_{q_i+1}^i+\tau_{q_i+1}^i} \\ &\quad + \int_{t_{q_i+1}^i+\tau_{q_i+1}^i}^{t_{k+1}}\Big)x_i(w)L_i\hat{x}(w)dw. \end{aligned} \tag{4.13}$$

Given any p, if $t_r = t_p^i + \tau_p^i$, $t_{r+k'} \leq t_{p+1}^i + \tau_{p+1}^i$, using (4.4), we have

$$-\int_{t_p^i+\tau_p^i}^{t_{r+k'}} x_i(w)L_i\hat{x}(w)dw$$

$$
\begin{aligned}
&= -\int_{t_p^i+\tau_p^i}^{t_{r+k'}} \Big(x_i(t_p^i+\tau_p^i) - \int_{t_p^i+\tau_p^i}^{w} L_i\hat{x}(t)dt\Big) L_i\hat{x}(w)dw \\
&= -\int_{t_p^i+\tau_p^i}^{t_{r+k'}} \Big(x_i(t_p^i) - \int_{t_p^i}^{t_p^i+\tau_p^i} L_i\hat{x}(t)dt\Big) L_i\hat{x}(w)dw + \int_{t_p^i+\tau_p^i}^{t_{r+k'}} \Big(\int_{t_p^i+\tau_p^i}^{w} L_i\hat{x}(t)dt\Big) L_i\hat{x}(w)dw \\
&= -\int_{t_p^i+\tau_p^i}^{t_{r+k'}} x_i(t_p^i) L_i\hat{x}(w)dw + \int_{t_p^i+\tau_p^i}^{t_{r+k'}} \int_{t_p^i}^{t_p^i+\tau_p^i} L_i\hat{x}(t)dt L_i\hat{x}(w)dw \\
&\quad + \int_{t_p^i+\tau_p^i}^{t_{r+k'}} \Big(\int_{t_p^i+\tau_p^i}^{w} L_i\hat{x}(t)dt\Big) L_i\hat{x}(w)dw. \qquad (4.14)
\end{aligned}
$$

We calculate the three terms in (4.14).

$$
\begin{aligned}
\Xi_1 = &\int_{t_p^i+\tau_p^i}^{t_{r+k'}} \Big(\int_{t_p^i+\tau_p^i}^{w} L_i\hat{x}(t)dt\Big) L_i\hat{x}(w)dw \\
=&\frac{1}{2}\Big(\int_{t_p^i+\tau_p^i}^{t_{r+k'}} L_i\hat{x}(t)dt\Big)^2 \\
=&\frac{1}{2}\Big(\int_{t_r}^{t_{r+1}} L_i\hat{x}(t)dt + \cdots + \int_{t_{r+k'-1}}^{t_{r+k'}} L_i\hat{x}(t)dt\Big)^2 \\
=&\frac{1}{2}\Big((t_{r+1}-t_r)L_i\hat{x}(t_r) + (t_{r+2}-t_{r+1})L_i\hat{x}(t_{r+1}) + \cdots \\
&+ (t_{r+k'}-t_{r+k'-1})L_i\hat{x}(t_{r+k'-1})\Big)^2 \\
=&2h^2\Big(\frac{t_{r+1}-t_r}{2h}L_i\hat{x}(t_r) + \frac{t_{r+2}-t_{r+1}}{2h}L_i\hat{x}(t_{r+1}) + \cdots \\
&+ \frac{t_{r+k'}-t_{r+k'-1}}{2h}L_i\hat{x}(t_{r+k'-1})\Big)^2 \\
\le&2h^2\Big(\frac{t_{r+1}-t_r}{2h}(L_i\hat{x}(t_r))^2 + \frac{t_{r+2}-t_{r+1}}{2h}(L_i\hat{x}(t_{r+1}))^2 \\
&+ \cdots + \frac{t_{r+k'}-t_{r+k'-1}}{2h}(L_i\hat{x}(t_{r+k'-1}))^2\Big) \\
=&h\sum_{q=r}^{r+k'-1}(t_{q+1}-t_q)\hat{x}(t_q)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(t_q), \qquad (4.15)
\end{aligned}
$$

and let $t_p^i = t_m$, we have

$$
\Xi_2 = \int_{t_p^i+\tau_p^i}^{t_{r+k'}} \int_{t_p^i}^{t_p^i+\tau_p^i} L_i\hat{x}(t)dt L_i\hat{x}(w)dw
$$

$$
\begin{aligned}
&\leq \frac{1}{2}\Big(\int_{t_p^i}^{t_p^i+\tau_p^i} L_i\hat{x}(t)dt\Big)^2 + \frac{1}{2}\Big(\int_{t_p^i+\tau_p^i}^{t_{r+k'}} L_i\hat{x}(w)dw\Big)^2 \\
&= \frac{1}{2}\Big(\int_{t_m}^{t_r} L_i\hat{x}(t)dt\Big)^2 + \Xi_1 \\
&= \frac{1}{2}\Big(\int_{t_m}^{t_{m+1}} L_i\hat{x}(t)dt + \cdots + \int_{t_{r-1}}^{t_r} L_i\hat{x}(t)dt\Big)^2 + \Xi_1 \\
&= \frac{h^2}{2}\Big(\frac{t_{m+1}-t_m}{h}L_i\hat{x}(t_m) + \cdots + \frac{t_r - t_{r-1}}{h}L_i\hat{x}(t_{r-1})\Big)^2 + \Xi_1 \\
&\leq \frac{h^2}{2}\Big(\frac{t_{m+1}-t_m}{h}(L_i\hat{x}(t_m))^2 + \cdots + \frac{t_r - t_{r-1}}{h}(L_i\hat{x}(t_{r-1}))^2\Big) + \Xi_1 \\
&= \frac{h}{2}\sum_{q=m}^{r-1}(t_{q+1}-t_q)\hat{x}(t_q)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(t_q) + \Xi_1. \qquad (4.16)
\end{aligned}
$$

Let $\Delta_i(t_p^i) = x_i(t_p^i) - \hat{x}_i(t_p^i + \tau_p^i)$. It follows that if the triggering condition in (4.5) is satisfied at t_p^i, agent i samples the state $x_i(t_p^i)$; the controller updates at the instant $t_p^i + \tau_p^i$, and then $\Delta_i(t_p^i) = 0$. Otherwise the triggering condition in (4.5) is not satisfied, $\hat{x}_i(t_p^i + \tau_p^i) = \hat{x}_i(t_{p-1}^i + \tau_{p-1}^i)$. We have

$$
|\Delta_i(t_p^i)| = \Big|x_i(t_p^i) - \hat{x}_i(t_{p-1}^i + \tau_{p-1}^i)\Big| \leq \sigma\sqrt{\frac{\int_{t_{p-1}^i+\tau_{p-1}^i}^{t_p^i}(L_i\hat{x}(w))^2dw}{h}}. \qquad (4.17)
$$

It follows that

$$
\begin{aligned}
\Xi_3 =& -\int_{t_p^i+\tau_p^i}^{t_{r+k'}} x_i(t_p^i)L_i\hat{x}(w)dw \\
=& -\int_{t_p^i+\tau_p^i}^{t_{r+k'}} \Big(\hat{x}_i(t_p^i+\tau_p^i) + \Delta_i(t_p^i)\Big)L_i\hat{x}(w)dw \\
=& -\int_{t_p^i+\tau_p^i}^{t_{r+k'}} \Delta_i(t_p^i)L_i\hat{x}(w)dw - \int_{t_p^i+\tau_p^i}^{t_{r+k'}} \hat{x}_i(t_p^i+\tau_p^i)L_i\hat{x}(w)dw \\
\leq& \frac{1}{2}\int_{t_p^i+\tau_p^i}^{t_{r+k'}} [\frac{1}{\sigma}\Delta_i(t_p^i)^2 + \sigma(L_i\hat{x}(w))^2]dw - \int_{t_p^i+\tau_p^i}^{t_{r+k'}} \hat{x}_i(t_p^i+\tau_p^i)L_i\hat{x}(w)dw \\
\leq& \frac{1}{2\sigma}(t_{r+k'} - t_p^i - \tau_p^i)\frac{\sigma^2}{h}\int_{t_{p-1}^i+\tau_{p-1}^i}^{t_p^i} (L_i\hat{x}(w))^2dw \\
&+ \frac{\sigma}{2}\int_{t_p^i+\tau_p^i}^{t_{r+k'}} \hat{x}(w)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(w)dw - \int_{t_p^i+\tau_p^i}^{t_{r+k'}} \hat{x}_i(t_p^i+\tau_p^i)L_i\hat{x}(w)dw
\end{aligned}
$$

$$\leq\sigma\int_{t_{p-1}^i+\tau_{p-1}^i}^{t_p^i}\hat{x}(w)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(w)dw$$
$$+\frac{\sigma}{2}\int_{t_p^i+\tau_p^i}^{t_{r+k'}}\hat{x}(w)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(w)dw-\int_{t_p^i+\tau_p^i}^{t_{r+k'}}\hat{x}_i(t_p^i+\tau_p^i)L_i\hat{x}(w)dw. \tag{4.18}$$

By using (4.17), we get

$$-\int_{t_1^i+\tau_1^i}^{t_{k+1}}\Delta_i(t_p^i)L_i\hat{x}(w)dw$$
$$=-\Big(\int_{t_1^i+\tau_1^i}^{t_2^i+\tau_2^i}+\int_{t_2^i+\tau_2^i}^{t_3^i+\tau_3^i}+\cdots+\int_{t_{q_i+1}^i+\tau_{q_i+1}^i}^{t_{k+1}}\Delta_i(t_p^i)L_i\hat{x}(w)dw\Big)$$
$$\leq\sigma\int_{t_0^i+\tau_0^i}^{t_1^i}\hat{x}(w)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(w)dw+\frac{3\sigma}{2}\int_{t_1^i+\tau_1^i}^{t_{k+1}}\hat{x}(w)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(w)dw. \tag{4.19}$$

Set $t_{k_0}=\max_{1\leq i\leq N}\{t_1^i+\tau_1^i\}$ and $t_{k_0^i}=t_1^i+\tau_1^i$. Define

$$V_0=V(t_0)-\sum_{i=1}^{N}\int_{t_0}^{t_1^i+\tau_1^i}x_i(w)L_i\hat{x}(w)dw-\sum_{i=1}^{N}\sum_{q=k_0^i}^{k_0-1}\hat{x}_i(w)L_i\hat{x}(w)dw$$
$$+\sigma\sum_{i=1}^{N}\int_{t_0^i+\tau_0^i}^{t_1^i}\hat{x}^{\mathrm{T}}(w)L_i^{\mathrm{T}}L_i\hat{x}(w)dw$$
$$+(\frac{3\sigma}{2}+2h)\sum_{i=1}^{N}\sum_{q=k_0^i}^{k_0-1}(t_{q+1}-t_q)\hat{x}(t_q)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(t_q). \tag{4.20}$$

Substitute (4.15)-(4.17) to (4.14), and it follows from (4.12) that

$$V(t_{k+1})=V(t_0)-\sum_{i=1}^{N}\int_{t_0}^{t_{k+1}}x_i(w)L_i\hat{x}(w)dw$$
$$=V(t_0)-\sum_{i=1}^{N}\int_{t_0}^{t_1^i+\tau_1^i}x_i(w)L_i\hat{x}(w)dw-\sum_{i=1}^{N}\sum_{q=k_0^i}^{k}\int_{t_q}^{t_{q+1}}x_i(w)L_i\hat{x}(w)dw$$
$$\leq V(t_0)-\sum_{i=1}^{N}\int_{t_0}^{t_1^i+\tau_1^i}x_i(w)L_i\hat{x}(w)dw-\sum_{i=1}^{N}\sum_{q=k_0^i}^{k}\int_{t_q}^{t_{q+1}}\hat{x}_i(w)L_i\hat{x}(w)dw$$
$$+\sigma\sum_{i=1}^{N}\int_{t_0^i+\tau_0^i}^{t_1^i}\hat{x}(w)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(w)dw+\frac{3\sigma}{2}\sum_{i=1}^{N}\int_{t_1^i+\tau_1^i}^{t_{k+1}}\hat{x}(w)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(w)dw$$

$$
\begin{aligned}
&+2h\sum_{i=1}^{N}\sum_{q=k_0^i}^{k}(t_{q+1}-t_q)\hat{x}(t_q)^{\mathrm{T}}L_i^{\mathrm{T}}L_i\hat{x}(t_q) \\
=&V(t_0)+V_0-\sum_{i=1}^{N}\sum_{q=k_0}^{k}\int_{t_q}^{t_{q+1}}\hat{x}_i(w)L_i\hat{x}(w)dw \\
&+(\frac{3\sigma}{2}+2h)\sum_{q=k_0}^{k}(t_{q+1}-t_q)\hat{x}(t_q)^{\mathrm{T}}L^{\mathrm{T}}L\hat{x}(t_q) \\
=&V(t_0)+V_0-(1-\lambda_N(2h+\frac{3\sigma}{2}))\sum_{q=k_0}^{k}(t_{q+1}-t_q)\hat{x}(t_q)^{\mathrm{T}}L\hat{x}(t_q).
\end{aligned}
\tag{4.21}
$$

The condition in (4.11) and the fact that $V(t)\geq 0$ gives

$$\lim_{k\to\infty}(t_{k+1}-t_k)\hat{x}(t_k)^{\mathrm{T}}L\hat{x}(t_k)=0.$$

Note that t_k and t_{k+1} are consecutive elements in the set $\{t_k^i+\tau_k^i\}$ for $i=1,2,\ldots,N$ and $k=0,1,\ldots$. Hence $t_{k+1}-t_k$ is bounded. We have

$$\lim_{k\to\infty}\hat{x}(t_k)^{\mathrm{T}}L\hat{x}(t_k)=0.$$

Moreover, we obtain that $0\leq\hat{x}(t_k)^{\mathrm{T}}L^2\hat{x}(t_k)\leq\lambda_N\hat{x}(t_k)^{\mathrm{T}}L\hat{x}(t_k)$ because L is a positive semi-definite matrix. Therefore, $\lim_{k\to\infty}\hat{x}(t_k)^{\mathrm{T}}L^2\hat{x}(t_k)=0$, and $\lim_{k\to\infty}L\hat{x}(t_k)=0$. From (4.4), it implies that $\lim_{t\to\infty}\dot{x}(t)=0$. Combining (4.17), we have

$$\lim_{k\to\infty}\left|x_i(t_k^i)-\hat{x}_i(t_{k-1}^i+\tau_{k-1}^i)\right|\leq\sigma\sqrt{\frac{\int_{t_{k-1}^i+\tau_{k-1}^i}^{t_k^i}(L_i\hat{x}(w))^2dw}{h}}=0. \tag{4.22}$$

For any t, one can find an instant $t_{k_t}^i$ such that $t\in[t_{k_t}^i,t_{k_t+1}^i]$, and

$$\lim_{t\to\infty}(x_i(t_{k_t}^i)-\hat{x}_i(t_{k_t}^i+\tau_{k_t}^i))=\lim_{t\to\infty}(x_i(t_{k_t}^i)-x_i(t_{k_t}^i))=0,$$

if the triggering condition in (4.5) is satisfied; otherwise

$$\lim_{t\to\infty}(x_i(t_{k_t}^i)-\hat{x}_i(t_{k_t}^i+\tau_{k_t}^i))=\lim_{t\to\infty}(x_i(t_{k_t}^i)-\hat{x}_i(t_{k_t-1}^i+\tau_{k_t-1}^i))=0.$$

It follows that

$$\begin{aligned}
&\lim_{t\to\infty}\big(x_i(t)-\hat{x}_i(t)\big)\\
&\quad=\lim_{t\to\infty}\big(x_i(t)-x_i(t^i_{k_t})+x_i(t^i_{k_t})-\hat{x}_i(t)\big)\\
&\quad=\lim_{t\to\infty}\Big(\int_{t^i_{k_t}}^{t}\dot{x}_i(w)dw+x_i(t^i_{k_t})-\hat{x}_i(t^i_{k_t}+\tau^i_{k_t})\Big)=0.
\end{aligned}$$

Then, $\lim_{t\to\infty}(x(t)-\hat{x}(t))=0$, and $\lim_{t\to\infty}(Lx(t)-L\hat{x}(t))=0$. Due to the fact that $\lim_{k\to\infty}L\hat{x}(t_k)=0$, we have

$$\lim_{t\to\infty}Lx(t)=0.$$

Therefore $\lim_{t\to\infty}L\xi(t)=\lim_{t\to\infty}L\big(x(t)-\overline{x}(0)\mathbf{1}_N\big)=0$. By **Lemma 4.1**, one obtains $\xi^{\mathrm{T}}(t)\xi(t)\leq\frac{1}{\lambda_2}\xi^{\mathrm{T}}(t)L\xi(t)$ and $\lim_{t\to\infty}\xi(t)=0$, which implies that $\lim_{t\to\infty}x(t)=\lim_{t\to\infty}(\xi(t)+\bar{x}(0)\mathbf{1}_N)=\bar{x}(0)\mathbf{1}_N$. Thus, the average consensus of the agents is asymptotically reached. □

Remark 4.3. *As discussed in Section 4.3, by the proposed* Rotate&Compensate&Run Rendezvous Scheme, *x state evolution of a 2WMR is the same as the state evolution of the agent in the MAS* (4.4) *except for the rotating and state compensating time intervals. Accordingly, the above analysis can also be applied to the convergence analysis of rendezvous control method for 2WMRs. Besides, the proposed 2WMRs rendezvous control method has the following merits: 1. Using the integral-type triggering condition, the average performance of the agent is comprehensively considered from the most recent controller update instant to the event-checking instant, and the redundant information transmissions can be reduced; 2. A practical application scenario is studied with the time-varying delays in the controller design; 3. The proposed* Rotate&Compensate&Run Rendezvous Scheme *is simple yet very effective in practice, as verified by experiments in the next section.*

4.5 Experiment

In this section, we experimentally verify the effectiveness of the proposed rendezvous control methods on a group of 2WMRs. Figure 4.3 demonstrates the experimental setup. A cluster of reflective markers are mounted on each 2WMR. The unique configuration of the markers on the robot can be recognized and tracked as a rigid body by the OptiTrack™ cameras around the wall. The designed *Rotate&Compensate&Run*

Rendezvous Scheme is compiled on the Host PC and downloaded to each 2WMR microprocessor through wireless network. The information transmission among 2WMRs is conducted via WiFi.

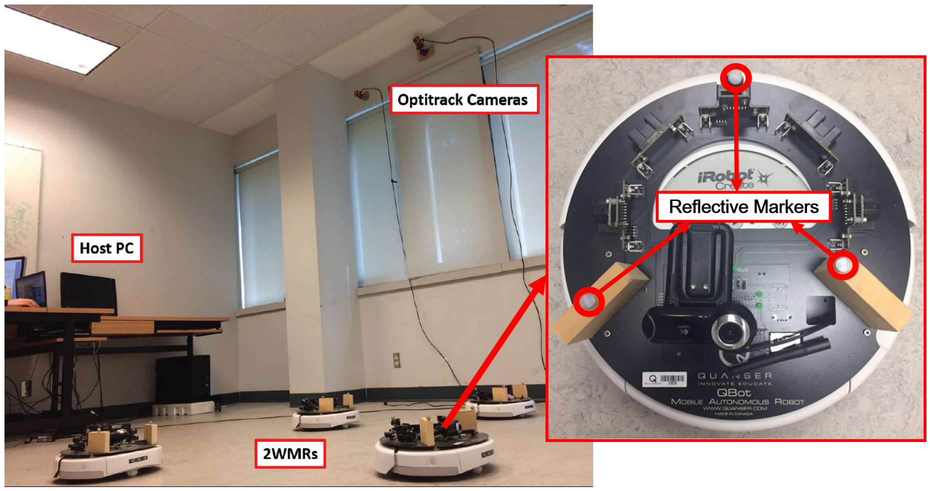


Figure 4.3: Experimental setup of 2WMRs.

We place four 2WMRs with random initial positions and headings in the two-dimensional workspace. The interaction graph is shown in Figure 4.4. The control gain $\tilde{a}_{ij}(t)$ in (4.2) is set as 0 for $t < \max\{t_0^i + \tau_0^i,\ t_0^j + \tau_0^j\}$, and $\tilde{a}_{ij}(t) = a_{ij}$ for $t \geq \max\{t_0^i + \tau_0^i,\ t_0^j + \tau_0^j\}$. We can tune the values of the event-checking period h, the triggering condition parameter σ and the adjacency matrix A such that the condition $2h + \frac{3}{2}\sigma < \frac{1}{\lambda_4}$ in (4.11) is satisfied. Here we choose $h = 4$sec, $\sigma = 2.4$ and $L = \begin{bmatrix} 0.05 & 0 & -0.025 & -0.025 \\ 0 & 0.025 & -0.025 & 0 \\ -0.025 & -0.025 & 0.05 & 0 \\ -0.025 & 0 & 0 & 0.025 \end{bmatrix}$ and the largest eigenvalue λ_4 is 0.0854.

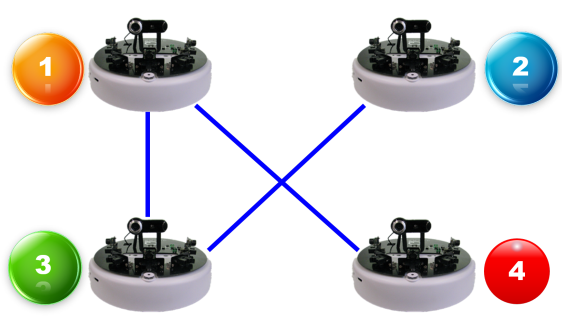


Figure 4.4: Interaction graph for 2WMRs.

The starting instants for 2WMRs are $t_0^1 = 0$sec, $t_0^2 = 1$sec, $t_0^3 = 2.2$sec and $t_0^4 = 3$sec. 2WMR i checks its triggering conditions in (4.5) and (4.10) at $t_0^i + kh$ with $k = 1, 2, \ldots$. The agent samples and broadcasts its x or y state if the triggering condition is satisfied. The time delay for the broadcast data is supposed to be upper bounded with 0.6sec such that $t_k^i + \tau_k^i < (k+1)h$. 2WMRs update the states using the proposed *Rotate&Compensate&Run Rendezvous Scheme*.

When the 2WMR receives the updated information or its broadcast data is received by the other agent, it calculates the desired orientation and the controller to be updated $v_i = \sqrt{(u_{x_i}(t))^2 + (u_{y_i}(t))^2}$. We set $v_l = -v_r$ when the robot is rotating in place until the 2WMR aims at the desired orientation. The rotating time is recorded as ΔT. Then set $v_l = v_r = v_i + \frac{v_i}{\alpha}$ during the following state compensating time period $\alpha\Delta T$, and $v_l = v_r = v_i$ before the next data update instant, as described in **Algorithm 4.1**. The left and right wheel velocity evolutions of a 2WMR are shown in Figure 4.5.

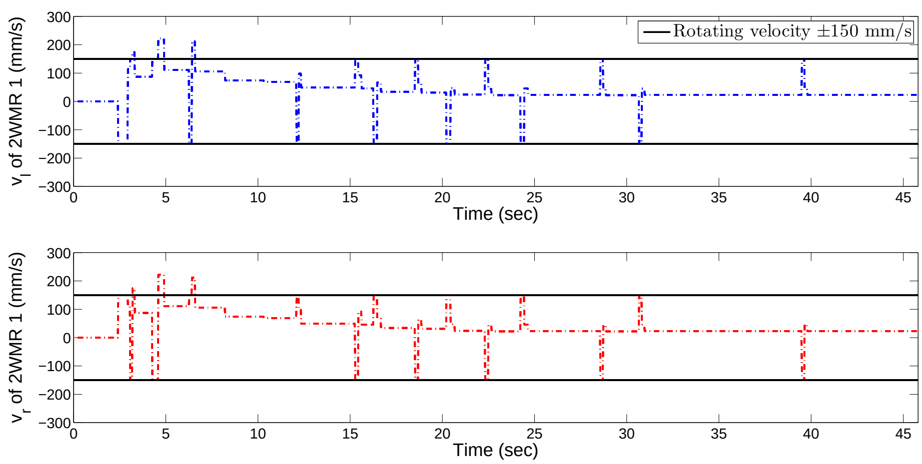


Figure 4.5: Experimental results of **Algorithm 4.1**: Time response of left and right wheel velocities of 2WMR 1.

The time response of x and y state evolutions of 2WMRs is shown in Figure 4.6, and the trajectories are given in Figure 4.7. Note that 2WMR positions in Figure 4.6 and 4.7 are represented by the geometry centers of the robots. Collision occur when 2WMRs are driven to reach rendezvous, and each 2WMR has the bumper to absorb the impact of the collision. Note that the robot is with a certain diameter such that their geometry centers can not overlap. From the experimental results, we see that each robot's edge touches at least one edge of the other robot with their final positions, and the rendezvous control for a group of 2WMRs is successful.

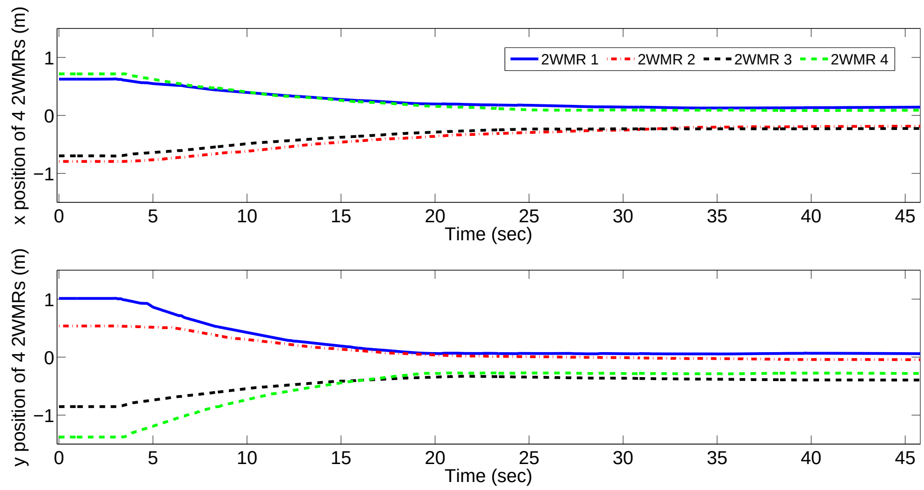


Figure 4.6: Experimental results of **Algorithm 4.1**: Time response of x and y positions of 2WMRs.

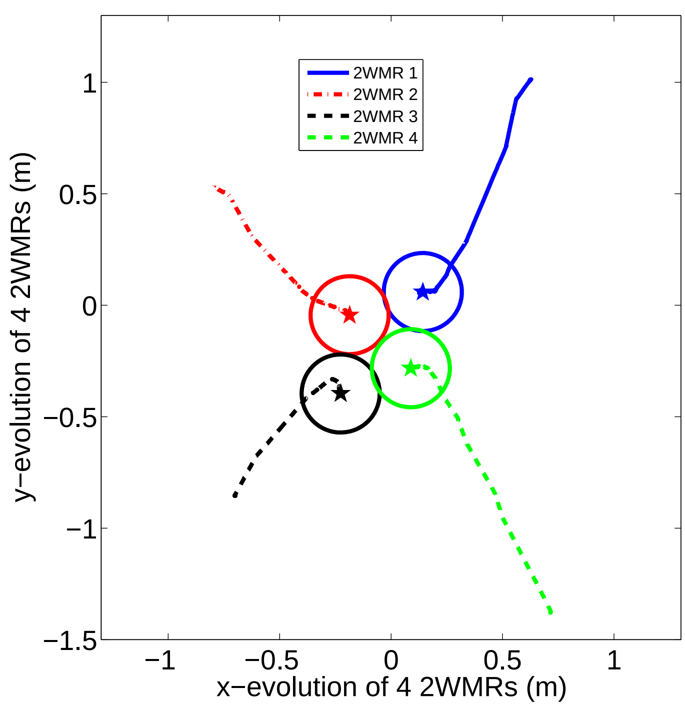


Figure 4.7: Experimental results of **Algorithm 4.1**: Trajectories of 2WMRs.

The sampling instants are shown as "$*$" in Figure 4.8, and with a communication time delay, the sampled data is received by the 2WMR's neighbors. Figure 4.9 demonstrates the triggering instants of 2WMRs. We observe that the robot does not need to sample the state at each triggering-checking instant, and the number of the sampling is reduced compared to the periodic sampling control strategy.

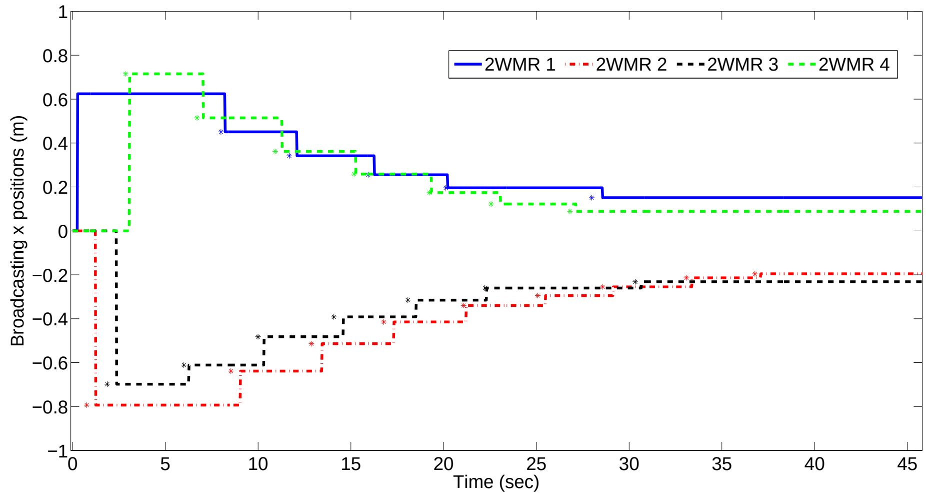


Figure 4.8: Experimental results of **Algorithm 4.1**: Time response of broadcasting x states of 2WMRs.

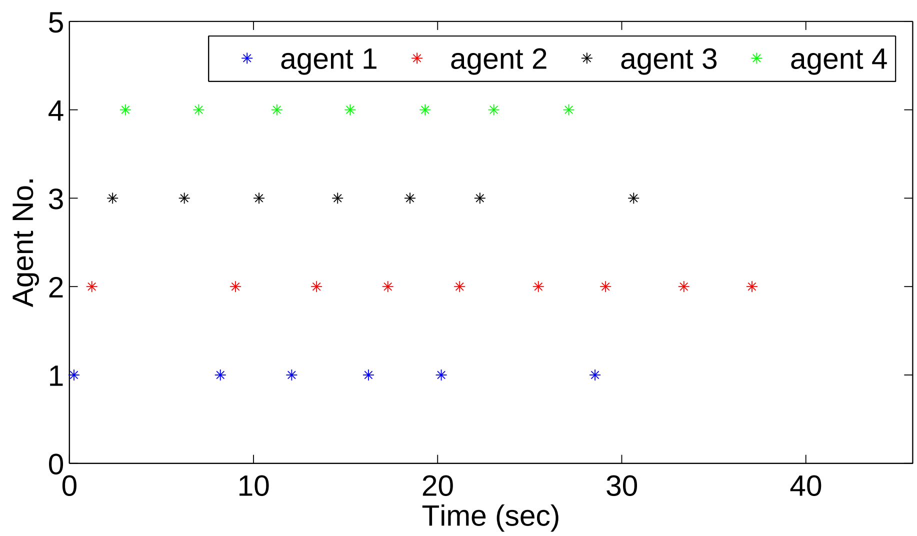


Figure 4.9: Experimental results of **Algorithm 4.1**: Event-triggering instants for x states of 2WMRs.

4.6 Conclusion

In this chapter, we investigated the event-triggered rendezvous control for a group of asynchronous 2WMRs with time-varying delays. 2WMRs periodically check the designed integral-type triggering conditions. We also provided a *Rotate&Compensate&Run Rendezvous Scheme* for the robots to update their states. The proposed control method was effectively applied to practical distributed 2WMRs. Future work will be focused on the rendezvous control for second-order and higher-order MASs.

A video of the experiment is posted on the following URL:
https://youtu.be/0Ojj6FpTODI

Chapter 5

Distributed LQR-based Consensus Control on Heterogeneous Multi-agent Systems

5.1 Introduction

In many studies of cooperative control of multi-agent systems, the agents are assumed to have identical dynamics. Examples are found in consensus problems for first-order dynamics [36], second-order dynamics [6], high-order dynamics [49], nonlinear dynamics [30], etc. However, in some situations, it is difficult to employ homogeneous MASs to cooperatively conduct the task. The robots in a heterogeneous MAS may not only have different dynamic model structures, but also have different working environments, capabilities and functions. Cooperative control of a heterogeneous MAS consisting of the first-order and second-order dynamics agents is considered in [54, 134, 135]. The authors in [55, 56] design consensus protocols for linear heterogeneous MASs. In this chapter, we formulate and solve a rendezvous problem for a group of vehicles consisting of two-wheeled mobile robots (2WMRs) and quadrotors. Some missions can be carried out with higher efficiency by employing both the autonomous ground vehicles and unmanned aerial vehicles; e.g., search and rescue, mine sweeping and so on.

The main contribution of this chapter is three-fold:

- The consensus control methods for a heterogeneous MAS are proposed. We consider the underactuation characteristic of the 2WMR dynamics and the physical constraint on the wheel velocity in designing the controller. An LQR-based *Rotate&Run Consensus Scheme* is provided. By using the proposed method, the motion control of the 2WMR in the two-dimensional plane is converted to controlling a system with 1-DOF. We also develop an LQR-based flight controller for the quadrotor and apply it to the consensus control of the heterogeneous MAS consisting of a quadrotor and 2WMRs.

- The dwell time after the information exchange time instant is employed to coordinate the motions of the heterogeneous MAS. Different from the continuous data sampling and broadcasting scheme in the existing literature [29], the agent only samples and broadcasts its state at the first instant in each dwell time. The desired position is calculated at the data interaction instant for the agent to approach until the next data interaction instant. Using the LQR-based controller, the agent cannot precisely reach the desired position in the finite dwell time, and thus a sufficient condition for the lower bound of the dwell time is given to guarantee consensus.

- The designed control methods are effectively implemented on the practical heterogeneous MAS.

The rest of the chapter is organized as follows: Section 5.2 reviews some preliminaries. Section 5.3 introduces the dynamic models of the agents, and then presents the control methods for the 2WMR and quadrotor respectively. Section 5.4 formulates the problem and describes the consensus algorithms for the heterogeneous MAS. Section 5.5 gives the convergence analysis of consensus. In Section 5.6, simulation examples are provided to verify the effectiveness of the proposed control methods and experimental testing is conducted and discussed in Section 5.7. Section 5.8 concludes the chapter.

5.2 Preliminaries

5.2.1 Infinite-time Linear Quadratic Regulator

Consider a linear time-invariant plant

$$\dot{X}(t) = AX(t) + Bu(t), \tag{5.1}$$

where $X(t) \in \mathbb{R}^m$ and $u(t) \in \mathbb{R}^p$ are the state and input of the system respectively. $A \in \mathbb{R}^{m\times m}$ and $B \in \mathbb{R}^{m\times p}$ are the constant state matrix and input matrix. The performance index is chosen as the following quadratic form:

$$J = \frac{1}{2}\int_0^{\infty} \left[X^{\mathrm{T}}(t)QX(t) + u^{\mathrm{T}}(t)Ru(t)\right] dt, \tag{5.2}$$

where $Q \in \mathbb{R}^{m\times m}$ is a symmetric, positive semi-definite matrix and $R \in \mathbb{R}^{p\times p}$ is a symmetric, positive definite matrix. Here we have the assumption that the pair (A, B) in (5.1) is stabilizable. To minimize the cost function in (5.2), we first solve the matrix Riccati equation:

$$PA + A^{\mathrm{T}}P + Q - PBR^{-1}B^{\mathrm{T}}P = 0. \tag{5.3}$$

Then the optimal control input is given as [136]

$$u(t) = -R^{-1}B^{\mathrm{T}}PX(t) = -KX(t). \tag{5.4}$$

It is shown that the optimal trajectory is obtained by solving

$$\dot{X}(t) = (A - BK)X(t), \tag{5.5}$$

and the system in (5.1) is asymptotically stable.

5.3 Dynamic Model and Control Strategies

In this section, we first introduce the system dynamics of the 2WMR and quadrotor, and then propose the LQR-based control methods for the agents to conduct the waypoint tracking task.

5.3.1 Dynamic Model of the 2WMR

The coordinate system of the 2WMR is presented in Section 3.3.1.

Suppose we have N_1 2WMRs in the MAS, and if the motions of the 2WMR along

x-axis and y-axis are independent, the decoupled dynamics of 2WMR i is written as

$$\begin{cases} \dot{X}_{1i}(t) = A_1 X_{1i}(t) + B_1 u_{1xi}(t), \\ \dot{Y}_{1i}(t) = A_1 Y_{1i}(t) + B_1 u_{1yi}(t), \end{cases} \tag{5.6}$$

where $X_{1i}(t) = [x_{1i}(t), v_{1xi}(t)]^{\mathrm{T}}$, $Y_{1i}(t) = [y_{1i}(t), v_{1yi}(t)]^{\mathrm{T}}$, $x_{1i}(t),\ y_{1i}(t),\ v_{1xi}(t),\ v_{1yi}(t),$ $u_{1xi}(t),\ u_{1yi}(t) \in \mathbb{R}, (i = 1, 2, \ldots, N_1)$ are the positions, velocities and control inputs of 2WMR i along x and y-axes respectively. $A_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, and $B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. Now we discuss the LQR-based control method for the 2WMR. Let Q_1 =diag$\{1, 0, 0, 0\}$ and $R_1 = 1$. By solving the following matrix Riccati equation as shown in (5.3):

$$P_1 A_1 + A_1^{\mathrm{T}} P_1 + Q_1 - P_1 B_1 R_1^{-1} B_1^{\mathrm{T}} P_1 = 0,$$

we get P_1. The optimal control inputs for the decoupled dynamics in (5.6) are then given by

$$\begin{cases} u_{1xi}(t) = -R_1^{-1} B_1^{\mathrm{T}} P_1 \left(X_{1i}(t) - \begin{bmatrix} \bar{x}_{1i}(\underline{t}) \\ 0 \end{bmatrix} \right), \\ u_{1yi}(t) = -R_1^{-1} B_1^{\mathrm{T}} P_1 \left(Y_{1i}(t) - \begin{bmatrix} \bar{y}_{1i}(\underline{t}) \\ 0 \end{bmatrix} \right), \quad \underline{t} \leqslant t < \bar{t}, \end{cases} \tag{5.7}$$

where $\bar{x}_{1i}(\underline{t})$ and $\bar{y}_{1i}(\underline{t})$ are the desired x and y positions for 2WMR i during the time interval $[\underline{t}, \bar{t})$. Substituting the control inputs in (5.7) into (5.6), we get the states of the system:

$$\begin{cases} \dot{X}_{1i}(t) = (A_1 - B_1 R_1^{-1} B_1^{\mathrm{T}} P_1) X_{1i}(t) + B_1 R_1^{-1} B_1^{\mathrm{T}} P_1 \begin{bmatrix} \bar{x}_{1i}(\underline{t}) \\ 0 \end{bmatrix}, \\ \dot{Y}_{1i}(t) = (A_1 - B_1 R_1^{-1} B_1^{\mathrm{T}} P_1) Y_{1i}(t) + B_1 R_1^{-1} B_1^{\mathrm{T}} P_1 \begin{bmatrix} \bar{y}_{1i}(\underline{t}) \\ 0 \end{bmatrix}, \\ \qquad \underline{t} \leqslant t < \bar{t}. \end{cases} \tag{5.8}$$

However, it is noticed that two wheels of the 2WMR are mounted on a common axis, and the wheels cannot have an acceleration along the connecting axis. The 2WMR is the so-called underactuated system which has a less number of the actuators than its DOF. The control protocols in (5.7) need to be modified. Figure 3.2 shows the body fixed frame (B-frame) of the 2WMR. x'-axis points to the forward and y'-

axis points to the left along the wheels' connecting axis. The plane is divided into four infinite regions by the body frame axes. We use the signs of the two coordinates (x', y') to describe the regions: Quadrant I $(+,\ -)$; II $(+,\ +)$; III $(-,\ +)$ and IV $(-,\ -)$, and the shaded region denotes Quadrant I in Figure 3.2. v_l and v_r denote the left and right wheel velocities. The 2WMR motions on x and y-axes in Figure 3.1 are determined by changing the velocities of each wheel of the 2WMR.

Based on the designed control inputs in (5.7), now we are ready to propose a *Rotate&Run Control Scheme* for the 2WMR to conduct the waypoint tracking task. The steps are as follows: At time instant $\underline{t}$, we update the desired position $\bar{x}_{1i}(\underline{t})$ and $\bar{y}_{1i}(\underline{t})$ of 2WMR i for $t \in [\underline{t}, \bar{t})$, and then solve the differential equations in (5.8) to obtain the states $(x_{1i}(\bar{t}), y_{1i}(\bar{t}))$ and $(v_{1xi}(t), v_{1yi}(t))$, $\underline{t} \leqslant t < \bar{t}$. Then we calculate the rotation direction for agent i's nose or tail to aim at $(x_{1i}(\bar{t}), y_{1i}(\bar{t}))$. After rotating in place to reach the goal heading, agent i moves in a straight line to $(x_{1i}(\bar{t}), y_{1i}(\bar{t}))$ with the velocity $v(t) = \sqrt{v_{1xi}^2(t) + v_{1yi}^2(t)}$ if $v(t) < v_{\max}$, otherwise $v(t) = v_{\max}$. Note that by using the proposed control scheme, the time interval $[\underline{t}, \bar{t})$ is divided into $[\underline{t}, \underline{t} + \Delta T)$ for the 2WMR to rotate in place and the time interval $[\underline{t} + \Delta T, \bar{t})$ for the 2WMR to update the states. By considering the rotating time ΔT and the physical constraints of the wheel velocity $\|v_l\| \leqslant v_{max}$ and $\|v_r\| \leqslant v_{max}$, the actual position of agent i at time instant $\bar{t}$ may not be at $(x_{1i}(\bar{t}), y_{1i}(\bar{t}))$, but it will apparently on the line segment from $(x_{1i}(\underline{t}), y_{1i}(\underline{t}))$ to $(x_{1i}(\bar{t}), y_{1i}(\bar{t}))$. The motion control of the 2WMR in the two-dimensional plane can be converted to controlling a system with 1-DOF.

5.3.2 LQR-based Flight Controller design for a Quadrotor

Dynamic Model of the Quadrotor

The quadrotor is a multirotor helicopter, and the frame structure is shown in Figure 5.1. Four propellers are mounted on the crossbeam with equal distances from the center of mass of the quadrotor. Propeller rotation axes are fixed and parallel. In order to generate upward lifts, four propellers have fixed-pitch blades with air flows pointing downward. The front and rear propellers are labeled as 1 and 3, which rotate clockwise, while the left and right propellers are labeled as 2 and 4, and rotate counter-clockwise. If four propellers spin at the same angular velocity, the net aerodynamics torque is balanced so that the yaw angular velocity is 0. The forces generated by four propellers are labeled as f_i in Figure 5.1.

In this work, the quadrotor is regarded as a rigid body and the states including

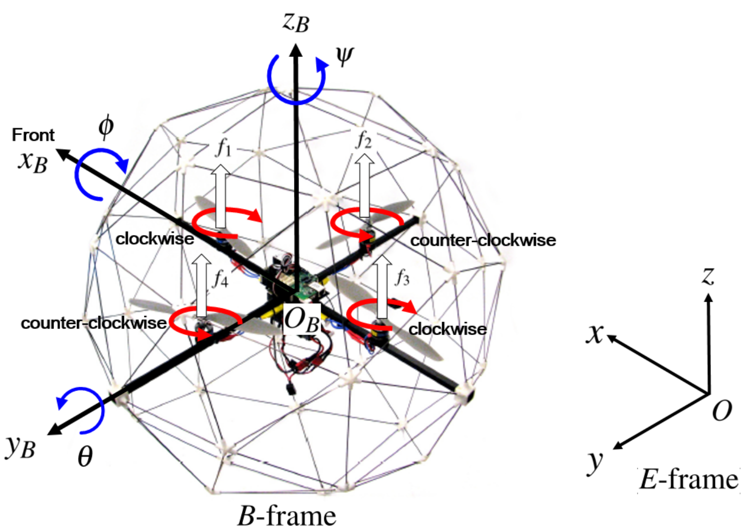


Figure 5.1: Coordinate systems of the quadrotor.

the positions and orientation evolve in a three dimensional space. The coordinate system of the quadrotor in the earth fixed frame (E-frame) is defined as

$$\Gamma_E = [x, y, z]^{\mathrm{T}}, \ \Theta_E = [\phi, \theta, \psi]^{\mathrm{T}}, \ \xi = \left[\Gamma_E^{\mathrm{T}}, \Theta_E^{\mathrm{T}}\right]^{\mathrm{T}},$$

where x, y and z are positions of the center of mass of the quadrotor; ϕ, θ and ψ are roll, pitch and yaw angles which denote the rotation around x, y and z-axes respectively.

We assume that the origin of the body fixed frame (B-frame) is the center of mass of the quadrotor. The positive direction of x_B and y_B are along the crossbeam pointing to the front and left directions of the quadrotor respectively. z_B-axis is vertical to the crossbeam plane and points to the top of the quadrotor. The velocities and angular velocities of the quadrotor in B-frame are as follows:

$$v_B = [v_{Bx}, v_{By}, v_{Bz}]^{\mathrm{T}}, \ \omega_B = [p, q, r]^{\mathrm{T}}, \ \eta_B = \left[v_B^{\mathrm{T}}, \omega_B^{\mathrm{T}}\right]^{\mathrm{T}}.$$

By using the Newton-Euler approach to describe the dynamics, we have [137, 138]

$$\dot{\xi} = J_\Theta \eta_B, \tag{5.9}$$

where $J_\Theta = \begin{bmatrix} R_{BE} & \mathbf{0_{3\times 3}} \\ \mathbf{0}_{3\times 3} & T_{BE} \end{bmatrix}$, R_{BE} is the rotation matrix from B-frame to E-frame.

$R_{BE} = \begin{bmatrix} c_\theta c_\psi & c_\psi s_\theta s_\phi - c_\phi s_\psi & s_\phi s_\psi + c_\phi c_\psi s_\theta \\ c_\theta s_\psi & c_\phi c_\psi + s_\theta s_\phi s_\psi & c_\phi s_\theta s_\psi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix}$. Since R_{BE} is orthogonal, the rotation matrix from E-frame to B-frame is $R_{BE}^{-1} = R_{BE}^{\mathrm{T}}$. $T_{BE} = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \frac{s_\phi}{c_\theta} & \frac{c_\phi}{c_\theta} \end{bmatrix}$, in which s_a =sin(a), c_a =cos(a), t_a =tan(a).

The dynamics of the quadrotor can be written as

$$\begin{aligned} \begin{bmatrix} mI_3 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \dot{v}_B \\ \dot{\omega}_B \end{bmatrix} + \begin{bmatrix} \omega_B \times (mv_B) \\ \omega_B \times (\boldsymbol{I}\omega_B) \end{bmatrix} \\ = \begin{bmatrix} F_B + R_{BE}^{\mathrm{T}} G \\ \tau_B \end{bmatrix}, \end{aligned} \tag{5.10}$$

where m is the total mass of the quadrotor. $F_B = [0, 0, F]^{\mathrm{T}}$, $F = f_1 + f_2 + f_3 + f_4$ is the total thrust generated by four propellers. $G = [0, 0, -mg]^{\mathrm{T}}$, g is acceleration due to gravity. $\tau_B = [\tau_\phi, \tau_\theta, \tau_\psi]^{\mathrm{T}}$ is the torques vector. $\boldsymbol{I} = \begin{bmatrix} I_{XX} & I_{XY} & I_{XZ} \\ I_{YX} & I_{YY} & I_{YZ} \\ I_{ZX} & I_{ZY} & I_{ZZ} \end{bmatrix}$ is the inertia matrix with I_{XX} as the moment of inertia around x_B-axes when the quadrotor rotates around x_B-axes; I_{XY} is the moment of inertia around y_B-axis when the quadrotor rotates around x_B-axis, and so on. In this work, the axes of B-frame and E-frame are assumed to coincide, which implies that $\boldsymbol{I}$ is diagonal. It is also assumed that the orientation angles are small so that the centrifugal force $\omega_B \times (mv_B)$ and the centripetal force $\omega_B \times (\boldsymbol{I}\omega_B)$ are considered small and thus they are omitted.

From (5.9) and (5.10), the linearized dynamics of the quadrotor can be written as

$$\begin{cases} \ddot{x} = g\theta, \ddot{y} = -g\phi, \ddot{z} = \dfrac{F}{m} - g, \\ \ddot{\phi} = \dfrac{\tau_\phi}{I_{XX}}, \ddot{\theta} = \dfrac{\tau_\theta}{I_{YY}}, \ddot{\psi} = \dfrac{\tau_\psi}{I_{ZZ}}. \end{cases} \tag{5.11}$$

Actuator Modeling

We use the following first-order dynamics to model the thrust $f_i,\ i = 1, 2, 3, 4$ generated by each propeller

$$f_i = K_m v = K_m \frac{\omega}{s + \omega} u_i, \tag{5.12}$$

where u_i is the pulse width modulation (PWM) input to the actuator, ω represents the actuator bandwidth and K_m denotes a positive gain. These parameters are determined, and verified by experimental studies [1]. Force changes in the motor pair (1, 3)/(2, 4) is to generate pitch/roll torque. Suppose $u = \tilde{u}$ is the control input for one propeller to increase/decrease the force, and the other paired motor generates an equal and opposite force. The net result will be a torque, and the thrust force for hovering the qaudrotor is ignored. The difference of the forces in the motor pair is determined by the difference of the motor inputs $\Delta u = 2\tilde{u}$. We define four control inputs of the system as u_{th}, u_θ, u_ϕ and u_ψ. Then we have

$$\begin{cases} F_B = 4K_m \dfrac{\omega}{s+\omega} u_{th}, \ \tau_\theta = 2K_m l \dfrac{\omega}{s+\omega} u_\theta, \\ \tau_\phi = 2K_m l \dfrac{\omega}{s+\omega} u_\phi, \ \tau_\psi = 2K_n u_\psi, \end{cases} \tag{5.13}$$

where K_n is a positive gain and l is the distance between the actuator and the center of mass of the quadrotor.

Quadrotor Flight Controller Design

In this section, we present the inner-outer loop structured controller based on the modelings of the quadrotor and the actuator.

By using (5.11) and (5.13), the decoupled dynamics along x, y and z-axes are written as

$$\begin{cases} \dot{\tilde{X}} = A_x \tilde{X} + B_x \theta, \\ \dot{\tilde{X}}_\theta = A_\theta \tilde{X}_\theta + B_\theta u_\theta, \end{cases} \tag{5.14a}$$

$$\begin{cases} \dot{\tilde{Y}} = A_y \tilde{Y} + B_y \phi, \\ \dot{\tilde{Y}}_\phi = A_\phi \tilde{Y}_\phi + B_\phi u_\phi, \end{cases} \tag{5.14b}$$

$$\dot{\tilde{Z}} = A_z \tilde{Z} + B_z u_{th} + \Omega, \tag{5.14c}$$

$$\dot{\Psi} = A_\psi \Psi + B_\psi u_\psi, \tag{5.14d}$$

where $\tilde{X} = [x, \dot{x}]^{\mathrm{T}}$, $\tilde{Y} = [y, \dot{y}]^{\mathrm{T}}$, $\tilde{Z} = [z, \dot{z}, v_z]^{\mathrm{T}}$, $\tilde{X}_\theta = [\theta, \dot{\theta}, v_\theta]^{\mathrm{T}}$, $\tilde{Y}_\phi = [\phi, \dot{\phi}, v_\phi]^{\mathrm{T}}$, $\Psi = [\psi, \dot{\psi}]^{\mathrm{T}}$, $A_x = A_y = A_\psi = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $A_\theta = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{2K_m l}{I_{YY}} \\ 0 & 0 & -\omega \end{bmatrix}$, $A_\phi = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{2K_m l}{I_{XX}} \\ 0 & 0 & -\omega \end{bmatrix}$, $A_z =$

$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{4K_m}{m} \\ 0 & 0 & -\omega \end{bmatrix}$, $B_\theta = B_\phi = B_z = [0, 0, \omega]^{\mathrm{T}}$, $B_x = [0, g]^{\mathrm{T}}$, $B_y = [0, -g]^{\mathrm{T}}$, $B_\psi = \left[0, \frac{4K_n}{I_{ZZ}}\right]^{\mathrm{T}}$ and $\Omega = [0, -g, 0]^{\mathrm{T}}$.

In order to track the desired x and y positions, the reference signals of θ and ϕ are calculated in the outer loop in (5.14a) and (5.14b), respectively. Control inputs u_θ and u_ϕ are to be designed to track the reference angles θ and ϕ [1], as shown in Figure 5.2.

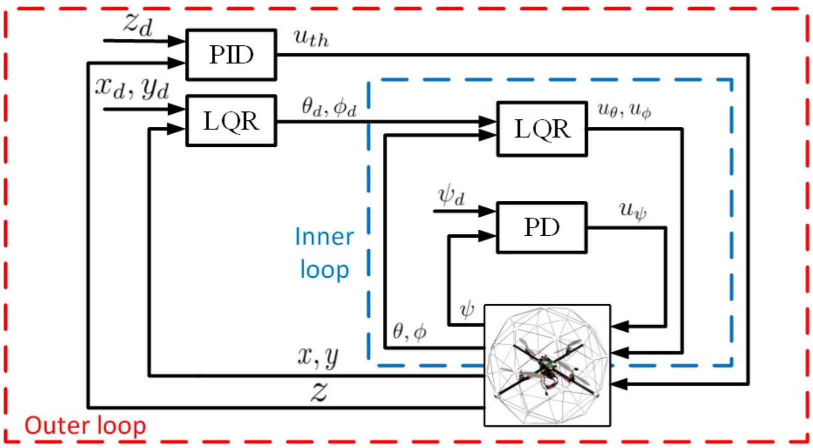


Figure 5.2: LQR-based inner-outer loop control scheme.

Suppose we have N_2 quadrotors in the MAS. Considering (5.14), the dynamics of the quadrotor i along x-axis and y-axis are linearized as

$$\begin{cases} \dot{X}_{2i}(t) = A_2 X_{2i}(t) + B_2 u_{2xi}(t), \\ \dot{Y}_{2i}(t) = \bar{A}_2 Y_{2i}(t) + B_2 u_{2yi}(t), \end{cases} \tag{5.15}$$

where $A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & g & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{2K_m l}{I_{XX}} \\ 0 & 0 & 0 & 0 & -\omega \end{bmatrix}$, $\bar{A}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -g & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{2K_m l}{I_{YY}} \\ 0 & 0 & 0 & 0 & -\omega \end{bmatrix}$ and $B_2 = [0, 0, 0, 0, \omega]^{\mathrm{T}}$. $X_{2i}(t) = \left[x_{2i}(t), \dot{x}_{2i}(t), \theta_i(t), \dot{\theta}_i(t), p_i(t)\right]^{\mathrm{T}}$, $Y_{2i}(t) = \left[y_{2i}(t), \dot{y}_{2i}(t), \phi_i(t), \dot{\phi}_i(t), q_i(t)\right]^{\mathrm{T}}$. $x_{2i}(t)$ and $y_{2i}(t)$ are positions of quadrotor i $(i = 1, 2, \ldots, N_2)$ along x and y-axes respectively. p_i and q_i are the variables representing the actuator dynamics. $u_{2xi}(t)$ and $u_{2yi}(t)$ are the control inputs for pitch and roll motor pairs. It is readily verified that the pairs

(A_2, B_2) are $(\bar{A}_2, B_2)$ are both stabilizable.

The height dynamics of the quadrotor is affected by the total thrust of the four propellers and the gravity

$$M\ddot{z}(t) = \sum_{i=1}^{4} f_i \cos(\phi_i(t)) \cos(\theta_i(t)) - mg. \tag{5.16}$$

As shown in Figure 5.2, the proportional-integral-derivative (PID) and proportional-derivative (PD) controllers are designed for the height and yaw angle controls respectively. More details of the quadrotor dynamics can be found in [1, 137].

5.4 Problem Formulation

The knowledge of directed graph is introduced in Section 3.2, which will be used in this work as well. Given a group of heterogeneous N agents of the form

$$\begin{cases} \dot{X}_i(t) = A_{xi}X_i(t) + B_{xi}u_{xi}(t), \\ \dot{Y}_i(t) = A_{yi}Y_i(t) + B_{yi}u_{yi}(t), \ i = 1, 2, \ldots, N. \end{cases} \tag{5.17}$$

where $X_i(t), Y_i(t) \in \mathbb{R}^{m_i}$ are the states, and $u_{xi}, u_{yi}(t) \in \mathbb{R}^{p_i}$ are the inputs of agent i along x and y-axes in the ground plane. The first elements in $X_i(t)$ and $Y_i(t)$ are $\tilde{x}_i(t)$ and $\tilde{y}_i(t)$, which denote the position of agent i along x and y-axes respectively. $A_i \in \mathbb{R}^{m_i \times m_i}$ and $B_i \in \mathbb{R}^{p_i}$. The pairs $(A_i, B_i), \forall i = 1, 2, \ldots, N$, are all stabilizable. Consensus of the MAS is reached if $\lim_{t\to\infty} \|\tilde{x}_i(t) - \tilde{x}_j(t)\| = 0, \ \lim_{t\to\infty} \|\tilde{y}_i(t) - \tilde{y}_j(t)\| = 0, \ \forall i, j = 1, 2, \ldots, N, i \neq j$.

In this work, the dwell time for the interaction topology is used to update the states. Only at the first time instant of each dwell time, $t_k, k = 0, 1, \ldots$, the agent samples and broadcasts data to other agents. Let $\mathcal{G}(t_k)$ and $\mathcal{N}_i(t_k)$ be the interaction graph and the neighbor set of agent i at instant t_k respectively. It is assumed that the switching topologies $\mathcal{G}(t_k), k = 0, 1, \ldots$ always have a spanning tree. The objective of this work aims to design control inputs for each agent to reach consensus. At instant t_k, agent i calculates

$$\bar{x}_i(t_k) = \frac{\tilde{x}_i(t_k) + \sum\limits_{j\in\mathcal{N}_i(t_k)} \tilde{x}_j(t_k)}{d_i(t_k) + 1}, \quad \bar{y}_i(t_k) = \frac{\tilde{y}_i(t_k) + \sum\limits_{j\in\mathcal{N}_i(t_k)} \tilde{y}_j(t_k)}{d_i(t_k) + 1}, \tag{5.18}$$

and then $(\bar{x}_i(t_k), \bar{y}_i(t_k))$ is set as the desired position for agent i to reach until the next interaction time instant t_{k+1}, where $d_i(t_k)$ is the number of neighbors of agent i at time instant t_k. Let $\bar{X}_i(t_k) = [\bar{x}_i(t_k), 0]^{\mathrm{T}}$, $\bar{Y}_i(t_k) = [\bar{y}_i(t_k), 0]^{\mathrm{T}}$ to be the desired states if agent i is a 2WMR, otherwise $\bar{X}_i(t_k) = [\bar{x}_i(t_k), 0, 0, 0, 0]^{\mathrm{T}}$, $\bar{Y}_i(t_k) = [\bar{y}_i(t_k), 0, 0, 0, 0]^{\mathrm{T}}$ for a quadrotor. Suppose the dynamics of each agent along x and y-axes are decoupled, in the dwell time interval $[t_k, t_{k+1})$, the designed controllers are applied, and the agent approaches its desired states $\bar{X}_i(t_k)$ and $\bar{Y}_i(t_k)$.

The control input of agent i along x-axis is then given as following: By choosing the proper Q_{xi} and R_{xi}, we solve the following matrix differential Riccati equation:

$$P_{xi}A_{xi} + A_{xi}^{\mathrm{T}}P_{xi} + Q_{xi} - P_{xi}B_{xi}R_{xi}^{-1}B_{xi}^{\mathrm{T}}P_{xi} = 0,$$

and then

$$u_{xi}(t) = -R_{xi}^{-1}B_{xi}^{\mathrm{T}}P_{xi}\left(X_i(t) - \bar{X}_i(t_k)\right), \ t_k \leqslant t < t_{k+1}. \tag{5.19}$$

Since (A_{xi}, B_{xi}) is stabilizable, the desired states $\bar{X}_i(t_k)$ will be reached asymptotically if the dwell time is infinite, i. e., $\lim_{t\to\infty} X_i(t) = \bar{X}_i(t_k)$. We have

$$X_i(t) - \bar{X}_i(t_k) = e^{\hat{A}_{xi}(t-t_k)}(X_i(t_k) - \bar{X}_i(t_k)), \ t_k \leqslant t < t_{k+1}, \tag{5.20}$$

where $\hat{A}_{xi} = A_{xi} - B_{xi}R_{xi}^{-1}B_{xi}^{\mathrm{T}}P_{xi}$, and all eigenvalues of $\hat{A}_{xi}$ have negative real parts. It is readily shown that

$$\begin{aligned} \|\tilde{x}_i(t_{k+1}) - \bar{x}_i(t_k)\| &\leqslant \left\|X_i(t_{k+1}) - \bar{X}_i(t_k)\right\| \\ &= \left\|e^{\hat{A}_{xi}(t_{k+1}-t_k)}\right\| \left\|(X_i(t_k) - \bar{X}_i(t_k))\right\|. \end{aligned} \tag{5.21}$$

The control input of agent i along y-axis can be obtained in the similar line. According to the dynamic models of the 2WMR and quadrotor introduced in Section 5.3, the designed LQR controller can only be applied to the quadrotor because the 2WMR dynamics is the underactuated. We propose the LQR-based *Rotate&Run Consensus Scheme* for the 2WMR in **Algorithm 1**.

By using the proposed control method, the position of the 2WMR will be on the line segment from $(\tilde{x}_i(t_k), \tilde{y}_i(t_k))$ to $(\bar{x}_i(t_k), \bar{y}_i(t_k))$ at time instant t_{k+1}. We have

$$\tilde{x}_i(t_{k+1}) = \tilde{x}_i(t_k) + \rho_i(t_k, t_{k+1})\left(\bar{x}_i(t_k) - \tilde{x}_i(t_k)\right), \tag{5.22}$$

Algorithm 5.1 The LQR-based *Rotate&Run Consensus Scheme* for 2WMR i ($i = 1, 2 \dots N_1$.)

Input: $x_{1i}(t_k),\ y_{1i}(t_k),\ \theta_i(t_k),\ \ \tilde{x}_j(t_k),\ \tilde{y}_j(t_k),\ \forall j \in \mathcal{N}_i(t_k)$, t.
Output: $v_l(t),\ v_r(t)$.

```
if  t = t_k then
    Substitute (x̄_i(t_k), ȳ_i(t_k)) into (5.8), and calculate (x_1i(t_{k+1}), y_1i(t_{k+1})).
    θ_iT = atan2(y_1i(t_{k+1}) − y_1i(t_k), x_1i(t_{k+1}) − x_1i(t_k)),
    v(t) = sqrt(v_1xi^2(t) + v_1yi^2(t)), for the case v(t) < v_max, otherwise v(t) = v_max.
    if (x_1i(t_{k+1}), y_1i(t_{k+1})) is in Quadrant I, then
        Rd= 1, Md= 1, θ_id(t_k) = θ_iT;
    else if (x_1i(t_{k+1}), y_1i(t_{k+1})) is in Quadrant II, then
        Rd= −1, Md=1, θ_id(t_k) = θ_iT;
    else if (x_1i(t_{k+1}), y_1i(t_{k+1})) is in Quadrant III, then
        Rd= 1, Md= −1;
        θ_id(t_k) = θ_iT ± 180°, θ_id(t_k) ∈ [−180°, 180°);
    else
        Rd= −1, Md= −1;
        θ_id(t_k) = θ_iT ± 180°, θ_id(t_k) ∈ [−180°, 180°).
    end if
end if
while t_k ⩽ t < t_{k+1} do
    if |θ_i(t) − θ_id(t_k)| > ε then
        v_l = v̄, v_r = −v̄ for the case Rd= 1 and
        v_l = −v̄, v_r = v̄ for the case Rd= −1.
    else
        v_l = v_i, v_r = v_i for the case Md= 1 and
        v_l = −v_i, v_r = −v_i for the case Md= −1.
    end if
end while
```

θ_{id}: The goal heading,
Rd$=1$: Rotating clockwise,
Rd$=-1$: Rotating counterclockwise,
Md$=1$: Moving forward when the nose of the 2WMR is pointing at $(x_{1i}(t_{k+1}), y_{1i}(t_{k+1}))$,
Md$=-1$: Moving backward when the tail of the 2WMR is pointing at $(x_{1i}(t_{k+1}), y_{1i}(t_{k+1}))$,
ε is the tolerance for the heading,
$\bar{v}$ satisfying $0 \leqslant \bar{v} \leqslant v_{max}$ is the constant wheel velocity for the 2WMR to rotate in place.

where $0 < \rho_i(t_k, t_{k+1}) < 1$ is a positive scalar which is determined by the length of the time interval $[t_k, t_{k+1})$.

Let $\hat{X}(t_k) = [\tilde{x}_1(t_k), \tilde{x}_2(t_k), \ldots, \tilde{x}_N(t_k)]^{\mathrm{T}}$, we have

$$\hat{X}(t_{k+1}) = \Phi(t_k)\hat{X}(t_k), k = 1, 2, \ldots \tag{5.23}$$

with
$\Phi(t_k) = I_N - \mathrm{diag}\{\frac{\rho_1(t_k,t_{k+1})}{d_1(t_k)+1}, \frac{\rho_2(t_k,t_{k+1})}{d_2(t_k,t_{k+1})+1}, \ldots, \frac{\rho_N(t_k,t_{k+1})}{d_N(t_k)+1}\}L(t_k)$.

5.5 Consensus Analysis

In this section, we study the condition for guaranteeing consensus of the heterogeneous MAS. We use the knowledge of *stochastic matrix*, which has been reviewed in Section 3.2.

Lemma 5.1 (Lemma 3.3, 3.7 & Corollary 3.5 in [29])**.** *A stochastic matrix $M \in \mathbb{R}^{n\times n}$ is SIA if all diagonal entries of M are positive and the associated directed graph of M has a spanning tree.*

Lemma 5.2 ([124])**.** *Let $\{S_1, S_2, \ldots, S_k\}$ be a set consisting of finite SIA matrices with the same dimension $n \times n$. Any sequence of matrix product $S_{i_m}S_{i_{m-1}}\ldots S_{i_1}$ of positive length is SIA. For the product $S_{i_m}S_{i_{m-1}}\ldots$ with infinite length, there exists a column vector c such that*

$$\lim_{m\to\infty} S_{i_m}S_{i_{m-1}}\ldots S_{i_1} = \mathbf{1}_n c^{\mathrm{T}}. \tag{5.24}$$

Lemma 5.3 (Theorem 6.1 in [139])**.** *Consider the system $\dot{x} = Ax$, with $A \in \mathbb{R}^{n\times n}$ as a constant matrix. If all eigenvalues of A have real negative parts, then for each $x(t_0) \in \mathbb{S}^n$, there exist positive constants H and μ such that*

$$\|x(t)\| \leqslant H\,\|x(t_0)\|\,e^{-\mu t},\ \forall t \geqslant 0. \tag{5.25}$$

In the sequel, the main result will be presented.

Theorem 5.1. *There exists a lower bound T_k for the dwell time $t_{k+1} - t_k \geqslant T_k$ such that $\Phi(t_k)$ in (5.23) is a stochastic matrix with positive diagonal entries. The heterogeneous MAS consisting of quadrotors and 2WMRs reaches consensus if $t_{k+1} -$*

$t_k \geqslant T_k$, and the switching interaction graphs $\mathcal{G}(t_k)$ have a directed spanning tree at time instant $t_k, k = 1, 2, \ldots$

Proof. Considering (5.23) and noting that $L(t_k)\mathbf{1}_N = \mathbf{0}_N$, it is shown that the ith ($i = 1, 2, \ldots, N$) row sum of $\Phi(t_k)$ is 1 and off-diagonal entries of $\Phi(t_k)$ are all nonnegative. The diagonal entries are then written as

$$s_i(t_k) = 1 - \frac{\rho_i(t_k, t_{k+1})d_i(t_k)}{d_i(t_k) + 1}. \tag{5.26}$$

To guarantee s_i ($i = 1, 2, \ldots, N$) to be positive, we require that

$$0 < \rho_i(t_k, t_{k+1}) < \frac{d_i(t_k) + 1}{d_i(t_k)}. \tag{5.27}$$

Eq. (5.22) can be further written as $\rho_i(t_k, t_{k+1}) = \frac{\tilde{x}_i(t_{k+1}) - \tilde{x}_i(t_k)}{\bar{x}_i(t_k) - \tilde{x}_i(t_k)}$, and then we have

$$0 < \frac{\tilde{x}_i(t_{k+1}) - \tilde{x}_i(t_k)}{\bar{x}_i(t_k) - \tilde{x}_i(t_k)} < \frac{d_i(t_k) + 1}{d_i(t_k)}. \tag{5.28}$$

We consider the following two cases respectively.
Case 1: $\tilde{x}_i(t_{k+1}) > \tilde{x}_i(t_k)$ and $\bar{x}_i(t_k) > \tilde{x}_i(t_k)$. It requires that $\tilde{x}_i(t_{k+1}) - \bar{x}_i(t_k) < \frac{\bar{x}_i(t_k) - \tilde{x}_i(t_k)}{d_i(t_k)}$ to make the inequality in (5.28) holds.
Case 2: $\tilde{x}_i(t_{k+1}) < \tilde{x}_i(t_k)$ and $\bar{x}_i(t_k) < \tilde{x}_i(t_k)$. Similarly, we need that $\bar{x}_i(t_k) - \tilde{x}_i(t_{k+1}) < \frac{\tilde{x}_i(t_k) - \bar{x}_i(t_k)}{d_i(t_k)}$ to guarantee (5.28). The sufficient condition for (5.28) to hold is

$$\|\bar{x}_i(t_k) - \tilde{x}_i(t_{k+1})\| < \frac{\|\bar{x}_i(t_k) - \tilde{x}_i(t_k)\|}{d_i(t_k)}. \tag{5.29}$$

Using (5.21) and **Lemma** 5.3, we have

$$\|\tilde{x}_i(t_{k+1}) - \bar{x}_i(t_k)\| \leqslant H_i \left\|(X_i(t_k) - \bar{X}_i(t_k))\right\| e^{-\mu_i(t_{k+1} - t_k - \frac{\pi}{2\bar{v}})}, \tag{5.30}$$

where $\bar{v}$ is the constant wheel velocity for the 2WMR to rotate in place, and $\frac{\pi}{2\bar{v}}$ is the maximum rotating time for a 2WMR to reach the goal heading. If we choose $t_{k+1} - t_k - \frac{\pi}{2\bar{v}} \geqslant T_{(i)k}$ appropriately for agent i, i.e.,

$$H_i \left\|(X_i(t_k) - \bar{X}_i(t_k))\right\| e^{-\mu_i(t_{k+1} - t_k - \frac{\pi}{2\bar{v}})} < \frac{\|\bar{x}_i(t_k) - \tilde{x}_i(t_k)\|}{d_i(t_k)}, \tag{5.31}$$

equivalently $t_{k+1} - t_k - \frac{\pi}{2\bar{v}} \geqslant T_{(i)k} = -\frac{1}{\mu_i} \ln \frac{\|\bar{x}_i(t_k) - \tilde{x}_i(t_k)\|}{d_i(t_k) H_i \left\| (X_i(t_k) - \bar{X}_i(t_k)) \right\|}$. It follows that (5.29) is satisfied, and the condition in (5.27) is guaranteed. Accordingly we choose the dwell time $T_K = t_{k+1} - t_k$ as

$$T_k = \max \left\{ T_{(1)k}, T_{(2)k}, \ldots, T_{(N)k} \right\} + \frac{\pi}{2\bar{v}} \tag{5.32}$$

such that $s_i(t_k), i = 1, 2, \ldots, N$ are all positive, which implies that $\Phi(t_k)$ is a stochastic matrix with positive diagonal entries.

From **Lemma** 5.1, it is shown that $\Phi(t_k)$ in (5.23) is SIA if the switching interaction graphs $\mathcal{G}(t_k)$ always have a directed spanning tree. Using **Lemma** 5.2, we have

$$\hat{X}(t_k) = \lim_{k \to \infty} \Phi(t_k)\Phi(t_{k-1}) \ldots \Phi(t_0)\hat{X}(t_0) = \mathbf{1}_N c_1^{\mathrm{T}} \hat{X}(t_0), \tag{5.33}$$

which implies that consensus is reached. □

5.6 Simulation

In this section, we conduct two sets of simulations under Matlab® environment to demonstrate the effectiveness of the designed consensus control methods for a group of heterogeneous MAS consisting of three 2WMRs and one quadrotor.

As shown in **Theorem** 5.1, the switching interaction topologies in the examples are directed and always have a spanning tree. The states of the agents are initialized with random values in the two-dimensional plane. At each information interaction time instant, an interaction topology is randomly chosen from Figure 5.3 and applied to the networked MAS. The quadrotor and 2WMRs calculate their desired positions using (5.18). In the first set of simulation, the initial distances among the agents are with the same scales as the example shown in the experimental testing, and the dwell time is set as 5 secs. Figures 5.4 and 5.5 show the simulation results of the position, velocity evolutions and trajectories of the agents. The second set of simulation is then conducted considering the MAS in a larger workspace, and the dwell time is set as 10 sec. The results are shown in Figures 5.6 and 5.7. In both sets of simulations, we observe that consensus can be reached. The 2WMRs' left wheel (LW) and right wheel (RW) velocities are within the bound of 0.5 m/sec. The trajectories of the agents in Figure 5.7 look different from the result as shown in Figure 5.5. This is because that the 2WMR's velocity upper bound is much smaller than the quadrotor's velocity such that the position evolution of the quadrotor is much larger than the 2WMR's in the fixed dwell time if the workspace is large.

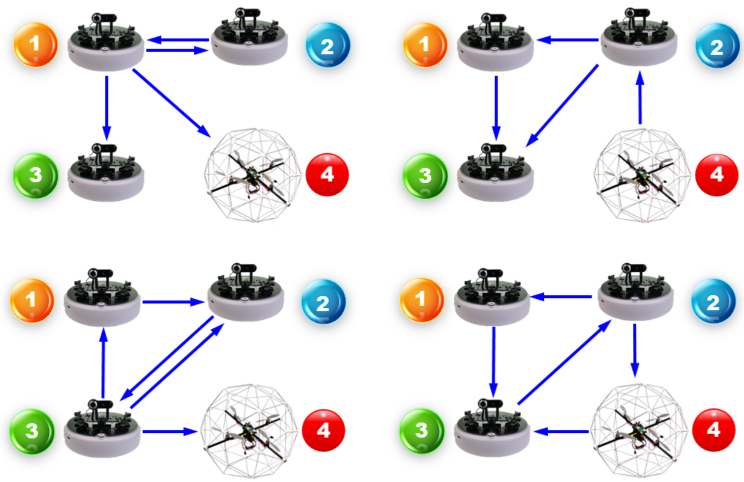


Figure 5.3: Interaction topologies with spanning trees.

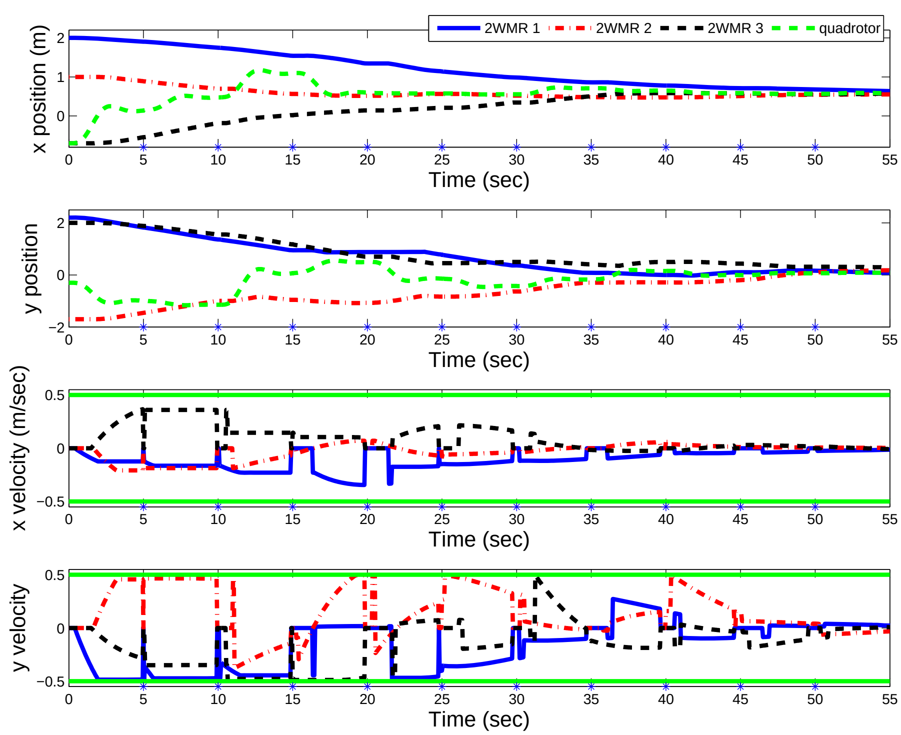


Figure 5.4: Simulation results of the proposed control methods: Time response of x and y positions and velocities of four agents.

Remark 5.1. *We prove that the dwell time lower bound T_k in* (5.32) *exists such that consensus can be reached. T_k can be calculated in the simulation/experiment, but this*

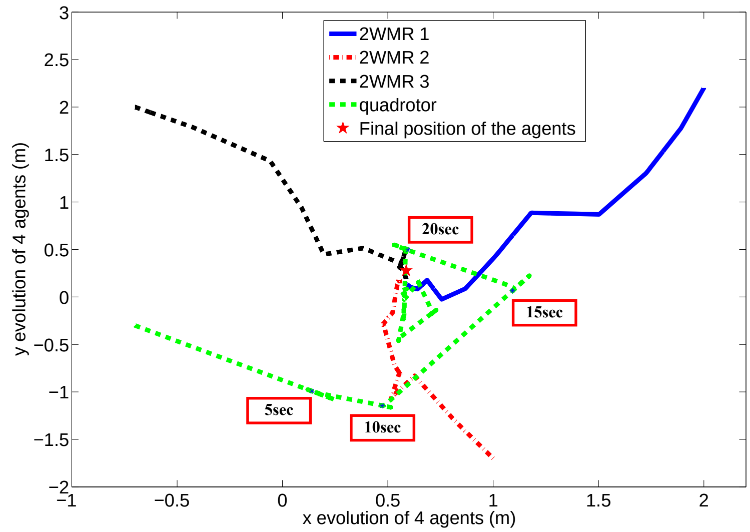


Figure 5.5: Simulation results of the proposed control methods: Trajectories of the agents on xy plane.

value cannot be practically set as the dwell time. The reason is that the communication of the networked agents is not "all-to-all" mode, which implies that max$\{T_{(i)k}\}$ *for* $i = 1, 2, \ldots, N$ *can not be known by all agents at one interaction instant. Since we conduct the simulations/experiments fitting the practical situations, an appropriate (might be conservative) dwell time for the specific application scenario is alternatively chosen.*

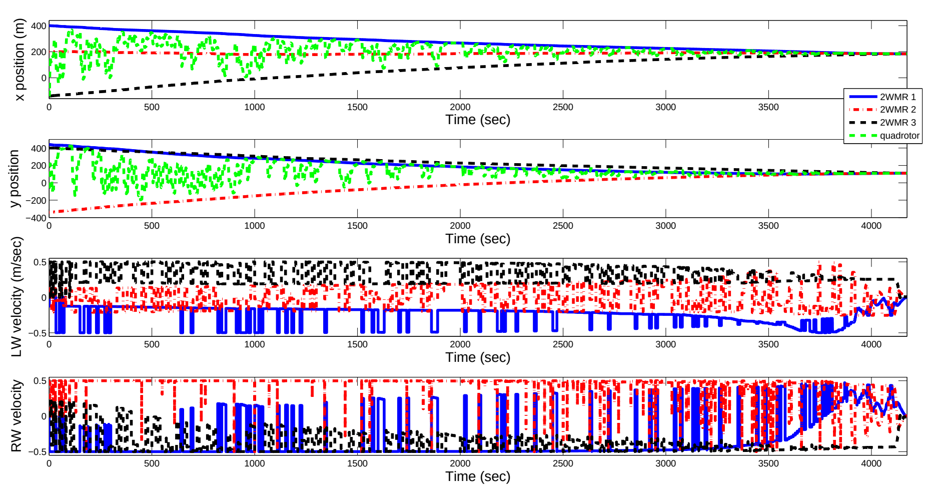


Figure 5.6: Simulation results of the proposed control methods: Time response of x and y positions and velocities of four agents in a larger workspace.

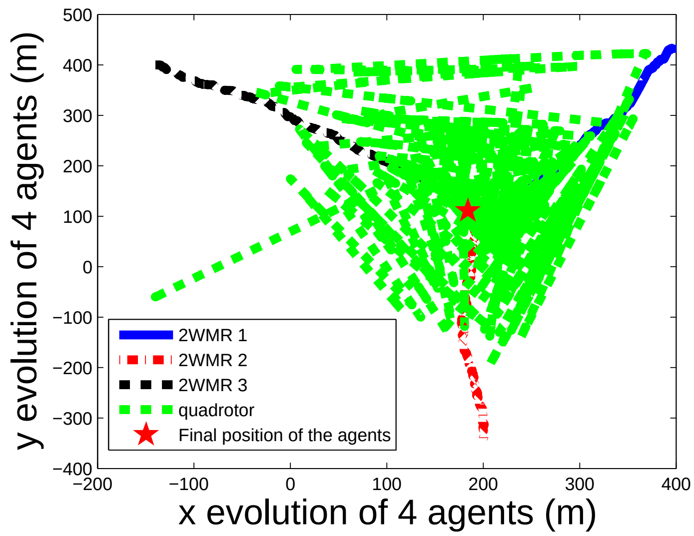


Figure 5.7: Simulation results of the proposed control methods: Trajectories of the agents on xy plane in a larger workspace.

5.7 Experiment

Figure 5.8 shows the experimental setup of the heterogeneous MAS consisting of 3 2WMRs and 1 quadrotor. A cluster of reflective markers are glued on each vehicle. The unique configuration of the markers on the vehicle is then recognized and tracked as a rigid body by OptiTrack$^{\text{TM}}$ cameras. The consensus protocols are compiled and then downloaded to the microcontroller Gumstix embedded on the vehicles. The wireless network is employed to transmit data among the MAS.

The quadrotor parameters in (5.15) are given as follows [1]: $K_m = 120\text{N}$, $K_n = 4\text{N}\cdot\text{m}$, $\omega = 15\text{rad/sec}$, $I_{XX} = I_{YY} = 0.03\text{kg}\cdot\text{m}^2$, $I_{ZZ} = 0.04\text{kg}\cdot\text{m}^2$, $m = 1.4\text{kg}$ and $l = 0.2\text{m}$.

2WMRs and quadrotor are placed at random positions in the workspace with zero initial velocities. The MAS starts to conduct the consensus behavior when the quadrotor reach a safety height 0.5m after taking off vertically. At time instant t_k, the interaction topology containing a spanning tree may change or not. The dwell time $t_{k+1} - t_k$, $(k = 0, 1, 2, \ldots)$ described in Section 5.4 is given as 5secs in the experiment. agent i calculates $\bar{x}_i(t_k)$ and $\bar{y}_i(t_k)$ using (5.18). 2WMRs then evolve the states by following **Algorithm 1**, and the quadrotor updates its states according to the LQR consensus protocol in (5.19). The x and y position evolutions of the agents are shown in Figure 5.9. The LW and RW velocities of 2WMRs are also presented, from which we see that the wheel velocity constraints $\|v_l\| \leqslant v_{max} = 0.5\text{m/sec}$,

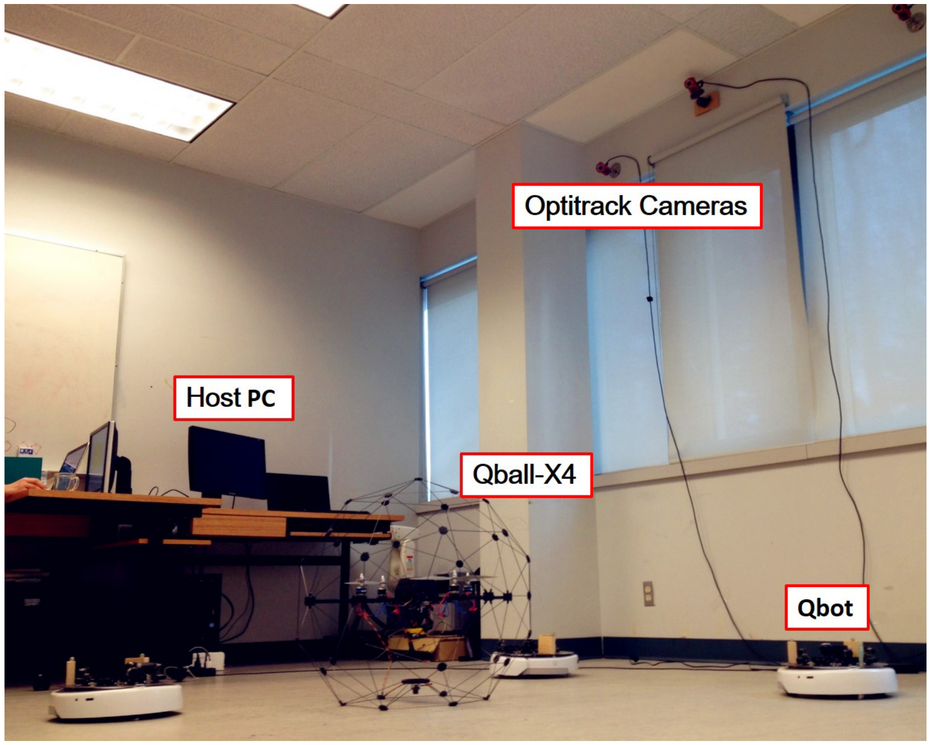


Figure 5.8: Experimental setup of the heterogeneous MAS.

$\|v_r\| \leqslant v_{max} = 0.5$m/sec are satisfied. The collision occurs when 2WMRs converge to the same position, and each 2WMR has the bumper to absorb the impact of the collision. The position evolutions in Figure 5.9 and trajectories of the agents in Figure 5.10 demonstrate the motions of the geometry centers of the agents.

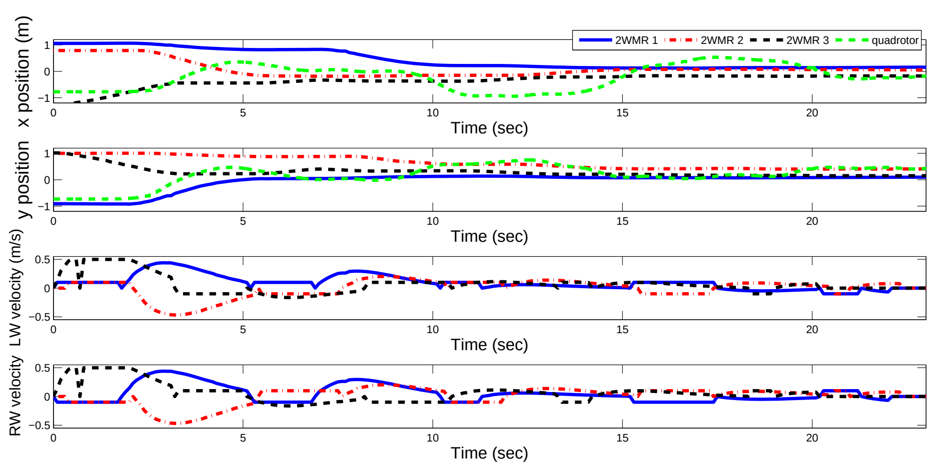


Figure 5.9: Experimental results of the proposed control methods: Time response of x, y positions and wheel velocities of four agents.

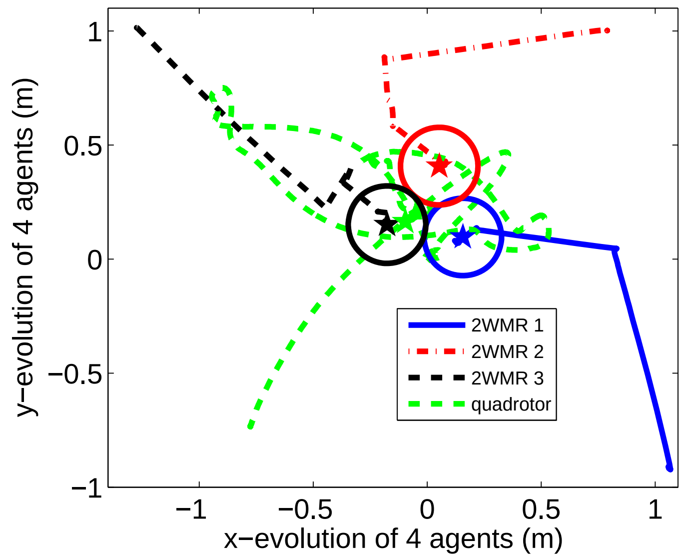


Figure 5.10: Experimental results of the proposed control methods: Trajectories of the agents on xy plane.

Note that the quadrotor dynamics in (5.15) assume that roll, pitch and yaw angles are small (less than 6° in practical application). Figure 5.11 shows the evolutions of the roll, pitch and yaw angles of the quadrotor. The motion on x/y-axis is controlled by the total thrust of the propellers and the change of the roll/pitch angle. A PWM with period 20ms is employed to control each motor, and there exists a constraint on the PWM duty cycle: 5% to 10%, [1]. Figure 5.12 shows the evolutions of the PWM duty cycles of the motors.

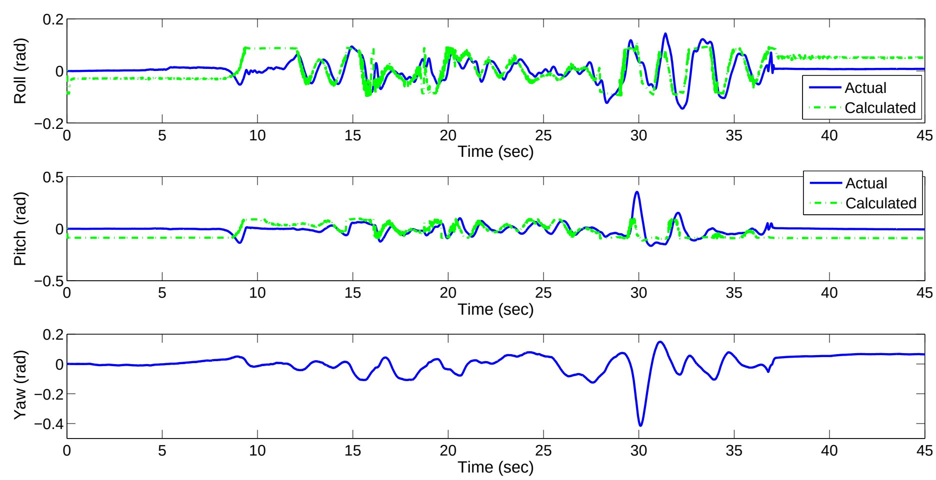


Figure 5.11: Experimental results of the proposed control methods: Time response of roll, pitch and yaw angles of the quadrotor.

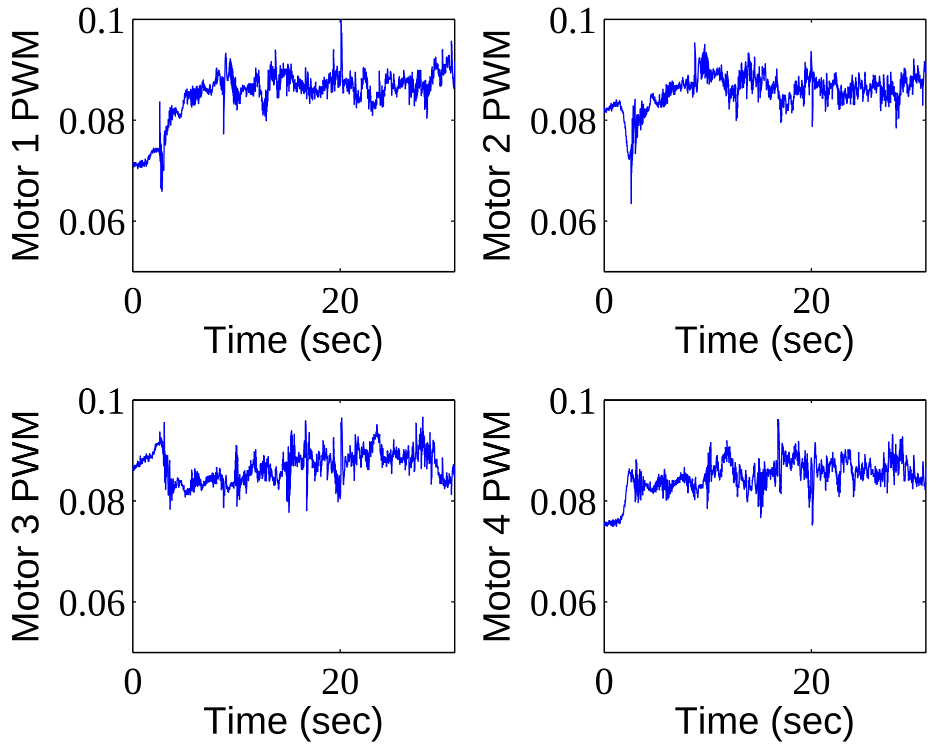


Figure 5.12: Experimental results of the proposed control methods: Time response of the PWM duty cycles of the motors.

Remark 5.2. *Both simulation and experiment are approaches for verifying the effectiveness of the controllers. The differences between two approaches lie in the following aspects: As the diameter of each 2WMR is 34cm, the final distances among the agents obtained in the experimental tests are larger than the results of simulations. From the engineering standpoint, consensus of the heterogeneous MAS is reached. In the simulations, the quadrotor changes its flying direction sharply at the information interaction time instant, and the trajectories are connected by the straight lines. This is the ideal control result for the quadrotor. However we observe that in Figure 5.11, there exist few instants, at which roll/pitch angle peaks are relatively large (one reaches 22°). The reason is that we use the linearized quadrotor dynamics for the flight controller design, and there might exist model uncertainties and external disturbances. Accordingly the quadrotor can not strictly follow the desired straight lines. It hovers around the desired position. The trajectories of the quadrotor are smooth. We observe that consensus of the heterogeneous MAS is reached in both simulation and experiments.*

Remark 5.3. *In the experimental testing, we use the ad-hoc WiFi network to transmit* *the data among the agents. The standard for the communication is IEEE 802.11b, and the working range of the WiFi adapter is within 50 meters. This work focuses on the practical application of cooperative control for the MAS, and the experiment in the large workspace is not conducted due to constraints such as the limited workspace and communication ranges, the battery life of the quadrotor and the wind disturbances in the outdoor environment. The effectiveness of the control methods for the application scenario with large workspace is validated through the simulation.*

5.8 Conclusion

In this chapter, the consensus problem for a heterogeneous MAS with directed and switching interaction topologies was investigated. We developed an LQR consensus protocol for the quadrotors and proposed an LQR-based consensus algorithm for the underactuated 2WMRs respectively. Experimental testing was conducted on the real-time MAS platform. The experiment results revealed that the proposed control methods are effective in solving the consensus problem for the heterogeneous MAS. Future research will be considered to conduct the experiment in a larger worksapce. Interesting extensions of this work can also be pursued in considering sensor failures in the vehicle dynamics [140], obstacle avoidance during cooperative control of the MAS and designing novel flight controller using the model predictive control (MPC) technique [141, 142].

A video of the experiment is posted on the following URL:
https://www.youtube.com/watch?v=eveKF7VnL48

Chapter 6

Integral Sliding Mode Flight Controller Design for a Quadrotor and its Application to a Heterogeneous Multi-Agent System

6.1 Introduction

In Chapter 5, we investigate the LQR-based rendezvous control for a group of heterogeneous MAS consisting of quadrotors and 2WMRs. Regarding the quadrotor waypoint tracking control, the proportional-integral-derivative (PID) flight control method [143] and the LQR-based flight control method [1, 144] are the most commonly used techniques in industry. However, these traditional control methods need the complete knowledge of the quadrotor dynamics, indicating that exact model parameters are known in the controller design. The errors in the parameters such as the mass, the center of mass and inertia can sometimes deteriorate the performance of the controller, and external disturbances also inevitably affect the performance of the flight controller. One of the approaches for dealing with model uncertainties and disturbances is the adaptive control [107]. In [145], disturbances are quantized by using the designed quantum logic, and then the adaptive controller is employed to stabilize the quadrotor. Some other methods for correcting parameter errors or estimating the quadrotor states can be found in [146–150]. Experimental studies are carried out on the flight controller design for quadrotors with extra payloads, e.g., [151, 152].

Sliding mode control (SMC) is a well developed technique to deal with uncertainties and disturbances such that the system can reach the desired states in finite time [153]. One of the major advantages of using SMC method is that matched uncertainties (the disturbances in the control input channel) can be effectively suppressed. Later, the integral sliding mode control (ISMC) technique has been deeply investigated [154]. By using the ISMC approach, the reaching phase is eliminated and the system trajectory starts in the designed sliding surface. The ISMC technique is used for the altitude control for a small helicopter with ground effect compensation in [155]. However, the work in [155] only considers the altitude information, which is not suitable for the flight controller design of a quadrotor.

Besides controlling a single quadrotor, many researchers study the control of a group of robots [21, 156, 157]. For the control of the MAS involving quadrotors, designing a robust flight controller that can reject model uncertainties and external disturbances plays an important role in guaranteeing the cooperative task to be successful. Up to date, cooperative control of the heterogeneous MAS considering model uncertainties and disturbances, especially the application-oriented research, needs to be further addressed.

Motivated by the above discussion, our work aims to design and implement an ISMC-based flight controller for a 6-DOF quadrotor. The designed controller is then applied to the consensus control of a heterogeneous MAS consisting of 2WMRs and quadrotors.

The main contribution of this chapter is three-fold:

- The ISMC strategy for the quadrotor flight control is proposed. We present the inner loop ISMC-based controller incorporating the reference angle signal and the desired position information. Accordingly, the stability analysis by using the Lyapunov approach is provided.

- We implement the LQR-based [1] and the proposed ISMC-based controllers on a quadrotor. The performances of the controllers are illustrated and compared in terms of mean square error (MSE). To emphasize the effectiveness of the designed ISMC-based flight controller, we further enlarge the model uncertainties and external disturbances by attaching an unknown weight to the quadrotor, and then the experimental comparisons between two controllers are conducted.

- The consensus control method for a group of MAS involving 2WMRs and quadrotors is proposed. The sufficient condition for the MAS to reach consensus is that switching graphs always have a spanning tree. The designed consensus control approaches are successfully realized in the experimental platform.

The rest parts of the chapter are organized as follows: Section 6.2 presents the dynamic model of the quadrotor and the actuator. Section 5.3.2 introduces the design of the quadrotor flight controller and analyzes the system stability. Section 6.3 shows cooperative control method of the heterogeneous MAS and gives the consensus condition. Experimental testings are presented in Section 6.4. Section 6.5 concludes the chapter.

6.2 Integral Sliding Mode Flight Controller Design for a Quadrotor

The dynamic model of a quadrotor and the inner-outer loop structured LQR-based flight controller are introduced in Section 5.3.2. However, model uncertainties and the external disturbances exist ubiquitously in practical situations and may destroy the stability of the system. Next, we present the design of the ISMC-based controller for the waypiont tracking in the presence of both model uncertainties and external disturbances. Following the similar stream described in [1], the PID and PD controllers are used for the altitude and ψ controls respectively. Thus they are omitted here. Instead of using the inner-outer loop structured LQR-based controller, desired positions x_d and y_d are also incorporated in the ISMC-based controller in the inner loop, see Figure 6.1. Since the dynamics of the quadrotor along x and y-axes are decoupled and similar, here we only present the ISMC-based controller along x-axis.

From (5.14a) and considering model uncertainties and external disturbances, we have

$$\dot{X}(t) = A_X X(t) + B_X u_\theta(t) + M_X \xi_X(t) + f_X(t), \tag{6.1}$$

where $X(t) = \left[x, \dot{x}, \theta, \dot{\theta}, v_\theta\right]^{\mathrm{T}}$, $A_X = \begin{bmatrix} 0\,1\,0\,0 & 0 \\ 0\,0\,g\,0 & 0 \\ 0\,0\,0\,1 & 0 \\ 0\,0\,0\,0 & \frac{2K_m l}{I_{XX}} \\ 0\,0\,0\,0 & -\omega \end{bmatrix}$, $B_X = [0, 0, 0, 0, \omega]^{\mathrm{T}}$. It is verified that the pair (A_X, B_X) is controllable. The matrix $M_X \in \mathbb{R}^{5\times l_r}$ is assumed to be

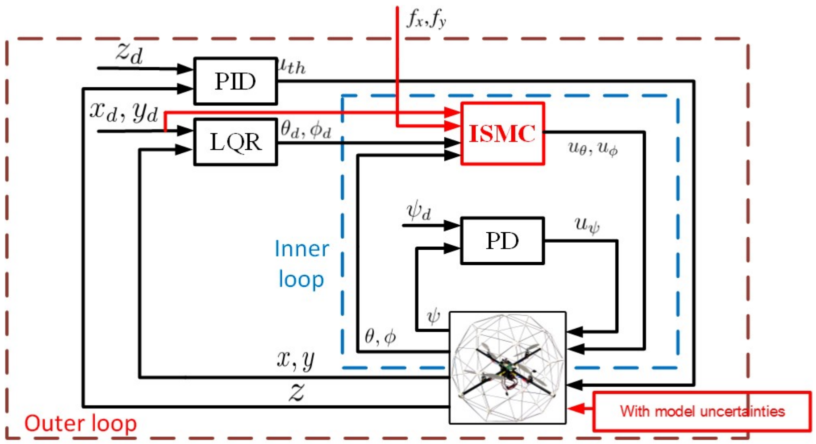


Figure 6.1: ISMC-based inner-outer loop control scheme.

known and can be written as $M_X = B_X D_X$, for some $D_X \in \mathbb{R}^{1\times l_r}$. $\xi_X(t)$ is the disturbance or model uncertainty which is unknown but with a known upper bound. $f_X(t)$ is the unmatched uncertainty which is also with a known upper bound. The ISMC-based controller is designed in light of the work in [154].

To deal with disturbances and uncertainties in (6.1), a sliding manifold for the quadrotor is defined as following:

$$\sigma_x(t) = GX_e(t) - GX_e(0) - G\int_0^t (A_X X_e(\tau) + B_X u_o(\tau))d(\tau), \tag{6.2}$$

where $X_e(t) = X(t) - \bar{X}(t)$, $\bar{X}(t) = [x_d, 0, \theta_d(t), 0, 0]^{\mathrm{T}}$ is the desired state for the system in (6.1), $\theta_d(t)$ is the reference signal for the pitch angle generated in the outer loop, and $G = (B_X^{\mathrm{T}} B_X)^{-1} B_X^{\mathrm{T}}$. The term $-GX_e(0)$ is designed to eliminate the reaching phase by ensuring that $\sigma_x(t) = 0$. The controller $u_\theta(t)$ for (6.1) consists of two parts:

$$u_\theta(t) = u_o(t) + u_n(t), \tag{6.3}$$

in which, $u_o(t) = -FX(t)$ is the nominal controller obtained by using the LQR method for the nominal system (A_X, B_X) to achieve the desired performance. General steps to design the LQR-based controller for the linear state space model can be found in [136]. We choose $u_n(t)$ as

$$u_n(t) = -\rho(t)(GB_X)^{-1}\frac{\sigma_x(t)}{\|\sigma_x(t)\|}$$

with $\rho(t)$ as the modulation gain to keep the system trajectories sliding along the sliding surface in (6.2). Using (6.1) and (6.3), we have

$$\begin{aligned}\dot{\sigma_x}(t) &= G(A_X X(t) + B_X u_\theta(t) + B_X D_X \xi_X(t) + f_X(t)) \\ &\quad -GA_X X(t) + GB_X F X(t) \\ &= GA_X X(t) - GB_X F X(t) \\ &\quad -GB_X \rho(t)(GB_X)^{-1}\frac{\sigma_x(t)}{\|\sigma_x(t)\|} + GB_X D_X \xi_X(t) \\ &\quad +Gf_X(t) - GA_X X(t) + GB_X F X(t) \\ &= -\rho(t)\frac{\sigma_x(t)}{\|\sigma_x(t)\|} + GB_X D_X \xi_X(t) + Gf_X(t).\end{aligned}$$

Construct the Lyapunov function

$$V(t) = \frac{1}{2}\sigma_x^{\mathrm{T}}(t)\sigma_x(t).$$

It is shown that

$$\begin{aligned}\dot{V}(t) &= \sigma_x^{\mathrm{T}}(t)\dot{\sigma_x}(t) \\ &= -\rho(t)\|\sigma_x(t)\| + \sigma_x^{\mathrm{T}}(t)D_X\xi_X(t) + \sigma_x^{\mathrm{T}}(t)Gf_X(t) \\ &\leq \|\sigma_x(t)\|(-\rho(t) + \|D_X\xi_X(t)\| + \|Gf_X(t)\|)\end{aligned}$$

Note that if we choose $\rho(t) \geq \|D_X\xi_X(t)\| + \|Gf_X(t)\| + \eta_X$, where η_X is a positive scalar, it follows that

$$\dot{V}(t) \leq -\eta_X\|\sigma_x(t)\| = -\eta_X\sqrt{2V(t)}. \tag{6.4}$$

Integrating (6.4) on both sides, we get

$$\sqrt{2V(t)} - \sqrt{2V(0)} \leq -\eta_X t.$$

The Lyapunov function will reach $V(t) = 0$ within the finite time $\frac{\sqrt{2V(0)}}{\eta_X}$.

6.3 Cooperative Control of the Heterogeneous Multi-agent System

6.3.1 Consensus Problem Formulation

Consider a group of heterogeneous MAS with N agents

$$\begin{aligned} X_i(t_{k+1}) &= f_i(X_i(t_k), A(t_k), u_x i(t)), \\ Y_i(t_{k+1}) &= g_i(Y_i(t_k), A(t_k), u_y i(t)),\ t \in [t_k, t_{k+1}), \end{aligned} \tag{6.5}$$

where $X_i(t_k), Y_i(t_k) \in \mathbb{R}^{m_i}$ are the states of agent i along x and y-axes at the time instant t_k. $\tilde{x}_i(t_k)$ and $\tilde{y}_i(t_k)$ are the first elements in $X_i(t_k)$ and $Y_i(t_k)$ respectively, which denote x and y position of agent i. $A(t_k)$ is the adjacency matrix. Agents only broadcast the information to the network at certain time instants $\{t_k, k = 0, 1, \ldots\}$. After receiving the information from the neighbors, each agent updates its desired position for $t \in [t_k, t_{k+1})$ as

$$\bar{x}_i(t_k) = \frac{\tilde{x}_i(t_k) + \sum\limits_{j \in \mathcal{N}_i(t_k)} \tilde{x}_j(t_k)}{d_i(t_k) + 1},\ \bar{y}_i(t_k) = \frac{\tilde{y}_i(t_k) + \sum\limits_{j \in \mathcal{N}_i(t_k)} \tilde{y}_j(t_k)}{d_i(t_k) + 1}, \tag{6.6}$$

where $\mathcal{N}_i(t_k)$ is the neighbor set and $d_i(t_k)$ is the number of neighbors of agent i at t_k. Consensus is reached if $\lim_{k\to\infty} \|\tilde{x}_i(t_k) - \tilde{x}_j(t_k)\| = 0, \lim_{k\to\infty} \|\tilde{y}_i(t_k) - \tilde{y}_j(t_k)\| = 0, \forall i, j = 1, 2, \ldots, N, i \neq j$.

6.3.2 Consensus Algorithms for 2WMRs and Quadrotors

2WMRs are operated in the two-dimensional ground plane, and the coordinate system of a 2WMR is presented in Section 3.3.1. The consensus algorithm for 2WMRs are given as following: At the information interaction time instant t_k, each 2WMR calculates the desired position $(\bar{x}_i(t_k),\ \bar{y}_i(t_k)), t_k \leqslant t < t_{k+1}$ according to (6.6). By setting the wheel velocities with the same magnitude but opposite directions, the 2WMR rotates in place until either the nose or tail points at $(\bar{x}_i(t_k),\ \bar{y}_i(t_k))$. Both wheel velocities are then set as a constant value $\bar{v}$ so that the 2WMR moves to $(\bar{x}_i(t_k),\ \bar{y}_i(t_k))$. After reaching the desired position, the 2WMR stops and waits for the next information interaction time instant t_{k+1}. Suppose the rotating time for agent i is $Tr_i(t_k)$ during $[t_k, t_{k+1})$, here we choose t_{k+1} large enough; i.e., $t_{k+1} - t_k >$

$Tr_i(t_k) + \frac{\sqrt{(\bar{x}_i(t_k)-\tilde{x}_i(t_k))^2+(\bar{y}_i(t_k)-\tilde{y}_i(t_k))^2}}{\bar{v}}$ such that the 2WMR can reach the desired position before t_{k+1}.

Similarly, for quadrotors, at the information interaction time instant t_k, each agent uses (6.6) to calculate its desired position $(\bar{x}_i(t_k),\ \bar{y}_i(t_k))$ for $t \in [t_k, t_{k+1})$. As we mentioned in Section 5.3.2, the dynamics of the quadrotor along x and y-axes are decoupled, we can apply the designed waypoint tracking flight controller to solve the consensus problem for this heterogeneous MAS. The quadrotor will move to and hover at $(\bar{x}_i(t_k),\ \bar{y}_i(t_k))$ until the next interaction time instant t_{k+1}.

Next we show two useful lemmas for analyzing the consensus of the heterogeneous MAS.

Lemma 6.1 (Lemma 3.3, 3.7 & Corollary 3.5 in [29]). *If all diagonal entries of a stochastic matrix M are positive and the associated directed graph of M has a spanning tree, then M is SIA.*

Lemma 6.2 ([124]). *If $\{S_1, S_2, \ldots, S_k\}$ is a set consisting of finite SIA matrices with the same dimension $n \times n$, any sequence of matrix product $S_{i_m}S_{i_{m-1}}\ldots S_{i_1}$ is SIA. For the product $S_{i_m}S_{i_{m-1}}\ldots$ with infinite length, there exists a column vector c such that*

$$\lim_{m\to\infty} S_{i_m}S_{i_{m-1}}\ldots S_{i_1} = \mathbf{1}_n c^{\mathrm{T}}.$$

Theorem 6.1. *Consider a group of heterogeneous MAS consisting of 2WMRs and quadrotors, and the desired position for agent i is given as $(\bar{x}_i(t_k), \bar{y}_i(t_k))$, for $t \in [t_k, t_{k+1})$ in (6.6). The quadrotor updates its position using the ISMC-based controller designed in Section 5.3.2, and the 2WMR updates the position using the methods described in this section. If the switching directed graphs $\mathcal{G}(t_k), \forall t_k, k = 1, 2, \ldots$ always have spanning trees, the consensus problem of this heterogeneous MAS is solved.*

Proof. By applying the designed controllers to the quadrotors and to 2WMRs for updating the desired positions during $t \in [t_k, t_{k+1})$, we get that agent i can reach $(\bar{x}_i(t_k),\ \bar{y}_i(t_k))$ before the next information interaction time instant t_{k+1}, $i = 1, 2, \ldots N$. Denote $\hat{X}(t_k) = [\tilde{x}_1(t_k), \tilde{x}_2(t_k), \ldots, \tilde{x}_N(t_k)]^{\mathrm{T}}$, we have

$$\hat{X}(t_{k+1}) = \frac{A(t_k) + I_N}{d_i(t_k) + 1}\hat{X}(t_k) = D(t_k)\hat{X}(t_k).$$

It is readily shown that each row sum of $D(t_k)$ equals to 1 and the diagonal entries of $D(t_k)$ are all positive. Using **Lemma 6.1**, if the switching graph $\mathcal{G}(t_k)$ always has a directed spanning tree, $D(t_k)$ is SIA, for $k = 0, 1, 2, \ldots$. Then according to **Lemma 6.2**, there exists a column vector c such that

$$\lim_{k\to\infty} \hat{X}(t_k) = D(t_{k-1})D(t_{k-2})\ldots D(t_1)D(t_0)\hat{X}(t_0) = \mathbf{1}_N c^{\mathrm{T}} \hat{X}(t_0),$$

which implies that consensus can be reached along x-axis. The consensus analysis along y-axis can be obtained in the same line. □

6.4 Experiment

6.4.1 ISMC-based Flight Control Test for a Single Quadrotor

Figure 6.2 demonstrates the experimental setup of the multi-agent system. A cluster of reflective markers are mounted with the unique configuration on the quadrotor. The markers on each agent are then regarded and tracked as a rigid body by OptiTrack™ cameras on the wall. The x, y, z velocities are estimated by using the derivative approach. The inertial measurement unit (IMU) with the 3-axis gyroscope, accelerometer and magnetometer are used to measure the quadrotor's angular rates and accelerations. The controllers for quadrotors and 2WMRs are compiled on the Host PC and then downloaded to embedded microcontrollers on the vehicles. The quadrotor parameters are given as follows [1]: $K_m = 120\mathrm{N}$, $K_n = 4\mathrm{N}\cdot\mathrm{m}$, $\omega = 15\mathrm{rad/sec}$, $I_{XX} = I_{YY} = 0.03\mathrm{kg{\cdot}m^2}$, $I_{ZZ} = 0.04\mathrm{kg{\cdot}m^2}$ $m = 1.4\mathrm{kg}$ and $l = 0.2\mathrm{m}$.

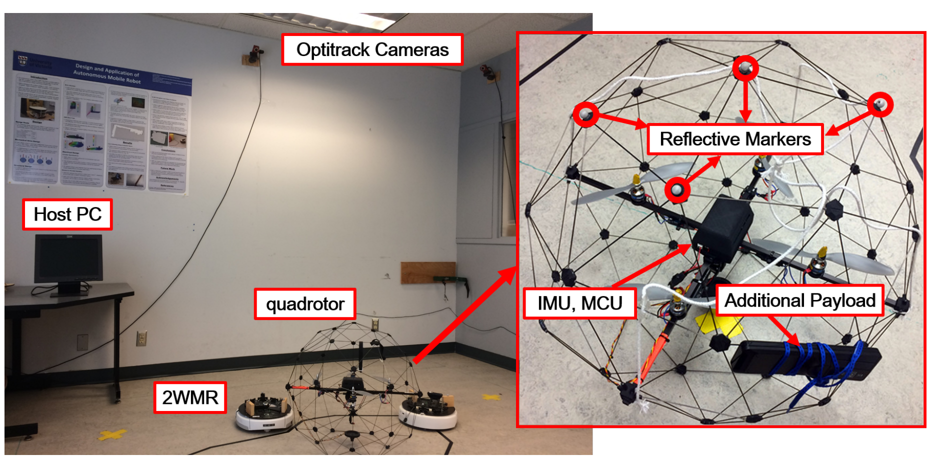


Figure 6.2: Experimental setup of the multi-agent system.

The designed waypoint tracking task is as following: The quadrotor first hovers at $[x_o, y_o, z_o]^{\mathrm{T}} = [-0.78, -0.65, 0.3]^{\mathrm{T}}$, and then it will move to and hover at the desired position $[x_d, y_d, z_d]^{\mathrm{T}} = [0.3, 0.3, 0.3]^{\mathrm{T}}$. When applying the ISMC-based controller to the quadrotor, we modify (6.3) to be

$$u_\theta(t) = -FX(t) - \rho(t)(GB_X)^{-1}\frac{\sigma_x(t)}{\|\sigma_x(t)\| + \delta}, \tag{6.7}$$

where $\delta = 0.0001$ is a small scalar, which is chosen for eliminating the chattering. Compared with using only the LQR-based flight controller $u_o(t)$, the computational complexity of using $u_\theta(t)$ increases due to the part of $u_n(t)$. Note that

$$u_n(t) = -\rho(t)(GB_X)^{-1}\frac{\sigma_x(t)}{\|\sigma_x(t)\|},$$

and $\sigma_x(t) = GX_e(t) - GX_e(0) - G\int_0^t (A_X X_e(\tau) + B_X u_o(\tau))d(\tau)$, where $G, B_X, A_X, \rho(t)$, are all constant matrices or parameters. The integral calculation $\int_0^t (A_X X_e(\tau) + B_X u_o(\tau))d(\tau)$ occupies few computational resource. The flight controller works at a sampling frequency 200 Hz, and our experimental case study was carried out using MATLAB on a PC with Intel(R) Core(TM) i5-3470 CPU @ 3.20 GHz, 3.20 GHz. It implies that the PC is capable of handling the increased complexity caused by the ISMC-based flight controller.

The linearizion approximation of the quadrotor dynamics unavoidably results in model uncertainties in (5.14), such as neglecting the intrinsic feature of nonlinearities in the dynamics, omitting of the centrifugal force, centripetal force, drag force and so on [137]. We present two sets of tests to verify the effectiveness of the proposed control method. In the first set of test, we apply the LQR-based controller [1] and the proposed ISMC-based controller to the quadrotor to conduct the waypoint tracing task. In the second set of test, the additional weight (0.17kg, 12.1% to the weight of the quadrotor itself), unknown to the controller is attached to the protective cage of the quadrotor. Significant model uncertainties are added to the quadrotor dynamics and more disturbances are brought to the controller: The center of mass of the quadrotor is displaced, the mass and moments of inertia of the B-frame axes are changed. Extra torques are generated as the disturbances during the flight tasks. Both the LQR-based and ISMC-based controllers are tested and compared in this situation. We summarize the experimental case studies as follows: a) No additional weight, LQR; b) no additional weight, ISMC; c) additional weight, LQR; d) additional weight, ISMC.

For each case, the quadrotor will first successfully hover at $[x_o, y_o, z_o]^{\mathrm{T}}$. The data

Table 6.1: MSEs of implementing the LQR-based and ISMC-based controllers in experimental studies

case	a	b	c	d
x	0.1855	0.1455	0.3071	0.1666
y	0.1457	0.1123	0.2165	0.1125
θ	0.0026	0.0020	0.0090	0.0047
ϕ	0.0023	0.0018	0.0054	0.0040

will be recorded and analyzed from the time instant of sending the command "*Move to* x_d" to the quadrotor. MSEs are calculated as follows

$$MSE_x = \frac{1}{N}\sum_{k=1}^{N}\left(x(k) - x_d\right)^2, MSE_\phi = \frac{1}{N}\sum_{k=1}^{N}\left(\phi(k) - \phi_d(k)\right)^2,$$
$$MSE_y = \frac{1}{N}\sum_{k=1}^{N}\left(y(k) - y_d\right)^2, MSE_\theta = \frac{1}{N}\sum_{k=1}^{N}\left(\theta(k) - \theta_d(k)\right)^2,$$

where N represents the total number of samplings during the waypoint tracking task. Results of the obtained MSEs for four cases are shown in TABLE 6.1.

Figure 6.3 shows x and y position evolutions of the quadrotor for the cases a) and b). It is shown that smaller overshoots are obtained, and the MSEs of x, y, θ and ϕ are all decreased by using our designed ISMC-based controller.

The effectiveness of the ISMC-based controller becomes more apparent when uncertainties and external disturbances are increased. We have conducted 20 experiments for the case c), there is little chance of achieving successful implementation of the LQR-based controller. There is only one *almost successful* demonstration: The quadrotor moves from $[x_o, y_o, z_o]^{\mathrm{T}}$ to $[x_d, y_d, z_d]^{\mathrm{T}}$, hovers at $[x_d, y_d, z_d]^{\mathrm{T}}$ with large overshoots and oscillations for several seconds and then the system becomes unstable. Comparative experimental testing results for implementing the ISMC-based controller are also given in Figures 6.4–6.6. In Figures 6.5 and 6.6, the desired pitch and roll angles are obtained from the on-line calculation in the outer loop using (6a) and (6b) respectively. It can be seen that more accurate orientation control (inner loop) is achieved by using the ISMC-based controller, and accordingly more precise position control (out loop) is realized. Note that MSEs of x, y, θ and ϕ are dramatically decreased by using our designed ISMC-based controller when the additional

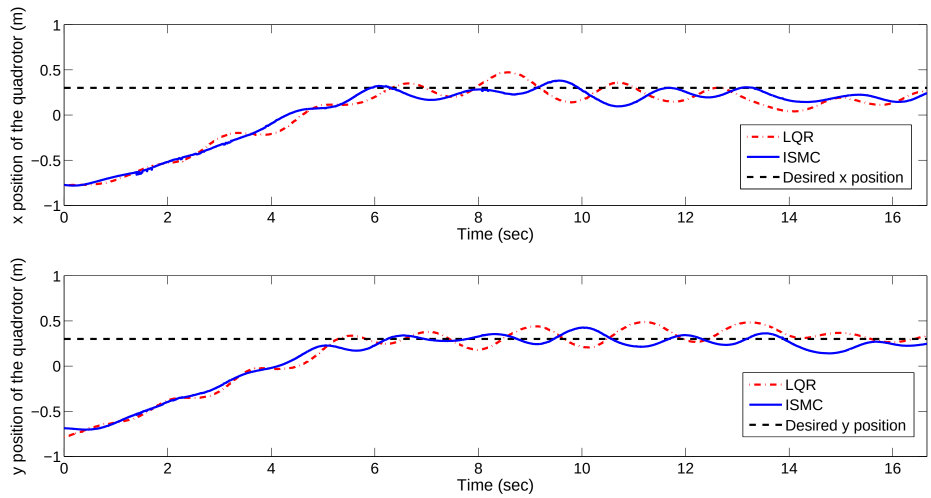


Figure 6.3: Comparison of the experimental results: Time responses of x and y positions by using the LQR-based and ISMC-based controllers for the cases a) and b).

weight is attached to the quadrotor. It is worthwhile to observe that MSEs of x and y for the case d) are even smaller than the results of the case a), which implies that the ISMC-based controller can result in better flight performance for the quadrotor with the additional weight than the LQR-based controller applied to the quadrotor without the additional weight. We further attach the payload with different mass to the different position on the quadrotor, and then repeatedly conduct the experimental cases a)-d). It is shown that our designed controller can significantly reject negative effects caused by bounded model uncertainties and external disturbances. The position and angle evolutions, MSEs analyses, experimental videos are all consistent with the results we have already shown in the work, and thus they are omitted here.

6.4.2 Consensus Control Test for the Heterogeneous MAS

The experiment is conducted with three 2WMRs and one quadrotor. The motions of 2WMRs are also tracked by OptiTrack™ cameras, and the information of the agents are interacted through the wireless network. Agents are initialized with random positions and zero velocities in the workspace. The quadrotor takes off vertically, reaches the safety height of 0.3m, and then consensus algorithms are applied to MAS. At each information interaction time instant t_k, the graph always has a spanning tree. Agent i moves to and stays at the calculated desired position $(\bar{x}_i(t_k),\ \bar{y}_i(t_k))$ in the time interval $[t_k, t_{k+1})$. The x and y position evolutions and trajectories are shown

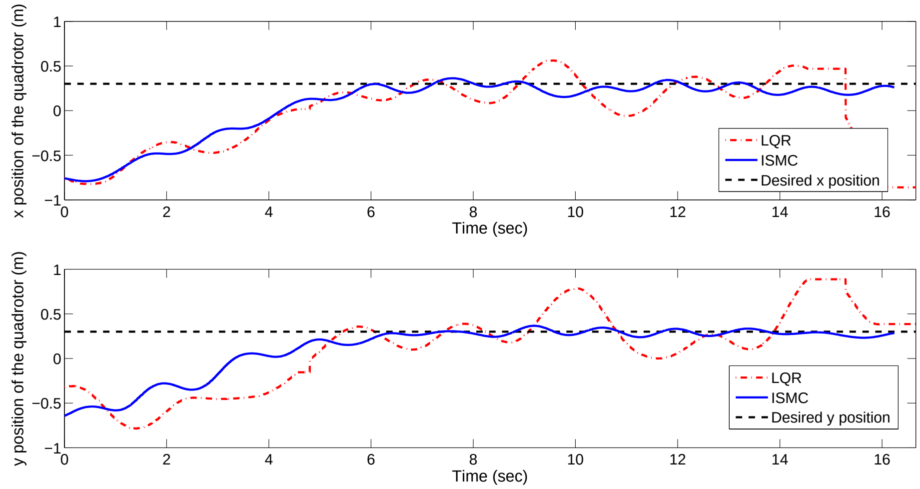


Figure 6.4: Comparison of the experimental results: Time response of x and y positions by using the LQR-based and ISMC-based controllers for the cases c) and d).

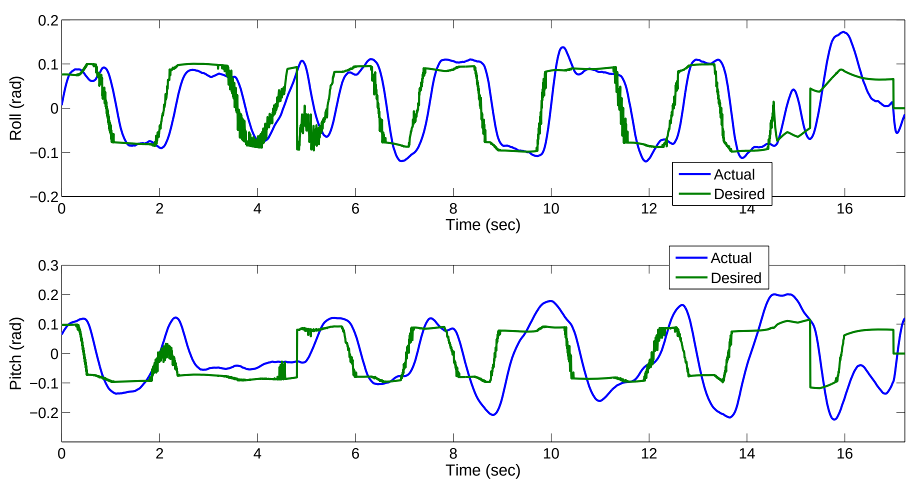


Figure 6.5: Experimental results of the LQR-based controller: Time response of ϕ and θ for the case c).

in Figures 6.7 and 6.8 respectively. It is shown that consensus of the heterogeneous MAS is reached. Note that the 2WMR is with the diameter 34cm, which implies that the distance between two geometry centers of two 2WMRs are at least 34cm. The collision happens when 2WMRs are trying to converge to the same spot, and the quadrotor hovers above 2WMRs when consensus is reached.

The desired and actual ϕ and θ of the quadrotor during the consensus task are shown in Figure 6.9. The reasonable PWM duty cycle of the motor on the quadrotor ranges from 5% to 10% [1]. Figure 6.10 shows evolutions of PWM duty cycles of the motors, from which we observe that saturations exist in PWM signals of Motors

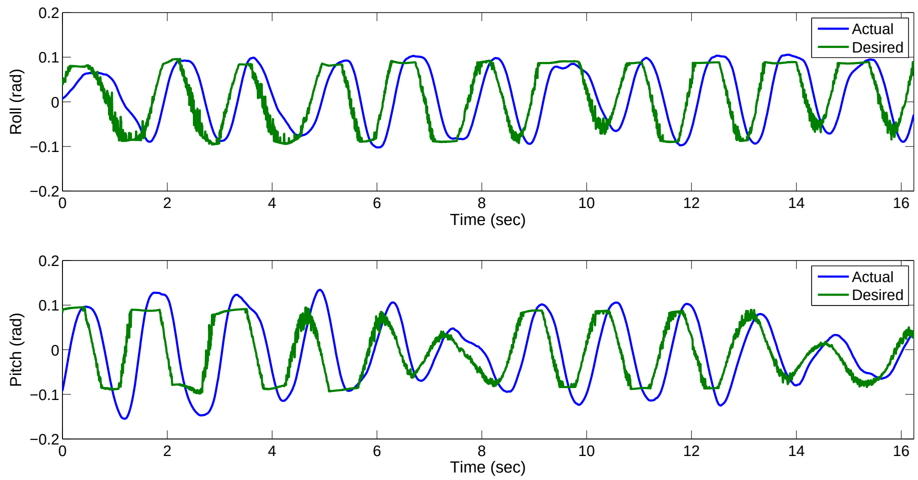


Figure 6.6: Experimental results of the ISMC-based controller: Time response of ϕ and θ for the case d).

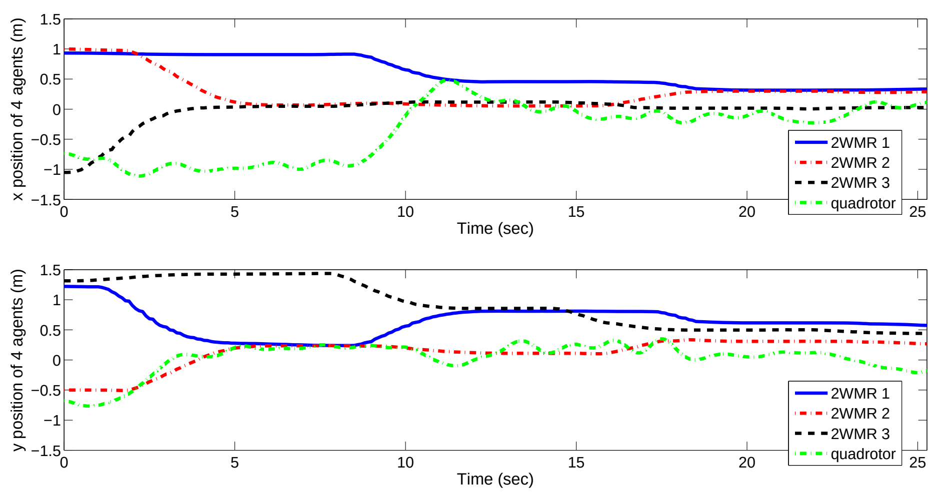


Figure 6.7: Experimental results of the consensus control algorithms: Time response of x and y positions of four agents.

1 and 4. This is because that in Figure 6.2, the unknown payload is attached to the protective cage, with the position close to Motors 1 and 4 (Motor 1 is on the crossbeam with red tape). In order to keep the quadrotor balanced during the flight, Motors 1 and 4 will intuitively give more efforts.

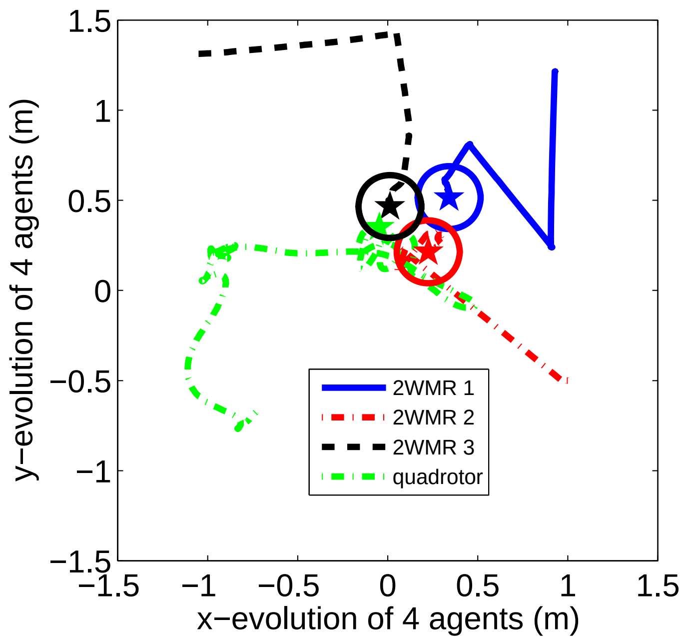


Figure 6.8: Experimental results of the consensus control algorithms: Trajectories of the agents on xy plane.

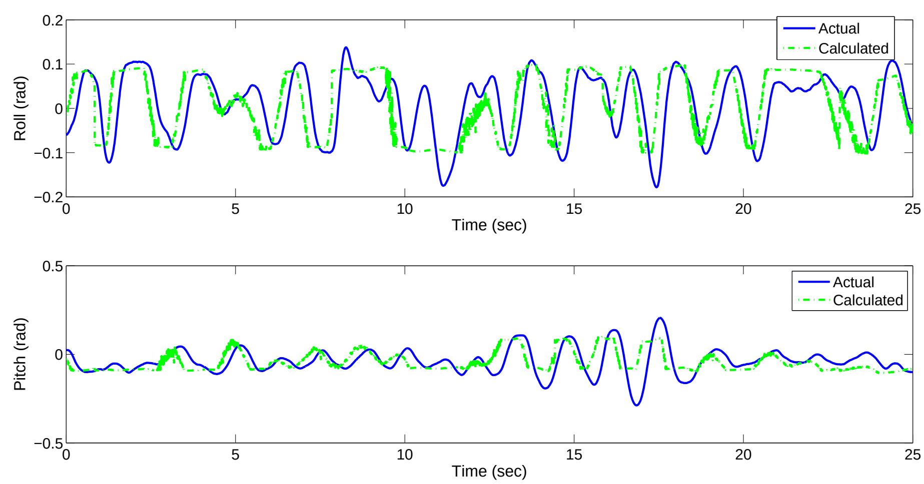


Figure 6.9: Experimental results of the consensus control algorithms: Time response of ϕ and θ of the quadrotor.

6.5 Conclusion

In this chapter, an ISMC-based flight controller for the quadrotor waypoint tracking task has been discussed. We first presented modelings of the quadrotor and actuators, and then introduced the inner-outer loop structured LQR-based control strategy. In order to decrease the negative effects caused by model uncertainties and the external

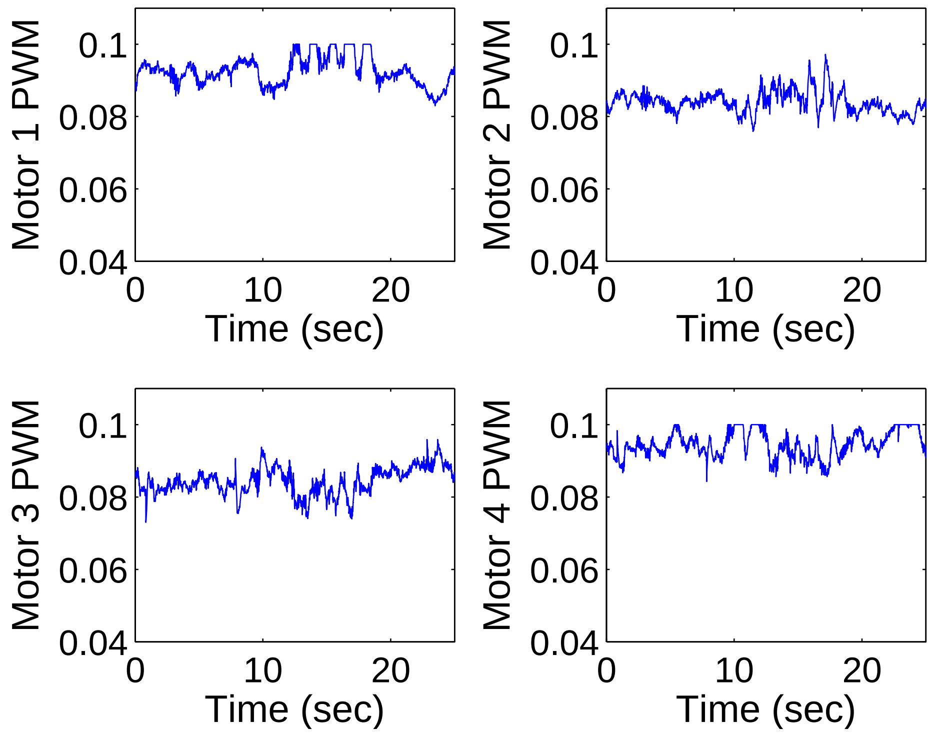


Figure 6.10: Experimental results of the consensus control algorithms: Time response of the PWM duty cycles of the motors.

disturbances, we proposed the ISMC-based flight controller incorporating the reference angular signal and the desired position information into the inner loop of the controller. By using the ISMC technique, we eliminated both the reaching phase to the sliding surface and the chattering in control signals. The detailed stability analysis was provided. The designed controller was then applied to solve the consensus problem for a heterogeneous multi-agent system. From the experimental testings, it is shown that the effects of the bounded model uncertainties and external disturbances are significantly rejected in conducting the waypoint tracking task, and consensus algorithms for 2WMRs and quadrotors work effectively. Future research will be focused on the quadrotor hovering control at the precise position and the flight controller design in the discrete-time domain. Interesting extensions of this work can also be pursued in considering actuator faults [158], time-delays [159] and designing the novel flight controller using the model predictive control (MPC) technique [141, 142].
A video of the experiments is posted on the following URL:
https://youtu.be/u1qqB166O-8

Chapter 7

Conclusions and Future Work

This thesis mainly addresses two concerns in consensus problem studies from both theoretical and application points of view: Irregular sampling cooperative control and consensus of heterogeneous MASs. We design appropriate control methods for each specific dynamics in MASs, and rigorously analyze consensus convergences. We also successfully implement the designed control methods on practical systems.

7.1 Conclusions

Chapter 3 investigates the consensus problem for a group of 2WMRs using non-uniform sampling. The directed and switching communication topologies are considered. The control protocols for first-order/second-order system dynamics are designed with bounded control gains. The *Rotate&Run Scheme* is proposed to update vehicles' states: 1) The vehicle calculates its goal orientation and the input of each wheel at the sampling time instants by using states of itself and its neighbors; 2) the vehicle rotates in place until it aims at the calculated direction; 3) the vehicle moves forward/backward with calculated wheel velocities until the next sampling time instant. It is shown that consensus in a group of 2WMRs can be achieved when switching directed graphs satisfy certain conditions. The convergence analysis of consensus is conducted based on algebraic graph theory and stochastic matrix analysis. Experiments demonstrate the effectiveness of the proposed methods.

In Chapter 4, we propose an event-triggered rendezvous control method for multiple 2WMRs subject to time-varying communication delays. By checking integral-type event-triggering conditions asynchronously and periodically, each 2WMR determines whether or not to sample and broadcast its states. When the information used in an agent's controller is updated, the 2WMR calculates its control input, and then a *Rotate&Compensate&Run Rendezvous Scheme* is provided for the 2WMR to update its state. We present a sufficient condition for 2WMRs to asymptotically reach rendezvous, and the convergence analysis is conducted by Lyapunov methods. Experiments are further presented to validate the effectiveness of the proposed control method.

Controlling heterogeneous MASs to cooperatively accomplish tasks is an emerging topic in the application-oriented research of robotics. Chapter 5 investigates the consensus problem of an MAS consisting of quadrotors and 2WMRs. Directed and switching interaction topologies over the network are considered. We propose a distributed LQR consensus protocol for the quadrotors and design an LQR-based *Rotate&Run Consensus Scheme* for the 2WMRs to update the states. We use the algebraic graph theory and stochastic matrix analysis to conduct the convergence analysis of consensus. The underactuation characteristic of the 2WMR dynamics is considered in the controller design. The effectiveness of the control methods is verified by experiments.

Chapter 6 investigates a novel ISMC-based strategy for the waypoint tracking control of a quadrotor in the presence of model uncertainties and external disturbances. The proposed controller has the inner-outer loop structure: The outer loop is to generate reference signals for roll and pitch angles, while the inner loop is designed by using the ISMC technique for the quadrotor to track the desired x, y positions, roll and pitch angles. The Lyapunov stability analysis is provided to show that the negative effects of bounded model uncertainties and external disturbances can be significantly decreased. The designed controller is then applied to a heterogeneous MAS consisting of quadrotors and 2WMRs to solve the consensus problem. We present control algorithms for the 2WMRs and quadrotors. Consensus of the heterogeneous MAS can be reached if switching graphs always have a spanning tree. Finally, experimental tests are conducted to verify the effectiveness of proposed control methods.

7.2 Future Work

The works presented in Chapters 3 and 4 have focused on the non-uniform sampling cooperative control and the event-triggered asynchronous sampling rendezvous control for a group of 2WMRs. It is noted that only sufficient conditions for consensus convergence are provided. Convergence rates of MASs are not discussed, which motivates us to conduct a more in-depth study to investigate how the chosen controller parameters affect rendezvous rates. Another interesting concern for the future work of Chapter 4 can be the event-triggered rendezvous control for a group of second-order, high-order and heterogeneous agents.

In Chapter 6, we study an ISMC-based quadrotor flight controller, which is robust against model uncertainties and external disturbances. However, it is observed that in the experiment, the quadrotor hovers at the desired position with relatively large oscillations. Future research will be focused on the quadrotor hovering control at the precise position. An observer can be designed for the accurate disturbance estimation [147, 160], and this technique can be incorporated with our ISMC-based controller to further improve flight performances. Interesting extensions of this work can also be pursued in considering actuator faults [158], time-delays [159], and designing the novel flight controller using model predictive control (MPC) techniques [141, 142].

To consider more practical applications of quadrotors, we plan to build a novel prototype of a quadrotor with a manipulator, and further develop the control method for the newly developed dynamics in an MAS. Specifically: 1) Based on the open-architecture hardware and the open-source software of the quadrotor, we propose to attach a manipulator and a camera system to the quadrotor mechanically and electrically, and accordingly derive the modeling of the system dynamics. 2) Relying on the vision-based control, we aim to design the autonomous control scheme for the individual agent to recognize, pick up and safely deliver the designated object. 3) By incorporating the designed individual agent controller into MAS control techniques, we plan to investigate the coordination of this practical vehicle-manipulator system from theoretical and experimental standpoints. Practical constraints such as time delays, data losses, and multirate data transmission will also be considered.

Appendix A

Publications

- **Refereed journal papers that have been accepted**

J1. **B. Mu**, K. Zhang and Y. Shi, "Integral sliding mode flight controller design for a quadrotor and the application in a heterogeneous multi-agent system," *IEEE Transactions on Industrial Electronics*, accepted for publication, 2017. [Online]. Available: http://dx.doi.org/10.1109/TIE.2017.2711575
(This work is presented in Chapter 6.)

J2. **B. Mu**, J. Chen, Y. Shi and Y. Chang, "Design and implementation of non-uniform sampling cooperative control on a group of two-wheeled mobile robots," *IEEE Transactions on Industrial Electronics*, vol. 64, no.6, pp. 5035-5044, Jun. 2017.
(This work is presented in Chapter 3.)

J3. **B. Mu**, H. Li, J. Ding and Y. Shi, "Consensus in second-order multiple flying vehicles with random delays governed by a Markov chain," *Journal of The Franklin Institute*, vol. 352, no. 9, pp. 3628-3644, Feb. 2015.

J4. **B. Mu** and Y. Shi, "Distributed LQR consensus control for heterogeneous multi-agent systems: Theory and experiments," submitted to *IEEE/ASME Transactions on Mechatronics*, accepted with minor revision, 2017.
(This work is presented in Chapter 5.)

J5. A. Wang, **B. Mu** and Y. Shi, "Consensus control for a multi-agent system with integral-type event-triggering condition and asynchronous periodic detection," *IEEE Transactions on Industrial Electronics*, vol. 64, no.7, pp. 5629-5639, Jun. 2017.

J6. H. Zhang, Y. Shi and **B. Mu**, "Optimal $\mathcal{H}_\infty$ based linear-quadratic regulator tracking control for discrete-time Takagi-Sugeno fuzzy systems with preview actions," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 135, no. 4, pp. 044501-1044501-5, May. 2013.

J7. A. Wang, **B. Mu** and Y. Shi, "Event-triggered consensus control for multi-agent systems with time-varying communication and event-detecting delays," submitted to *IEEE Transactions on Control Systems Technology*, accepted with minor revision, 2017.

- **Refereed journal papers that are under review**

J8. **B. Mu** F. Xiao and Y. Shi, "Event-based rendezvous control for a group of robots with asynchronous periodic detection and communication time delays," submitted.
(This work is presented in Chapter 4.)

- **Refereed conference papers that have appeared or been accepted**

C1. **B. Mu**, Y. Pei and Y. Shi, "Integral sliding mode control for a quadrotor in the presence of model uncertainties and external disturbances," in *Proceedings of American Control Conference*, Seattle, USA, May 24-26, 2017, pp. 5818-5823.

C2. **B. Mu**, H. Li, W. Li, and Y. Shi, "Consensus for multiple Euler-Lagrange dynamics with arbitrary sampling periods and event-Triggered strategy," in *Proceedings of 11th World Congress on Intelligent Control and Automation*, Shenyang, China, June 29-July 4, 2014, pp. 2596-2601.

C3. **B. Mu** and Y. Shi, "Cooperative control of multiple flying vehicles with uncertain Markov delays under switching topologies," in *Proceedings of 24th Canadian Congress of Applied Mechanics*, Saskatoon, Saskatchewan, Canada, June 2-6, 2013.

Bibliography

[1] *Quanser Qball-X4 User Manual*, 2nd ed., Quanser Consulting Inc., Markham, Ontario, Canada, 2012.

[2] *Quanser Qbot User Manual*, 7th ed., Quanser Consulting Inc., Markham, Ontario, Canada, 2012.

[3] *Multi-vehicle Coordination*, Quanser Consulting Inc., Markham, Ontario, Canada, 2012.

[4] Y. Jia and L. Wang, "Leader-follower flocking of multiple robotic fish," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 3, pp. 1372–1383, Jun. 2015.

[5] S. Li, R. Kong, and Y. Guo, "Cooperative distributed source seeking by multiple robots: Algorithms and experiments," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 6, pp. 1810–1820, Dec. 2014.

[6] B. Mu, H. Li, J. Ding, and Y. Shi, "Consensus in second-order multiple flying vehicles with random delays governed by a Markov chain," *J. Franklin Inst.*, vol. 352, no. 9, pp. 3628–3644, Feb. 2015.

[7] X. Zhang, A. J. Flueck, and C. P. Nguyen, "Agent-based distributed volt/var control with distributed power flow solver in smart grid," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 600–607, Mar. 2016.

[8] Q. Li, F. Chen, M. Chen, J. M. Guerrero, and D. Abbott, "Agent-based decentralized control method for islanded microgrids," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 637–649, Mar. 2016.

[9] M. J. Ghorbani, M. A. Choudhry, and A. Feliachi, "A multiagent design for power distribution systems automation," *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 329–339, Jan. 2016.

[10] S. Zhu, C. Chen, W. Li, B. Yang, and X. Guan, "Distributed optimal consensus filter for target tracking in heterogeneous sensor networks," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1963–1976, Dec. 2013.

[11] G. Scutari and S. Barbarossa, "Distributed consensus over wireless sensor networks affected by multipath fading," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 4100–4106, Aug. 2008.

[12] C. Liu, K. Sun, Z. H. Rather, Z. Chen, C. L. Bak, P. Thgersen, and P. Lund, "A systematic approach for dynamic security assessment and the corresponding preventive control scheme based on decision trees," *IEEE Trans. Power Syst.*, vol. 29, no. 2, pp. 717–730, Mar. 2014.

[13] Y. Hao, J. Tang, and Y. Cheng, "Secure cooperative data downloading in vehicular ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 523–537, Sept. 2013.

[14] K. D. Do, "Bounded assignment formation control of second-order dynamic agents," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 2, pp. 477–489, Apr. 2014.

[15] Z. Lin, L. Wang, Z. Han, and M. Fu, "Distributed formation control of multi-agent systems using complex Laplacian," *IEEE Trans. Autom. Control*, vol. 59, no. 7, pp. 1765–1777, Jul. 2014.

[16] L. Brinon-Arranz, A. Seuret, and C. Canudas, "Cooperative control design for time-varying formations of multi-agent systems," *IEEE Trans. Autom. Control*, vol. 59, no. 8, pp. 2283–2288, Aug. 2014.

[17] D. Gu and Z. Wang, "Leader-follower flocking: Algorithms and experiments," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 5, pp. 1211–1219, Sept. 2009.

[18] J. Zhan and X. Li, "Flocking of multi-agent systems via model predictive control based on position-only measurements," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 377–385, Feb. 2013.

[19] Z. Li, H. Liu, B. Zhu, and H. Gao, "Robust second-order consensus tracking of multiple 3-DOF laboratory helicopters via output feedback," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 5, pp. 2538–2549, Oct. 2015.

[20] Y. C. Choi and H. S. Ahn, "Nonlinear control of quadrotor for point tracking: Actual implementation and experimental tests," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 3, pp. 1179–1192, Jun. 2015.

[21] Y. Tang, X. Xing, H. Karimi, L. Kocarev, and J. Kurths, "Tracking control of networked multi-agent systems under new characterizations of impulses and its applications in robotic systems," *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1299–1307, Feb. 2016.

[22] J. Mei, W. Ren, B. Li, and G. Ma, "Distributed containment control for multiple unknown second-order nonlinear systems with application to networked lagrangian systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 9, pp. 1885–1899, Sept. 2015.

[23] C. Godsil and G. Royle, *Algebraic Graph Theory.* New York: Springer-Verlag, 2001.

[24] N. A. Lynch, *Distributed Algorithms.* Morgan Kaufmann Publishers, Inc., 1996.

[25] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods.* Englewood Cliffs, NJ: Prentice-Hall, 1989.

[26] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.

[27] T. Vicsek, A. Czirok, E. B. Jacob, I. Cohen, and O. Schochet, "Novel type of phase transitions in a system of self-driven particles," *Phys. Rev. Lett.*, vol. 75, no. 6, pp. 1226–1229, 1995.

[28] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.

[29] W. Ren and R. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 655–661, May. 2005.

[30] K. Liu, G. Xie, W. Ren, and L. Wang, "Consensus for multi-agent systems with inherent nonlinear dynamics under directed topologies," *Syst. Control Lett.*, vol. 63, no. 2, pp. 152–162, Feb. 2013.

[31] C. Huang and X. Ye, "A nonlinear transformation for reaching dynamic consensus in multi-agent systems," *IEEE Trans. Autom. Control*, vol. 60, no. 12, pp. 3263–3268, Dec. 2015.

[32] Y. Cao, W. Ren, D. W. Casbeer, and C. Schumacher, "Finite-time connectivity-preserving consensus of networked nonlinear agents with unknown lipschitz terms," *IEEE Trans. Autom. Control*, vol. 61, no. 6, pp. 1700–1705, Jun. 2016.

[33] R. Chipalkatty, M. Egerstedt, and S.-I. Azuma, "Multi-pendulum synchronization using constrained agreement protocols," in *Proc. Int. Conf. Robot. Commun. Coordinat.*, pp. 1–6, Odense, Denmark, March 31–April 2, 2009.

[34] C. Ma, P. Shi, X. Zhao, and Q. Zeng, "Consensus of Euler-Lagrange systems networked by sampled-data information with probabilistic time delays," *IEEE Trans. Cybern.*, vol. 45, no. 6, pp. 1126–1133, Jun. 2015.

[35] X. Zhao, C. Ma, X. Xing, and X. Zheng, "A stochastic sampling consensus protocol of networked Euler-Lagrange systems with application to two-link manipulator," *IEEE Trans. Ind. Informat.*, vol. 11, no. 4, pp. 907–914, Aug. 2015.

[36] J. Wu and Y. Shi, "Consensus in multi-agent systems with random delays governed by a Markov chain," *Syst. Control Lett.*, vol. 60, no. 11, pp. 863–870, Oct. 2011.

[37] ——, "Average consensus in multi-agent systems with time-varying delays and packet losses," in *Proc. Amer. Control Conf.*, pp. 1579–1584, Montréal, Canada, June 27–29, 2012.

[38] J. Wu, Y. Shi, and H. Li, "Consensus in multi-agent systems with non-uniform sampling," in *Proc. Amer. Control Conf.*, pp. 3260–3265, Washington, USA, June 17–19, 2013.

[39] F. Xiao, L. Wang, J. Chen, and Y. Gao, "Finite-time formation control for multi-agent systems," *Automatica*, vol. 45, no. 11, pp. 2605–2611, Nov. 2009.

[40] F. Xiao and L. Wang, "Asynchronous consensus in continuous-time multi-agent systems with switching topology and time-varying delays," *IEEE Trans. Autom. Control*, vol. 53, no. 8, pp. 1804–1816, Sept. 2008.

[41] B. Mu, H. Li, W. Li, and Y. Shi, "Consensus for multiple Euler-Lagrange dynamics with arbitrary sampling periods and event-triggered strategy," in *Proc. 11th World Congr. Intelligent Control and Automation*, pp. 2596–2601, Shenyang, Liaoning, China, June 27–30, 2014.

[42] F. Xiao, L. Wang, and J. Chen, "Partial state consensus for networks of second-order dynamic agents," *Syst. Control Lett.*, vol. 59, no. 12, pp. 775–781, Dec. 2010.

[43] C. E. Ren and C. L. P. Chen, "Sliding mode leader-following consensus controllers for second-order non-linear multi-agent systems," *IET Control Theory Appl.*, vol. 9, no. 10, pp. 1544–1552, Sept. 2015.

[44] Y. Zhao, Z. Duan, and G. Wen, "Finite-time consensus for second-order multi-agent systems with saturated control protocols," *IET Control Theory Appl.*, vol. 9, no. 3, pp. 312–319, Sept. 2015.

[45] H. Li, X. Liao, T. Huang, W. Zhu, and Y. Liu, "Second-order global consensus in multiagent networks with random directional link failure," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 565–575, Mar. 2015.

[46] X. Ai, S. Song, and K. You, "Second-order consensus of multi-agent systems under limited interaction ranges," *Automatica*, vol. 68, no. 6, pp. 329–333, Jun. 2016.

[47] W. Yu, G. Chen, W. Ren, J. Kurths, and W. X. Zheng, "Distributed higher order consensus protocols in multiagent dynamical systems," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 8, pp. 1924–1932, 2011.

[48] J. Xi, Z. Xu, G. Liu, and Y. Zhong, "Stable-protocol output consensus for high-order linear swarm systems with time-varying delays," *IET Control Theory Appl.*, vol. 7, no. 7, pp. 975–984, May. 2013.

[49] H. Rezaee and F. Abdollahi, "Average consensus over high-order multiagent systems," *IEEE Trans. Autom. Control*, vol. 60, no. 11, pp. 3047–3052, Nov. 2015.

[50] W. Ren, K. Moore, and Y. Chen, “High-order and model reference consensus algorithms in cooperative control of multi-vehicle systems,” *ASME J. Dyn. Syst., Meas., Control*, vol. 129, no. 5, pp. 678–688, Dec. 2007.

[51] S. Su and Z. Lin, “Distributed consensus control of multi-agent systems with higher order agent dynamics and dynamically changing directed interaction topologies,” *IEEE Trans. Autom. Control*, vol. 61, no. 2, pp. 515–519, Feb. 2016.

[52] L. Xiao, S. Boyd, and S.-J. Kim, “Distributed average consensus with least-mean-square deviation,” *J. Parallel Distrib. Comput.*, vol. 67, no. 1, pp. 33–46, Jan. 2007.

[53] M. Huang and J. H. Manton, “Stochastic consensus seeking with noisy and directed inter-agent communication: Fixed and randomly varying topologies,” *IEEE Trans. Autom. Control*, vol. 55, no. 1, pp. 235–241, Jan. 2010.

[54] Y. Zheng, Y. Zhu, and L. Wang, “Consensus of heterogeneous multi-agent systems,” *IET Control Theory Appl.*, vol. 5, no. 16, pp. 1881–1888, Apr. 2011.

[55] H. Grip, T. Yang, A. Saberi, and A. Stoorvogel, “Output synchronization for heterogeneous networks of non-introspective agents,” *Automatica*, vol. 48, no. 10, pp. 2444–2453, Oct. 2012.

[56] H. Kim, H. Shim, and J. H. Seo, “Output consensus of heterogeneous uncertain linear multi-agent systems,” *IEEE Trans. Autom. Control*, vol. 56, no. 1, pp. 200–206, Jan. 2011.

[57] H. Wang, “Second-order consensus of networked thrust-propelled vehicles on directed graphs,” *IEEE Trans. Autom. Control*, vol. 61, no. 1, pp. 222–227, Jan. 2016.

[58] M. Hutagalung and T. Hayakawa, “Attitude consensus of two underactuated spacecraft,” in *Proc. SICE Annual Conference*, 2008, pp. 3299–3304, Chofu, Tokyo, Japan, August 20–August 22, 2008.

[59] M. Hutagalung, T. Hayakawa, and T. Urakubo, “Configuration consensus of two underactuated planar rigid bodies,” in *Proc. IEEE Conf. Decision Control*, 2008, pp. 5016–5021, Cancun, Mexico, December 9–December 11, 2008.

[60] Y. Wang and Q. Sun, “Distributed path following control of multiple underactuated auvs via constructing an interconnected system structure approach,” in *Proc. IEEE 27th Canadian Conf. Elect. Comput. Eng.*, 2015, pp. 1000–1005, Halifax, NS, Canada, May 03–06, 2015.

[61] L. Moreau, “Stability of continuous-time distributed consensus algorithms,” in *Proc. IEEE Conf. Decision Control*, vol. 4, 2004, pp. 3998–4003, Atlantis, Paradise Island, Bahamas, December 1417, 2004.

[62] C. Wang, X. Wang, and H. Ji, “A continuous leader-following consensus control strategy for a class of uncertain multi-agent systems,” *IEEE/CAA J. Automatica Sinica*, vol. 1, no. 2, pp. 187–192, Apr. 2014.

[63] Y. Han, W. Lu, and T. Chen, “Achieving cluster consensus in continuous-time networks of multi-agents with inter-cluster non-identical inputs,” *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 793–798, Mar. 2015.

[64] S. Li, G. Feng, X. Luo, and X. Guan, “Output consensus of heterogeneous linear discrete-time multiagent systems with structural uncertainties,” *IEEE Trans. Cybern.*, vol. 45, no. 12, pp. 2868–2879, Dec. 2015.

[65] C. Tan and G. P. Liu, “Consensus of discrete-time linear networked multi-agent systems with communication delays,” *IEEE Trans. Autom. Control*, vol. 58, no. 11, pp. 2962–2968, Nov. 2013.

[66] M. Franceschelli, A. Giua, and C. Seatzu, “A gossip-based algorithm for discrete consensus over heterogeneous networks,” *IEEE Trans. Autom. Control*, vol. 55, no. 5, pp. 1244–1249, May. 2010.

[67] J. Zhan and X. Li, “Asynchronous consensus of multiple double-integrator agents with arbitrary sampling intervals and communication delays,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 9, pp. 2301–2311, Sept. 2015.

[68] L. Cheng, Z. G. Hou, and M. Tan, “A mean square consensus protocol for linear multi-agent systems with communication noises and fixed topologies,” *IEEE Trans. Autom. Control*, vol. 59, no. 1, pp. 261–267, Jan. 2014.

[69] Y. Su and J. Huang, “Cooperative output regulation with application to multi-agent consensus under switching network,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 3, pp. 864–875, Jun. 2012.

[70] D. Jakovetic, J. Xavier, and J. M. F. Moura, "Weight optimization for consensus algorithms with correlated switching topology," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3788–3801, Jul. 2010.

[71] B. Liu and T. Chen, "Consensus in networks of multiagents with cooperation and competition via stochastically switching topologies," *IEEE Trans. Neural Netw.*, vol. 19, no. 11, pp. 1967–1973, Nov. 2008.

[72] P. Lin and Y. Jia, "Consensus of a class of second-order multi-agent systems with time-delay and jointly-connected topologies," *IEEE Trans. Autom. Control*, vol. 55, no. 3, pp. 778–784, Mar. 2010.

[73] Y. G. Sun and L. Wang, "Consensus of multi-agent systems in directed networks with nonuniform time-varying delays," *IEEE Trans. Autom. Control*, vol. 54, no. 7, pp. 1607–1613, Jul. 2009.

[74] X. Liu, W. Lu, and T. Chen, "Consensus of multi-agent systems with unbounded time-varying delays," *IEEE Trans. Autom. Control*, vol. 55, no. 10, pp. 2396–2401, Oct. 2010.

[75] Y. Zhang and Y. P. Tian, "Consensus of data-sampled multi-agent systems with random communication delay and packet loss," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 939–943, Apr. 2010.

[76] M. Yu, C. Yan, D. Xie, and G. Xie, "Event-triggered tracking consensus with packet losses and time-varying delays," *IEEE/CAA J. Automatica Sinica*, vol. 3, no. 2, pp. 165–173, Apr. 2016.

[77] Y. Zhang and Y. P. Tian, "Maximum allowable loss probability for consensus of multi-agent systems over random weighted lossy networks," *IEEE Trans. Autom. Control*, vol. 57, no. 8, pp. 2127–2132, Aug. 2012.

[78] N. A. M. Subha and G. P. Liu, "External consensus in multi-agent systems with large consecutive data loss under unreliable networks," *IET Control Theory Appl.*, vol. 10, no. 9, pp. 989–1000, Oct 2016.

[79] W. Liu, F. Deng, J. Liang, and H. Liu, "Distributed average consensus in multi-agent networks with limited bandwidth and time-delays," *IEEE/CAA J. Automatica Sinica*, vol. 1, no. 2, pp. 193–203, Apr. 2014.

[80] C. N. Hadjicostis and T. Charalambous, "Average consensus in the presence of delays in directed graph topologies," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 763–768, Mar. 2014.

[81] Y. G. Sun, L. Wang, and G. Xie, "Average consensus in networks of dynamic agents with switching topologies and multiple time-varying delays," *Syst. Control Lett.*, vol. 57, no. 2, pp. 175–183, Feb. 2008.

[82] M. M. Zavlanos and G. J. Pappas, "Potential fields for maintaining connectivity of mobile networks," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 812–816, Aug. 2007.

[83] Y. Kuriki and T. Namerikawa, "Consensus-based cooperative formation control with collision avoidance for a multi-UAV system," in *Proc. Amer. Control Conf.*, 2014, pp. 2077–2082, Portland, Oregon, USA, June 4–6, 2014.

[84] ——, "Formation control with collision avoidance for a multi-UAV system using decentralized mpc and consensus-based control," in *Proc. Eur. Control Conf.*, 2015, pp. 3079–3084, Linz, Austria, June 15–17, 2015.

[85] L. Wang and F. Xiao, "Finite-time consensus problems for networks of dynamic agents," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 950–955, Apr. 2010.

[86] S. Khoo, L. Xie, and Z. Man, "Robust finite-time consensus tracking algorithm for multirobot systems," *IEEE/ASME Trans. Mechatronics*, vol. 14, no. 2, pp. 219–228, Apr. 2009.

[87] D. Meng, Y. Jia, and J. Du, "Finite-time consensus for multiagent systems with cooperative and antagonistic interactions," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 4, pp. 762–770, Apr. 2016.

[88] S. Yu and X. Long, "Finite-time consensus tracking of perturbed high-order agents with relative information by integral sliding mode," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 63, no. 6, pp. 563–567, June 2016.

[89] B. Francis and W. Wonham, "The internal model principle of control theory," *Automatica*, vol. 12, no. 5, pp. 457–465, Sept. 1976.

[90] Z. Ding, "Output regulation of uncertain nonlinear systems with nonlinear exosystems," *IEEE Trans. Autom. Control*, vol. 51, no. 3, pp. 498–503, Mar. 2006.

[91] ——, "Consensus output regulation of a class of heterogeneous nonlinear systems," *IEEE Trans. Autom. Control*, vol. 58, no. 10, pp. 2648–2653, Oct. 2013.

[92] H. Yang, Z. Zhang, and S. Zhang, "Consensus of second-order multi-agent systems with exogenous disturbances," *Int. J. Robust Nonlinear Control*, vol. 21, no. 9, pp. 945–956, Jun. 2011.

[93] N. Ili, M. S. Stankovi, and S. S. Stankovi, "Adaptive consensus-based distributed target tracking in sensor networks with limited sensing range," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 2, pp. 778–785, Mar. 2014.

[94] S. Nosrati, M. Shafiee, and M. B. Menhaj, "Dynamic average consensus via nonlinear protocols," *Automatica*, vol. 48, no. 9, pp. 2262–2270, Sept. 2012.

[95] Y. Cao and W. Ren, "Optimal linear-consensus algorithms: An LQR perspective," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, pp. 819–830, Jun. 2010.

[96] J. Ma, Y. Zheng, and L. Wang, "LQR-based optimal topology of leader-following consensus," *Int. J. Robust Nonlinear Control*, vol. 25, no. 17, pp. 3404–3421, Nov. 2015.

[97] G. Ferrari-Trecate, L. Galbusera, M. P. E. Marciandi, and R. Scattolini, "Model predictive control schemes for consensus in multi-agent systems with single- and double-integrator dynamics," *IEEE Trans. Autom. Control*, vol. 54, no. 11, pp. 2560–2572, Nov. 2009.

[98] J. Zhan and X. Li, "Consensus of sampled-data multi-agent networking systems via model predictive control," *Automatica*, vol. 49, no. 8, pp. 2502–2507, Aug. 2013.

[99] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1291–1297, May. 2012.

[100] A. Wang, B. Mu, and Y. Shi, "Consensus control for a multi-agent system with integral-type event-triggering condition and asynchronous periodic detection," *IEEE Trans. Ind. Electron.*, vol. 64, no. 7, pp. 5629–5639, Jul. 2017.

[101] M. Zhao, C. Peng, W. He, and Y. Song, "Event-triggered communication for leader-following consensus of second-order multiagent systems," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–10, 2017.

[102] W. Xu, D. W. C. Ho, L. Li, and J. Cao, "Event-triggered schemes on leader-following consensus of general linear multiagent systems under different topologies," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 212–223, Jan 2017.

[103] X. Zhang, M. Chen, and L. Wang, "Distributed event-triggered consensus in multi-agent systems with non-linear protocols," *IET Control Theory Appl.*, vol. 9, no. 18, pp. 2626–2633, Nov. 2015.

[104] H. Zhang, G. Feng, H. Yan, and Q. Chen, "Observer-based output feedback event-triggered control for consensus of multi-agent systems," *IEEE Trans. Ind. Electron.*, vol. 61, no. 9, pp. 4885–4894, Sept. 2014.

[105] Y. Fan, L. Liu, G. Feng, and Y. Wang, "Self-triggered consensus for multi-agent systems with zeno-free triggers," *IEEE Trans. Autom. Control*, vol. 60, no. 10, pp. 2779–2784, Oct. 2015.

[106] S. Rao and D. Ghose, "Sliding mode control-based algorithms for consensus in connected swarms," *Int. J. Control*, vol. 84, no. 9, pp. 1477–1490, Sept. 2011.

[107] W. Chen, X. Li, W. Ren, and C. Wen, "Adaptive consensus of multi-agent systems with unknown identical control directions based on a novel nussbaum-type function," *IEEE Trans. Autom. Control*, vol. 59, no. 7, pp. 1887–1892, Jul. 2014.

[108] H. Zhang, J. Zhang, G. H. Yang, and Y. Luo, "Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 1, pp. 152–163, Feb. 2015.

[109] S. M. Kang and H. S. Ahn, "Design and realization of distributed adaptive formation control law for multi-agent systems with moving leader," *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1268–1279, Feb. 2016.

[110] H. G. Tanner and A. Boddu, "Multiagent navigation functions revisited," *IEEE Trans. Robot.*, vol. 28, no. 6, pp. 1346–1359, Dec. 2012.

[111] E. Montijano, J. Thunberg, X. Hu, and C. Sags, "Epipolar visual servoing for multirobot distributed consensus," *IEEE Trans. Robot.*, vol. 29, no. 5, pp. 1212–1225, Oct. 2013.

[112] L. Vig and J. A. Adams, "Multi-robot coalition formation," *IEEE Trans. Robot.*, vol. 22, no. 4, pp. 637–649, Aug. 2006.

[113] B. Ranjbar-Sahraei, F. Shabaninia, A. Nemati, and S. D. Stan, "A novel robust decentralized adaptive fuzzy control for swarm formation of multiagent systems," *IEEE Trans. Ind. Electron.*, vol. 59, no. 8, pp. 3124–3134, Aug. 2012.

[114] Y. Cao and W. Ren, "Sampled-data discrete-time coordination algorithms for double-integrator dynamics under dynamic directed interaction," *Int. J. Control*, vol. 20, no. 3, pp. 506–515, Nov. 2009.

[115] G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin, "Precision flight control for a multi-vehicle quadrotor helicopter testbed," *Control Eng. Pract.*, vol. 19, no. 9, pp. 1023–1036, Sept. 2011.

[116] H. Lim, J. Park, D. Lee, and H. J. Kim, "Build your own quadrotor: Open-source projects on unmanned aerial vehicles," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 33–45, Sept. 2012.

[117] Park 480 brushless outrunner motor, 1020kv. [Online]. Available: http://www.e-fliterc.com/Products/Default.aspx?ProdID=EFLM1505

[118] Reverse rotation APC 10x4.7SFP propeller. [Online]. Available: http://www.apcprop.com/product_p/lp10047sfp.htm

[119] Hobbyking brushless ESC user manual. [Online]. Available: https://hobbyking.com/media/file/1006513628X70599X0.pdf

[120] Logitech quickcam pro 9000 usb camera. [Online]. Available: https://support.logitech.com/en_us/product/quickcam-pro-9000/specs

[121] F. Xiao and T. chen, "Sampled-data consensus for multiple double integrators with arbitrary sampling," *IEEE Trans. Autom. Control*, vol. 57, no. 12, pp. 3230–3235, Dec. 2012.

[122] J. Xu, Z. Guo, and T. H. Lee, “Design and implementation of integral sliding-mode control on an underactuated two-wheeled mobile robot,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 7, pp. 3671–3681, Jul. 2014.

[123] C. Moon and W. Chung, “Kinodynamic planner dual-tree RRT (DT-RRT) for two-wheeled mobile robots using the rapidly exploring random tree,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 2, pp. 1080–1090, Feb. 2015.

[124] J. Wolfowitz, “Products of indecomposable, aperiodic, stochastic matrices,” *Proc. Amer. Math. Soc.*, vol. 14, no. 5, pp. 733–737, Oct. 1963.

[125] Y. Gao and L. Wang, “Asynchronous consensus of continuous-time multi-agent systems with intermittent measurements,” *Int. J. Control*, vol. 83, no. 3, pp. 552–562, Mar. 2010.

[126] Y. Gao, M. Zuo, T. Jiang, J. Du, and J. Ma, “Asynchronous consensus of multiple second-order agents with partial state information,” *Int. J. Syst. Sci.*, vol. 44, no. 5, pp. 966–977, May. 2013.

[127] S. M. Dibaji and H. Ishii, “Resilient consensus of second-order agent networks: Asynchronous update rules with delays,” *Automatica*, vol. 81, pp. 123–132, Jul. 2017.

[128] Y. Fan, G. Feng, Y. Wang, and C. Song, “Distributed event-triggered control of multi-agent systems with combinational measurements,” *Automatica*, vol. 49, no. 2, pp. 671–675, Feb. 2013.

[129] Z. Zhang, Z. Zhang, and H. Zhang, “Finite-time stability analysis and stabilization for uncertain continuous-time system with time-varying delay,” *J. Franklin Inst.*, vol. 352, no. 3, pp. 1296–1317, Mar. 2015.

[130] F. Xiao and L. Wang, “Consensus protocols for discrete-time multi-agent systems with time-varying delays,” *Automatica*, vol. 44, no. 10, pp. 2577–2582, 2008.

[131] J. Qin and H. Gao, “A sufficient condition for convergence of sampled-data consensus for double-integrator dynamics with nonuniform and time-varying communication delays,” *IEEE Trans. Autom. Control*, vol. 57, no. 9, pp. 2417–2422, Sept. 2012.

[132] A. Eqtami, D. V. Dimarogonas, and K. Kyriakopoulos, "Event-based model predictive control for the cooperation of distributed agents," in *Proc. Amer. Control Conf.*, 2012, pp. 6473–6478, Montréal, Canada, June 27–29, 2012.

[133] Z. Zhang, Z. Zhang, H. Zhang, B. Zheng, and H. R. Karimi, "Finite-time stability analysis and stabilization for linear discrete-time system with time-varying delay," *J. Franklin Inst.*, vol. 351, no. 6, pp. 3457–3476, Jun. 2014.

[134] Y. Zheng and L. Wang, "Containment control of heterogeneous multi-agent systems," *Int. J. Control*, vol. 87, no. 1, pp. 1–8, Jul. 2013.

[135] G. Wen, J. Huang, C. Wang, Z. Chen, and Z. Peng, "Group consensus control for heterogeneous multi-agent systems with fixed and switching topologies," *Int. J. Control*, vol. 89, no. 2, pp. 259–269, Aug. 2015.

[136] D. S. Naidu, *Optimal Control Systems.* Boca Raton: CRC press, 2003.

[137] L. R. G. Carrillo, R. Lozano, A. E. D. López, and C. Pégard, *Quad Rotorcraft Control: Vision-Based Hovering and Navigation.* Springer, 2013.

[138] P. H. Zipfel, *Modeling and Simulation of Aerospace Vehicle Dynamics.* American Institute of Aeronautics and Astronautics, 2007.

[139] F. Verhulst, *Nonlinear Differential Equations and Dynamical Systems.* Springer, 1996.

[140] S. Aouaouda, M. Chadli, and H. R. Karimi, "Robust static output-feedback controller design against sensor failure for vehicle dynamics," *IET Control Theory Appl.*, vol. 8, no. 9, pp. 728–737, Jun. 2014.

[141] H. Li, Y. Shi, and W. Yan, "Distributed receding horizon control of constrained nonlinear vehicle formations with guaranteed γ-gain stability," *Automatica*, vol. 68, pp. 148–154, Jan. 2016.

[142] H. Li and Y. Shi, *Robust Receding Horizon Control for Networked and Distributed Nonlinear Systems.* Springer, 2017.

[143] 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 *Proc. IEEE Int. Conf. Autom. Quality Testing Robot. Conf.*, vol. 1, Jun. 2010, pp. 1–5.

[144] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pp. 2451–2456, Sendal, Japan, September 28–October 2, 2004.

[145] F. Chen, Q. Wu, B. Jiang, and G. Tao, "A reconfiguration scheme for quadrotor helicopter via simple adaptive control and quantum logic," *IEEE Trans. Ind. Electron.*, vol. 62, no. 7, pp. 4328–4335, Jul. 2015.

[146] J. Chang, J. Cieslak, J. Davila, A. Zolghadri, and J. Zhou, "Adaptive second-order sliding mode observer for quadrotor attitude estimation," in *Proc. Amer. Control Conf.*, pp. 2246–2251, Boston, MA, USA, July 6–8, 2016.

[147] F. Chen, R. Jiang, K. Zhang, B. Jiang, and G. Tao, "Robust backstepping sliding-mode control and observer-based fault estimation for a quadrotor UAV," *IEEE Trans. Ind. Electron.*, vol. 63, no. 8, pp. 5044–5056, Aug. 2016.

[148] X. Zhang, B. Xian, B. Zhao, and Y. Zhang, "Autonomous flight control of a nano quadrotor helicopter in a GPS-denied environment using on-board vision," *IEEE Trans. Ind. Electron.*, vol. 62, no. 10, pp. 6392–6403, Oct. 2015.

[149] F. Chen, W. Lei, K. Zhang, G. Tao, and B. Jiang, "A novel nonlinear resilient control for a quadrotor UAV via backstepping control and nonlinear disturbance observer," *Nonlinear Dyn.*, vol. 85, no. 2, pp. 1281–1295, Jul. 2016.

[150] F. Chen, K. Zhang, B. Jiang, and C. Wen, "Adaptive sliding mode observer-based robust fault reconstruction for a helicopter with actuator fault," *Asian J. Control*, vol. 18, no. 4, pp. 1–8, Jul. 2016.

[151] C. E. Doyle, J. J. Bird, T. A. Isom, J. C. Kallman, D. F. Bareiss, D. J. Dunlop, R. J. King, J. J. Abbott, and M. A. Minor, "An avian-inspired passive mechanism for quadrotor perching," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 2, pp. 506–517, Apr. 2013.

[152] Q. Quan, G. X. Du, and K. Y. Cai, "Proportional-integral stabilizing control of a class of mimo systems subject to nonparametric uncertainties by additive-state-decomposition dynamic inversion design," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 2, pp. 1092–1101, Apr. 2016.

[153] C. Edwards and S. K. Spurgeon, *Sliding Mode Conrol: Theory and Applications*. London: Taylor & Francis, 1998.

[154] M. T. Hamayun, C. Edwards, and H. Alwi, *Fault Tolerant Control Schemes Using Integral Sliding Modes.* Cham: Springer International Publishing AG Switzerland, 2016.

[155] K. Nonaka and H. Sugizaki, "Integral sliding mode altitude control for a small model helicopter with ground effect compensation," in *Proc. Amer. Control Conf.*, pp. 202–207, San Francisco, CA, USA, June 29–July 01, 2011.

[156] H. J. Savino, C. R. P. dos Santos, F. O. Souza, L. C. A. Pimenta, M. de Oliveira, and R. M. Palhares, "Conditions for consensus of multi-agent systems with time-delays and uncertain switching topology," *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1258–1267, Feb. 2016.

[157] H. Li, Y. Shi, and W. Yan, "On neighbor information utilization in distributed receding horizon control for consensus-seeking," *IEEE Trans. Cybern.*, vol. 46, no. 9, pp. 2019–2027, Sept. 2016.

[158] Y. Wei, J. Qiu, and H. R. Karimi, "Reliable output feedback control of discrete-time fuzzy affine systems with actuator faults," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 1, pp. 170–181, Jan 2017.

[159] Y. Wei, J. Qiu, H. R. Karimi, and M. Wang, "New results on $\mathcal{H}_\infty$ dynamic output feedback control for Markovian jump systems with time-varying delay and defective mode information," *Optimal Control Appl. Methods*, vol. 35, no. 6, pp. 656–675, Oct. 2014.

[160] J. Zhang, X. Liu, Y. Xia, Z. Zuo, and Y. Wang, "Disturbance observer-based integral sliding-mode control for systems with mismatched disturbances," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7040–7048, Nov. 2016.