COOPERATIVE AERIAL MANIPULATION WITH FORCE CONTROL AND ATTITUDE STABILIZATION

By

SANDESH THAPA

Bachelor of Science in Mechanical Engineering

McNeese State University

Lake Charles, Louisiana
2015

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 2018

COOPERATIVE AERIAL MANIPULATION WITH FORCE CONTROL AND ATTITUDE STABILIZATION

Thesis Approved:

Dr. He Bai

Thesis Advisor and Committe Chair

Dr. Rushikesh Kamalapukar

Committee Member

Dr. Jerome Hausselle

Committee Member

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. He Bai for his time and patience in guiding me through this research. It has been a tremendous honor to work with him. His direction and consistent support helped me develop a clear framework for this research. He was always kind when I did something, and even kind when I did nothing at all.

I would also like to thank Dr. José Á. Acosta from University of Sevilla, Spain for his consistent support and creative ideas during this project over the last few years. His expertise in aerial manipulation made our project a reality. I would like to thank him for his time and multiple skype calls which laid a clear foundation of our project.

I would also like to thank all of my lab mates for their help throughout this project. In particular, I would to thank Collin Thornton and Sam Allison in helping me building first prototype of our experimental platform. Despite my misgivings and breaking the landing gear, they were always supportive and helped me throughout the entire process.

Additionally, I would like to thank my committee members; Dr. Rushikesh Kamalapurkar and Dr. Jerome Hausselle, for taking their time to serve in my committee, and am very for their valuable input.

Finally, I would like to thank my parents and my siblings for their continued support and patience over the years. I would like to thank my sister Anita, for her support during my time here in OSU and the beautiful meals she cooked during my numerous trips to Dallas.

Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Name: SANDESH THAPA

Date of Degree: DECEMBER, 2018

Title of Study: COOPERATIVE AERIAL MANIPULATION WITH FORCE CONTROL AND ATTITUDE STABILIZATION

Major Field: MECHANICAL & AEROSPACE ENGINEERING

Abstract: Ranging from autonomous flying cars, fixed wing and rotorcraft UAVs, there has been a tremendous interest in aerial robotics over the last decade. This thesis presents contributions to the state-of-art in cooperative payload transport with force synthesis and dynamic interaction using quadcopter UAVs. In this report, we consider multiple quadcopter aerial robots and develop decentralized force controller for them to manipulate a payload. We use quadcopters with a rigid link attached to it to collaborately manipulate the payload. We develop a dynamic model of the payload for both point mass and rigid body cases. We model the contact force between the agents and the payload as a mass spring model. This assumption is valid when the vehicles are connected to the payload via elastic cables or when the payload is flexible or surrounded by elastic bumper materials. We also extend our aerial manipulation system to a multi-link arm attached to the quadcopter.

We develop an adaptive decentralized control law for transporting a payload of unknown mass without explicit communication between the agents. Our controller ensures that all quadcopters and the payload asymptotically converges to a constant reference velocity. It also ensures that all of the forces applied to the payload converges to desired set-points. Desired thrusts and attitude angles are computed from the control algorithms and a low-level PD controller is implemented to track the desired commands for each quadcopter. The sum of the estimates of the unknown mass from all the agents converge to the true mass. We also employ a consensus algorithm based on connected graphs to ensure that each agent gets an equal share of the payload mass. Furthermore, we develop an orientation control algorithm that guarantees attitude stabilization of the payload. In particular, we develop time varying force set-points to enforce attitude regulation without any moment inputs from the quadcopters.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER 1

INTRODUCTION

Aerial robotics has a tremendous potential to impact the world in a positive way. In this report, we focus on cooperative aerial manipulation using multi-rotors. In particular, we focus on quadcopters because of their popularity and accessibility.

Quadcopters are a type of small unmanned rotorcraft with four rotors. It is an underactuated mechanical system and has six degrees of freedom with only four control inputs. Thus, quadcopters are dynamically unstable. This rich dynamics and mechanical simplicity of quadcopters can be exploited for agile and dynamic maneuvers in constrained spaces.

Quadcopter aerial robots have become increasingly popular because of their superior mobility in three dimensional space and high thrust-to-weight ratio. Because of their size and hovering properties, quadcopters can operate in constrained spaces such as damaged buildings, adversarial and nuclear environments. They can be used to collect information from the environment and can also dynamically interact with the environment using grippers. Possible applications of such robots include disaster response, transport, photography, search and rescue operations, construction and so on.

Multiple quadcopters can be employed together to increase the capability of a single quadcopter, which enables better manipulation and control of heavier payloads for various applications. Cooperative aerial manipulation of a payload requires motion coordination, regulation of forces applied to the payload by the quadcopters, attitude regulation of the payload and multirotors and analysis of stability of the combined system. This thesis presents a framework for force regulation, motion coordination and orienta-

tion control of the payload during cooperative aerial manipulation.

1.1 Motivation

Ranging from autonomous flying cars, fixed wing and rotor-craft UAVs, there has been tremendous interest in aerial robotics over the last decade. Application of aerial robotic systems range from military and civilian sectors. The ability to manipulate in 3D allows multiple dynamic tasks like aerial gripping, manipulation, photography and so on. Using quadcopter aerial robots dynamic interactions, vertical take off and landing, hovering in constraint spaces are possible. Recently, we have already seen the commercial applications of such aerial robots in commercial sectors. If such commercial applications like air package delivery [2] is to be realistic, it should be able to handle both rigid and flexible payloads and the dynamic interaction between the quadcopter and payload. It is clear that the aerial robotics community lacks a general method of force and dynamic motion synthesis during cooperative payload transport.

Motivated by the current challenges and their potential applications, in this work, we design decentralized adaptive force control algorithm for multiple quadcopters to manipulate a load of an unknown mass. We develop a complete motion control algorithm for cooperative payload transport to address both the orientation control of the payload and the force control during cooperative aerial manipulation [3–6]. The developed algorithm achieves asymptotic tracking of a desired velocity, controls the applied forces to the payload to predefined set-points and ensures that the individual estimate of the mass from each agent converges to the total mass divided by the number of agents. We develop an attitude control algorithm for the payload using time-varying force control without no any moments input from the rotors.

1.2 Related Work

Because of the complexity of the problem, cooperative aerial manipulation has recently become a popular research area (see for example [7–11]). Two distinct types of framework exist in the literature. Inspired by the traditional approach of helicopter-based towing via cables for load transport, most of the work in cooperative aerial load transport has been focused on suspended payload transport using cables [7, 8, 11–13]. Recently, quadcopters are quipped with robotic manipulators and control laws are designed for dynamic interactions, perching and aerial gripping [14–17].

Load Transport via towed cables has been extensively studied by researchers worldwide [7, 11, 12, 18]. Reference [11] develops a geometric controller to follow a reference trajectory of a suspended payload with multiple quadcopters. The authors in [12] develop quasi-static models for payload transport via cables at a desired pose. Reference [7] develops a complete dynamic model of cooperative aerial manipulation and exploits the differential flatness property of the system to generate smooth trajectories for path planning. Reference [8] develops a software architecture for planning, control, collision avoidance and deployment of cable suspended payloads using multiple helicopters. Recently, [13] develops a passivity based formation control for multiple UAVs cooperatively carrying a suspended payload.

Aerial load transport via towed cables is a hybrid system and there are problems with swing load attenuation. So, different control laws have to be designed when the cable is tout and flexible. One possible solution to this problem is attach a robotic manipulator to a Unmanned Aerial Vehicle (UAV) and develop an aerial manipulator (AM). Since each AM is a coupled system, the control problem now becomes more challenging but it introduces more dynamic features to the system such as better gripping and control of payload as opposed to aerial towing. In one the earlier works, the authors in [19] develop gripping mechanisms and control laws for cooperative grasping and manipulation of a group of quadcopters. In some recent work, references [16,17,20] explore the problem of

cooperative aerial load transport by using multirotors attached with multi-DOF robotic arms and develop robust control and motion planning algorithms.

There also exist control strategies based on port Hamiltonian framework [21, 22] towards control of aerial manipulators. Acosta *et al.* propose an Interconnection and Damping Assignment Passivity Based Control (IDA-PBC) based on energy shaping methodology by exploiting the passivity of aerial manipulator. Nonlinear matching types of controller developed in [22] utilizes the cyclo-passivity property of AMs and combines the robust controllers for the quadcopter and the manipulator by exploiting their stability margins and optimizing different prescribed criteria in real time.

On the other hand, Wang and Schwager [23] develop a scalable, decentralized control strategy by which a large number of ground mobile robots can manipulate a comparatively large object through a desired trajectory without explicit communication network among the robots. Reference [24] considers a decentralized velocity controller to transport a common payload by using a team of robots in $\mathbb{R}^2$ rigidly attached to the payload. The authors in [25] develop decentralized adaptive controller to transport a rigid payload in $\mathbb{R}^2$ or $\mathbb{R}^3$ by controlling the spatial velocity of the payload.

Inspired by aerial towing and integration of robotic manipulator we suggested cooperative aerial manipulation with rigid extension. This approach allows better control of the payload and there is no problem of swing load attenuation. We can also extend the rigid link to a multi-link robot and develop similar manipulation system. In aerial manipulator there is a lot of redundancy in the system, this approach helps us reduce the redundancy of the system and develop a unified framework for cooperative aerial manipulation. As a possible example of our suggestions, recently, reference [26] develops a distributed wrench controller that enables a group of rigidly attached quadcopters to manipulate heavy payloads along a predefined trajectory without communication between the robots.

1.3 Contribution

Most of the aforementioned references have not considered dynamic interactions and force control between the payload and the quadcopters. In this paper, we develop an unified framework for cooperative aerial manipulation. Reference [27] considers a ring shaped flexible structure with multiple attached flying vehicles and develops a LQR controller for linearized dynamics to track the desired pose of the payload. It is assumed that the vehicles can exert both forces and moments on the payload. However, we assume that the inputs from the quadcopters to the payload are only contact forces with no moments. In particular, we develop force, motion and orientation control algorithm that grantees force regulation, attitude stabilization and translational motion of the quadcopters and the payload. This thesis makes the following contributions:

- We develop adaptive decentralized controller for transport of unknown payload without communication between agents. The sum of estimates of each payload converges to the true mass.
- We develop force consensus along the vertical direction such that the agents share the mass of the payload equally.
- We develop a force and orientation control algorithm that guarantees attitude stabilization of the payload and the convergence of the contact force and the velocity of each quadcopter to desired setpoints. We develop time varying force setpoints to enforce the orientation regulation of the payload.

1.4 Thesis Organization

This thesis presents force, motion and orientation control algorithms during cooperative aerial manipulation. The remainder of the paper is structured as follows. In Chapter 2, we develop a complete dynamic model for payload transportation using point

mass model. In Chapter 3, we present nominal design for force control law while transporting a payload of known mass. In Chapter 4, we consider quadcopters with rigid extension and develop adaptive decentralize control laws. We also employ a consensus algorithm based on connected graphs to ensure that each agent gets an equal share of the payload mass. Desired thrusts and attitude angles are computed from the control algorithms and a low-level PD controller is implemented to track the desired commands for each quadcopter. Furthermore, in Chapter 5 we also develop orientation control algorithm that guarantees attitude stabilization of the payload. In particular, we develop time varying force setpoints to enforce attitude regulation. In Chapter 6, we developed complete dynamic model for payload transport using aerial manipulator. Using the inverse kinematics of the aerial manipulator, we implement the developed algorithm in Chapter 4 at the kinematic level for the aerial manipulators and demonstrate its effectiveness in simulations. We include the preliminary experimental results in Chapter 7. We present conclusion and future work in Chapter 8.

CHAPTER 2

PROBLEM FORMULATION

In this chapter we develop a dynamic model of the aerial manipulation system. We consider a group of quadcopters with a rigid extension and develop force control law without explicit communication between the agents.

2.1 Payload Transportation with Quadcopter with rigid extension

Consider N agents holding a common flexible load as shown in Fig. 2.1 ($N = 3$). Each agent is a quadcopter with a rigid link extension. We model the payload as a point mass and assume that it is initially undeformed. Agent i is attached to the load at the point a_i. Let $x_i \in \mathbb{R}^3$ be the position of the end-effector of agent i in the inertial frame and $r_i \in \mathbb{R}^3$ be a fixed vector in the body frame of the load. Initially, $x_i(0) = a_i(0) = x_c(0) + r_i$, where $x_c \in \mathbb{R}^3$ is the position of the center of mass of the load in the inertial frame. Fig. 2.1 also shows the coordinate system defined to derive the kinematics of the system. Σ_I is the world fixed inertial frame and $O_{c,i}$ is the body-fixed frame attached to each agent i.

2.2 Kinematics

The position of the attaching point by agent i is given by

$$a_i(t) := x_c(t) + r_i. \tag{2.1}$$

The kinematics of this point is given by

$$\dot{a_i} = \dot{x_c}. \tag{2.2}$$

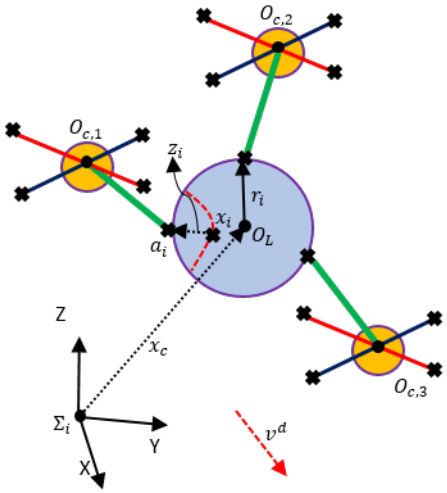


Figure 2.1: Three quadcopters transport a common flexible load. Note that a_i is the initial position of agent i.

As the quadcopters interact with the load, the load may be deformed. Therefore, $a_i(t) \neq x_i(t)$ and the load experiences contact forces due to the deformation. As shown in Fig. 2.1, we approximate the deformation as

$$z_i = x_i - a_i, \quad i = 1, \ldots, N. \tag{2.3}$$

We consider that the reaction force f_i exerted by each quadcopters is obtained by the gradient of some positive definite function $P_i(z_i)$, i.e.,

$$f_i = \nabla P_i(z_i). \tag{2.4}$$

We also assume that $P_i(z_i)$ satisfies

$$P_i(z_i) = 0 \iff z_i = 0, \tag{2.5}$$

$$\nabla P_i(z_i) = 0 \iff z_i = 0. \tag{2.6}$$

The contact force model in (2.4) includes the linear spring models used in [28, 29] if the contact force is elastic.

2.3 Dynamics

The translational dynamics of the load can be expressed as

$$M_c \ddot{x}_c = \sum_{i=1}^{N} f_i - M_c g e_3, \tag{2.7}$$

where g is the gravitational acceleration, M_c denotes the mass of the payload, and e_3 is the unit vector $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$. In (2.7), we approximate the payload dynamics as a rigid body. This is valid when a load is surrounded by elastic or deformable materials and the deformations are small.

The translational dynamics for each agent i can be modeled as

$$m_i \ddot{x}_i = F_i - f_i - m_i g e_3, \quad i = 1, ..., N, \tag{2.8}$$

where F_i denotes the force applied by agent i, m_i denotes the mass of agent i, and f_i is the force exerted to agent i from the load. The applied force F_i is given by

$$F_i = T_i R_i(\Phi) e_3, \tag{2.9}$$

where

$$T_i = \sum_{i=1}^{4} F_{\text{rotors},i}, \tag{2.10}$$

$\Phi = (\phi, \theta, \psi)$, which are the roll, pitch and yaw $(Z-Y-X)$ angles, respectively, and $R_i(\Phi) \in SO(3)$ is the rotation matrix that relates the body frame $O_{c,i}$ and the inertial frame Σ_i, which is given by

$$R_i = \begin{bmatrix} c\psi_i c\theta_i & c\psi_i s\phi_i s\theta_i - c\phi_i s\psi_i & s\phi_i s\psi_i + c\phi_i c\psi_i s\theta_i \\ c\theta_i s\psi_i & c\phi_i c\psi_i + s\phi_i s\psi_i s\theta_i & c\phi_i s\psi_i s\theta_i - c\psi_i s\phi_i \\ -s\theta_i & c\theta_i s\phi_i & c\phi_i c\theta_i \end{bmatrix}, \tag{2.11}$$

in which $c\theta_i = \cos\theta_i$ and $s\theta_i = \sin\theta_i$.

2.3.1 Singularities

A fundamental problem with Euler angles is their singularity. This is also well known as *gimbal lock* [1]. This occurs when the rotational axis of the middle term in the sequence becomes parallel to the rotation axis of the first or third term. For ZYX Euler angles, singularity occurs at $\theta_i = \pm(2k+1)\frac{\pi}{2}$, $k = 0, 1, 2, \cdots$.

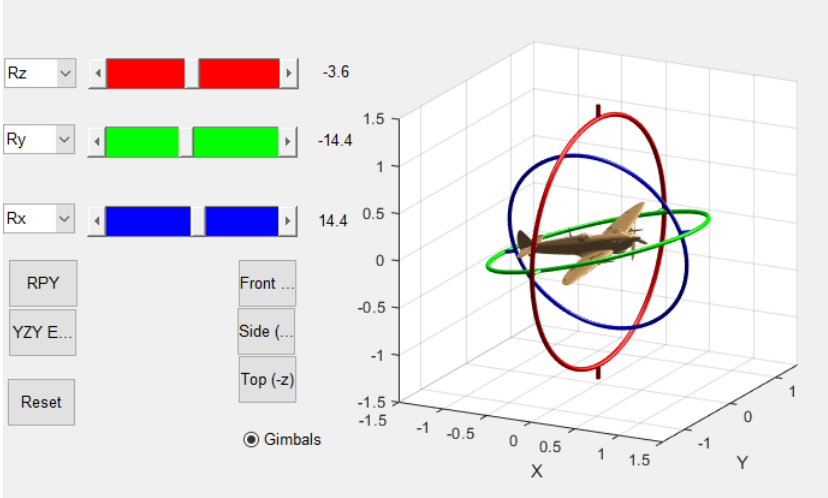


Figure 2.2: ZYX Euler angle representation using *tripleangle* toolbox [1]

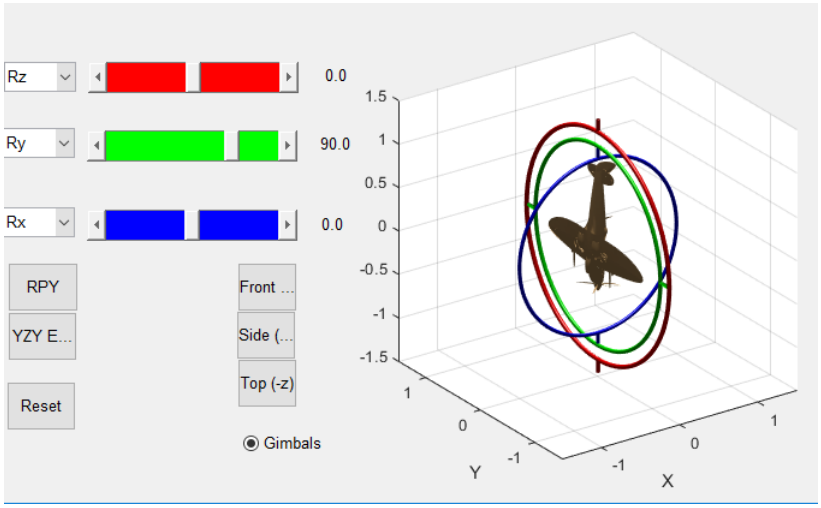


Figure 2.3: ZYX Euler Angle Representation and Gimbal lock. Note $\theta_i = 90°$

2.4 Control Objective

The control objective is to design the desired thrusts T_i and the desired attitude angles Φ_i in a decentralized way such that all the agents and the payload converge to a constant velocity v^d and the contact force f_i is regulated to a setpoint f_i^d, $i = 1, \cdots, N$. Since the load converges to a constant velocity, the sum of f_i^d's in x and y direction is zero, respectively, and the sum of f_i^d's in the z direction balances the weight of the load, i.e.,

$$\sum_{i=1}^{N} f_i^d = M_c g e_3. \tag{2.12}$$

We assume that for a given f_i^d there exists a locally unique deformation z_i^d, such that

$$f_i^d = \nabla P_i(z_i^d) \quad \text{and} \quad \nabla^2 P_i(z_i^d) > 0. \tag{2.13}$$

Assumption (2.13) is satisfied by linear spring-force model as well as certain classes of nonlinear models, such as $f_i = b_i |z_i|^2 z_i$, with $b_i > 0$.

2.5 Quadcopter Attitude Controller

We will use the following attitude control for rest of this report. We model the attitude dynamics of each quadcopter as [30]:

$$\begin{bmatrix} \ddot{\phi}_i \\ \ddot{\theta}_i \\ \ddot{\psi}_i \end{bmatrix} = \begin{bmatrix} \frac{I_y - I_z}{I_x} \dot{\theta}_i \dot{\psi}_i + \frac{1}{I_x} \tau_{\phi_i} \\ \frac{I_z - I_x}{I_y} \dot{\phi}_i \dot{\psi}_i + \frac{1}{I_y} \tau_{\theta_i} \\ \frac{I_x - I_y}{I_z} \dot{\theta}_i \dot{\psi}_i + \frac{1}{I_x} \tau_{\psi_i} \end{bmatrix} \tag{2.14}$$

where ϕ_i, θ_i and ψ_i are the roll, pitch and yaw angles for the i-th quadcopter. τ_{ϕ_i}, τ_{θ_i} and τ_{ψ_i} are the moments in x, y and z directions respectively and I_x, I_y and I_z are the moments of inertia of the quadcopter in its respective directions. The desired attitude Φ_i^{des} is given by,

$$\Phi_i^{des} = \begin{bmatrix} \phi_i^{des} \\ \theta_i^{des} \\ \psi_i^{des} \end{bmatrix}. \tag{2.15}$$

We set ψ_i^{des} as constant and solve for θ_i^{des} and ϕ_i^{des} from (3.6) and (3.7) respectively for both cases presented in Section VI.

To track the desired attitude angles computed from our force controllers in Section 3.1 and 6.3, we follow a Proportional-derivative (PD) controller from [31] with some minor changes. The controller is modeled as:

$$\begin{bmatrix} T_i \\ \tau_{\phi_i} \\ \tau_{\theta_i} \\ \tau_{\psi_i} \end{bmatrix} = \begin{bmatrix} T_i^{des} \\ \frac{I_x}{L}(-K_1^i\dot{\phi}_i - \frac{I_y-I_z}{I_x}\dot{\theta}_i\dot{\psi}_i + K_{p,1}^i\tilde{\phi}_i + K_{d,1}^i\dot{\tilde{\phi}}_i) \\ \frac{I_y}{L}(-K_2^i\dot{\theta}_i - \frac{I_z-I_x}{I_y}\dot{\phi}_i\dot{\psi}_i + K_{p,2}^i\tilde{\theta}_i + K_{d,2}^i\dot{\tilde{\theta}}_i) \\ \frac{I_z}{L}(-K_3^i\dot{\psi}_i - \frac{I_x-I_y}{I_z}\dot{\phi}_i\dot{\theta}_i + K_{p,3}^i\tilde{\psi}_i + K_{d,3}^i\dot{\tilde{\psi}}_i) \end{bmatrix}, \tag{2.16}$$

where $K_{p,j}^i$ and $K_{d,j}^i$, $i = 1,2,..,N,\ j = 1,2,3$ are the PD control gains, and $\tilde{\phi}_i$, $\tilde{\theta}_i$ and $\tilde{\psi}_i$ are the difference between the desired and actual roll, pitch and yaw angles, respectively, defined as

$$\begin{bmatrix} \tilde{\phi}_i \\ \tilde{\theta}_i \\ \tilde{\psi}_i \end{bmatrix} = \begin{bmatrix} \phi_i^{des} - \phi_i \\ \theta_i^{des} - \theta_i \\ \psi_i^{des} - \psi_i \end{bmatrix}. \tag{2.17}$$

2.6 Geometric Control

Although we mainly use the PD controller in our simulation results, we have also implemented a geometric controller in the simulations.

2.6.1 Related Works

Geometric tracking control of a quadcopter UAV on $SE(3)$ is developed in [32], which uses the $SE(3)$ instead of the Euler angles. It's a nonlinear tracking controller which is similar to PD controller used in the literature for small angle approximations (see [33]) with two new important features. First, the orientation error is not based on Euler angles, which contain singularities. Second, the desired force is projected onto the actual z

body axis. Results shown in [32,34] verifies that this type of controller can be successfully applied for large attitude tracking error.

2.6.2 Geometric Control of Cooperative UAVs

Geometric control developed in [32] is extended to quadrotor with a cable suspended payload in [35] and the system is proved to be differentially flat. Similar paradigm was applied to multiple quadcopters transporting a cable-suspended regid body in [11, 18]. In a similar paper [7] developed a full dynamic model for cooperative manipulation of payloads (both point mass and rigid model) suspended by cables from multiple quadcopters for a hybrid model when the cables are tout and slack and utilized the differential flatness property to plan nominal trajectories in flat space. Recently [11] developed a controller based on [18] such that the payload asymptotically follows a given desired trajectory for its position and attitude in presence of uncertainties.

In [11] the force and moment acting on the payload are controlled by using a PID controller. These forces and moments are used to balance the tension along and normal to the cables. The force and moment controller for the payload is based on the error dynamics between the desired and actual trajectory. Similarly, the forces and moments for each quadrotor UAV is controlled using a PID controller.

Rewrite (2.12)

$$\sum_{i=1}^{N} f_i^d = M_c g e_3. \tag{2.18}$$

Also rewrite (3.1)

$$F_i = -\Gamma_i(\dot{x}_i - v^d) + f_i^d + m_i g e_3, \tag{2.19}$$

where $\Gamma_i = \Gamma_i^T > 0$

$$F_i^T F_i = (T_i R_i(\Phi_i) e_3)^T T_i R_i(\Phi_i) e_3 \Longrightarrow T_i^d = \sqrt{F_i^T F_i}. \tag{2.20}$$

The third body axis is always parallel to the control force F_i.

$$b_{3,i}^d = \frac{F_i}{||F_i||} \tag{2.21}$$

The second body fixed axis $b_{1,i}^d$ is chosen such that it can form an orthonormal frame with $b_{3,i}^d$ Assuming $b_{1,i}^d$ is not parallel to $b_{3,i}^d$, the desired rotation matrix can be computed as

$$R_i^d = \begin{bmatrix} b_{2,i}^d \times b_{3,i}^d & b_{2,i}^d, & b_{3,i}^d \end{bmatrix} \tag{2.22}$$

The error vectors for attitude and angular velocity for i-th quadrotor is defined as:

$$\begin{aligned} e_{R,i} &= \frac{1}{2}({R_i^d}^T R_i - R_i^T R_i^d)^\vee, \\ e_{\Omega,i} &= \Omega_i - R_i^T R_i^d \Omega_i^d. \end{aligned} \tag{2.23}$$

where *vee map* $\vee : \mathsf{so}(3) \to \mathbb{R}^3$.

We assume that the desired trajectory is given by constant velocity for each agent as

$$\dot{x}^d = v^d = \begin{bmatrix} v_x^d & v_y^d & v_z^d \end{bmatrix}, \quad \ddot{x}^d = a^d = 0, \quad \dot{a}^d = 0. \tag{2.24}$$

The kinematic equation for each quadrotor can be written as

$$\dot{R}_i = R_i \hat{\Omega}_i, \tag{2.25}$$

where $R_i \in SO(3)$ is the rotation matrix between the body frame $O_{c,i}$ and the inertial frame Σ_i, $\Omega_i \in \mathbb{R}^3$ is the angular velocity of i-th quadrotor in body-fixed frame and the operator $\widehat{\ }$ converts a given vector $\omega = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^T$, into a skew symmetric matrix as:

$$\widehat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \tag{2.26}$$

The angular dynamics of the i-th quadrotor is given by

$$J_i \dot{\Omega}_i = M_i - \Omega_i \times J_i \Omega_i, \tag{2.27}$$

where J_i is the inertia matrix and $M_i \in \mathbb{R}^3$ is the moment generated by the i-quadcopter. Based on [11], M_i is designed as

$$M_i = -\frac{k_R}{\epsilon^2} e_{R,i} - \frac{K_\Omega}{\epsilon} e_{\Omega,i} + \Omega_i \times J_i \Omega_i - J_i(\hat{\Omega}_i R_i^T R_i^d \Omega_i^d - R_i^T R_i^d \dot{\Omega}_i^d), \tag{2.28}$$

where ϵ, k_R, k_Ω are positive constants. $\Omega_i^{des} \in \mathbb{R}^3$ is obtained from the attitude kinematics equation

$$\Omega_i^d = ({R_i^d}^T \dot{R}_i^d)^\vee \tag{2.29}$$

2.6.3 Implementation

Rewrite the attitude kinematics as

$$\begin{aligned} \dot{R}_i^d &= R_i^d \hat{\Omega}_i^d, \\ \dot{R}_i^d &= (\hat{\Omega}_i^d)^I R_i^d = (\Omega_i^d)^I \times R_i^d \\ \dot{R}_i^d &= R_i^d \Omega_i^d \times R_i^d \end{aligned} \tag{2.30}$$

Time derivative of (2.30) yields,

$$\begin{aligned} \ddot{R}_i^d &= \frac{d}{dt}(R_i^d \hat{\Omega}_i^d) \\ \ddot{R}_i^d &= \dot{R}_i^d \hat{\Omega}_i^d + R_i^d \dot{\hat{\Omega}}_i^d \\ \ddot{R}_i^d &= R_i^d (\hat{\Omega}_i^d)^2 + R_i^d \dot{\hat{\Omega}}_i^d \\ \implies \dot{\hat{\Omega}}_i^d &= (R_i^d)^T \ddot{R}_i^d - (\hat{\Omega}_i^d)^2 \\ \implies \dot{\Omega}_i^d &= \left((R_i^d)^T \ddot{R}_i^d - (\hat{\Omega}_i^d)^2\right)^\vee \end{aligned} \tag{2.31}$$

2.6.4 Desired Rotation matrix R_i^d

$$R_i^d = \begin{bmatrix} b_{2,i}^d \times b_{3,i}^d & b_{2,i}^d, & b_{3,i}^d. \end{bmatrix} \tag{2.32}$$

Define

$$b_{1,i}^d = \begin{bmatrix} \cos \pi t, & \sin \pi t, & 0 \end{bmatrix} \tag{2.33}$$

Rewrite (2.21)

$$b_{3,i}^d = \frac{F_i}{\|F_i\|} = \frac{-\Gamma_i(\dot{x}_i - v^d) + f_i^d + m_i g e_3}{\| -\Gamma_i(\dot{x}_i - v^d) + f_i^d + m_i g e_3\|} \tag{2.34}$$

$$b_{2,i}^d = \frac{b_{3,i}^d \times b_{1,i}^d}{||b_{3,i}^d \times b_{1,i}^d||} \tag{2.35}$$

Define

$$\beta = b_{3,i}^d \times b_{1,i}^d \tag{2.36}$$

$$\dot{\beta} = \dot{b}_{3,i}^d \times b_{1,i}^d + b_{3,i}^d \times \dot{b}_{1,i}^d \tag{2.37}$$

$$\ddot{\beta} = \ddot{b}_{3,i}^d \times b_{1,i}^d + 2(\dot{b}_{3,i}^d \times \dot{b}_{1,i}^d) + b_{3,i}^d \times \ddot{b}_{1,i}^d \tag{2.38}$$

Rewrite the rotation matrix as

$$R_i^d = \left[b_{2,i}^d \times b_{3,i}^d, \quad \frac{\beta}{\|\beta\|}, \quad b_{3,i}^d \right] \tag{2.39}$$

2.6.5 Desired rate of Rotation $\dot{R}_i^d$

Derivative of (2.32) yields

$$\dot{R}_i^d = \left[\dot{b}_{2,i}^d \times b_{3,i}^d + b_{2,i}^d \times \dot{b}_{3,i}^d, \quad \dot{b}_{2,i}^d, \quad \dot{b}_{3,i}^d \right] \tag{2.40}$$

Derivative of (2.34) yields

$$\begin{aligned} \dot{b}_{3,i}^d &= \frac{d}{dt}\left(\frac{F_i}{\|F_i\|}\right) \\ \dot{b}_{3,i}^d &= \frac{\dot{F}_i\|F_i\| - F_i\frac{F_i^T \cdot \dot{F}_i}{\|F_i\|}}{\|F_i\|^2} \\ \dot{b}_{3,i}^d &= \frac{\dot{F}_i}{\|F_i\|} - \frac{F_i F_i^T \dot{F}_i}{\|F_i\|^3} \end{aligned} \tag{2.41}$$

$$\dot{b}_{1,i}^d = \left[-\pi \sin \pi t, \quad \pi \cos \pi t, \quad 0 \right] \tag{2.42}$$

Calculate $\dot{b}_{2,i}^d$

$$
\begin{aligned}
\dot{b}^d_{2,i} &= \frac{d}{dt}\frac{b^d_{3,i}\times b^d_{1,i}}{||b^d_{3,i}\times b^d_{1,i}||} \\
&= \frac{\frac{d}{dt}(b^d_{3,i}\times b^d_{1,i})\|b^d_{3,i}\times b^d_{1,i}\| - (b^d_{3,i}\times b^d_{1,i})\frac{d}{dt}(\|b^d_{3,i}\times b^d_{1,i}\|)}{\|b^d_{3,i}\times b^d_{1,i}\|^2} \\
&= \frac{\frac{d}{dt}(b^d_{3,i}\times b^d_{1,i})}{\|b^d_{3,i}\times b^d_{1,i}\|} - \frac{(b^d_{3,i}\times b^d_{1,i})(b^d_{3,i}\times b^d_{1,i})^T\frac{dt}{dt}(b^d_{3,i}\times b^d_{1,i})}{\|b^d_{3,i}\times b^d_{1,i}\|^3}
\end{aligned}
\tag{2.43}
$$

where

$$
\frac{d}{dt}(b^d_{3,i}\times b^d_{1,i}) = (\dot{b}^d_{3,i}\times b^d_{1,i}) + (b^d_{3,i}\times \dot{b}^d_{1,i}) \tag{2.44}
$$

Rewrite (2.45) as

$$
\begin{aligned}
\dot{b}^d_{2,i} &= \frac{(\dot{b}^d_{3,i}\times b^d_{1,i}) + (b^d_{3,i}\times \dot{b}^d_{1,i})}{\|b^d_{3,i}\times b^d_{1,i}\|} - \frac{(b^d_{3,i}\times b^d_{1,i})(b^d_{3,i}\times b^d_{1,i})^T\left((\dot{b}^d_{3,i}\times b^d_{1,i}) + (b^d_{3,i}\times \dot{b}^d_{1,i})\right)}{\|b^d_{3,i}\times b^d_{1,i}\|^3} \\
\dot{b}^d_{2,i} &= \frac{\dot{\beta}}{\|\beta\|} - \frac{\beta\beta^T\dot{\beta}}{\|\beta\|^3}
\end{aligned}
\tag{2.45}
$$

Rewrite $\dot{R}^d_i$ as

$$
\dot{R}^d_i = \left[\dot{\beta},\quad \frac{\dot{\beta}}{\|\beta\|} - \frac{\beta\beta^T\dot{\beta}}{\|\beta\|^3},\quad \frac{\dot{F}_i}{\|F_i\|} - \frac{F_iF_i^T\dot{F}_i}{\|F_i\|^3}\right] \tag{2.46}
$$

2.6.6 $\ddot{R}^d_i$

Derivative of (2.40) yields

$$
\ddot{R}^d_i = \left[\ddot{b}^d_{2,i}\times b^d_{3,i} + 2(\dot{b}^d_{2,i}\times \dot{b}^d_{3,i}) + b^d_{2,i}\times \ddot{b}^d_{3,i},\quad \ddot{b}^d_{2,i},\quad \ddot{b}^d_{3,i}\right] \tag{2.47}
$$

Differentiate (2.41)

$$
\begin{aligned}
\ddot{b}^d_{3,i} &= \frac{d}{dt}\left\{\frac{\dot{F}_i\|F_i\|^2}{\|F_i\|^3} - \frac{F_iF_i^T\dot{F}_i}{\|F_i\|^3}\right\} \\
&= \frac{d}{dt}\left\{\frac{\dot{F}_i}{\|F_i\|} - \frac{F_iF_i^T\dot{F}_i}{\|F_i\|^3}\right\} \\
&= \frac{\ddot{F}_i\|F_i\| - \dot{F}_i\|\dot{F}_i\|}{\|F_i\|^2} - \frac{\frac{dt}{dt}(F_IF_i^T\dot{F}_i)\|F_i\|^3 - (F_iF_i^T\dot{F}_i)\frac{d}{dt}(\|F_i\|^3)}{\|F_i\|^6} \\
&= \frac{\ddot{F}_i}{\|F_i\|} - \frac{\dot{F}_iF_i^T\dot{F}_i}{\|F_i\|^3} - \frac{(\dot{F}_iF_i^T\dot{F}_i) + (F_i\dot{F}_i^T\dot{F}_i) + (F_iF_i^T\ddot{F}_i)}{\|F_i\|^3} + \frac{3(F_iF_i^T\dot{F}_i)F_i^T\dot{F}_i}{\|F_i\|^5} \\
&= \frac{\ddot{F}_i}{\|F_i\|} - \frac{2(\dot{F}_iF_i^T\dot{F}_i) + (F_i\dot{F}_i^T\dot{F}_i) + (F_iF_i^T\ddot{F}_i)}{\|F_i\|^3} + \frac{3(F_iF_i^T\dot{F}_i)F_i^T\dot{F}_i}{\|F_i\|^5}
\end{aligned}
\tag{2.48}
$$

Compute the second derivative of $b_{2,i}^d$ as

$$\begin{aligned}
\ddot{b}_{2,i}^d &= \frac{d}{dt}\left\{\frac{(\dot{b}_{3,i}^d \times b_{1,i}^d) + (b_{3,i}^d \times \dot{b}_{1,i}^d)}{\|b_{3,i}^d \times b_{1,i}^d\|} - \frac{(b_{3,i}^d \times b_{1,i}^d)(b_{3,i}^d \times b_{1,i}^d)^T\left((\dot{b}_{3,i}^d \times b_{1,i}^d) + (b_{3,i}^d \times \dot{b}_{1,i}^d)\right)}{\|b_{3,i}^d \times b_{1,i}^d\|^3}\right\} \\
\ddot{b}_{2,i}^d &= \frac{d}{dt}\left\{\frac{\dot{\beta}}{\|\beta\|} - \frac{\beta\beta^T\dot{\beta}}{\|\beta\|^3}\right\} \\
&= \frac{\ddot{\beta}\|\beta\| - \dot{\beta}\|\dot{\beta}\|}{\|\beta\|^2} - \frac{\frac{dt}{dt}(\beta\beta^T\dot{\beta})\|\beta\|^3 - (\beta\beta^T\dot{\beta})\frac{d}{dt}(\|\beta\|^3)}{\|\beta\|^6} \\
&= \frac{\ddot{\beta}}{\|\beta\|} - \frac{\dot{\beta}\beta^T\dot{\beta}}{\|\beta\|^3} - \frac{(\dot{\beta}\beta^T\dot{\beta}) + (\beta\dot{\beta}^T\dot{\beta}) + (\beta\beta^T\ddot{\beta})}{\|\beta\|^3} + \frac{3(\beta\beta^T\dot{\beta})\beta^T\dot{\beta}}{\|\beta\|^5} \\
&= \frac{\ddot{\beta}}{\|\beta\|} - \frac{2(\dot{\beta}\beta^T\dot{\beta}) + (\beta\dot{\beta}^T\dot{\beta}) + (\beta\beta^T\ddot{\beta})}{\|\beta\|^3} + \frac{3(\beta\beta^T\dot{\beta})\beta^T\dot{\beta}}{\|\beta\|^5}
\end{aligned} \tag{2.49}$$

Rewrite $\ddot{R}_i^d$ as

$$\ddot{R}_i^d = \left[\ddot{\beta},\quad \frac{\ddot{\beta}}{\|\beta\|} - \frac{2(\dot{\beta}\beta^T\dot{\beta})+(\beta\dot{\beta}^T\dot{\beta})+(\beta\beta^T\ddot{\beta})}{\|\beta\|^3} + \frac{3(\beta\beta^T\dot{\beta})\beta^T\dot{\beta}}{\|\beta\|^5},\quad \frac{\ddot{F}_i}{\|F_i\|} - \frac{2(\dot{F}_iF_i^T\dot{F}_i)+(F_i\dot{F}_i^T\dot{F}_i)+(F_iF_i^T\ddot{F}_i)}{\|F_i\|^3} + \frac{3(F_iF_i^T\dot{F}_i)F_i^T\dot{F}_i}{\|F_i\|^5}\right] \tag{2.50}$$

CHAPTER 3

NOMINAL DESIGN

In this chapter, we review the main results in [36] and extend it to address cooperative transportation of a payload in 3D.

3.1 Force Control with a Payload of Known Mass

We assume that the orientation of the load is controlled separately and consider only the translational part of the motion in this chapter. We first choose F_i as

$$F_i = -\Gamma_i(\dot{x}_i - v^d) + f_i^d + m_i g e_3, \tag{3.1}$$

where $\Gamma_i = \Gamma_i^T > 0$, which component wise becomes

$$\begin{bmatrix} F_i^x \\ F_i^y \\ F_i^z \end{bmatrix} = -\Gamma_i \begin{pmatrix} \dot{x}_{i,x} - v_x^d \\ \dot{x}_{i,y} - v_y^d \\ \dot{x}_{i,z} - v_z^d \end{pmatrix} + \begin{pmatrix} f_{i,x}^d \\ f_{i,y}^d \\ \frac{1}{N} M_c g + m_i g \end{pmatrix}. \tag{3.2}$$

Note that in (3.2) we have chosen $f_{i,z}^d = \frac{1}{N} M_c g$ and the x and y components of f_i^d are pre-designed to satisfy the following constraints

$$\sum_{i=1}^{N} f_{i,x}^d = 0 \quad \text{and} \quad \sum_{i=1}^{N} f_{i,y}^d = 0. \tag{3.3}$$

3.1.1 Computation of T_i^{des} and Φ_i^{des}

For a coplanar quadcopter and away from singularities, we next establish the following map (for any yaw ψ_i)

$$\begin{bmatrix} F_i^x \\ F_i^y \\ F_i^z \end{bmatrix} \rightarrow \begin{bmatrix} \phi_i^{des}(\psi_i) \\ \theta_i^{des}(\psi_i) \\ T_i^{des} \end{bmatrix}. \tag{3.4}$$

Starting from

$$T_i^{des} R_i(\Phi_i^{des}) e_3 = F_i, \tag{3.5}$$

and premultiplying by $R_i^T(\Phi_i^{des})$, we get

$$T_i^{des} e_3 = R_i^T(\Phi_i^{des}) \begin{bmatrix} F_i^x \\ F_i^y \\ F_i^z \end{bmatrix},$$

which are three scalar equations. From the first we get

$$\theta_i^{des} = atan\left(\frac{F_i^x c\psi_i^{des} + F_i^y s\psi_i^{des}}{F_i^z}\right), \tag{3.6}$$

and from the other two equations

$$\phi_i^{des} = asin\left(\frac{F_i^x s\psi_i^{des} - F_i^y c\psi_i^{des}}{\sqrt{F_i^T F_i}}\right), \tag{3.7}$$

where in (3.7) we have used (3.5) to obtain the relationship

$$T_i^{des} = \sqrt{F_i^T F_i}. \tag{3.8}$$

Remark 3.1 *Define the velocity error vector in* (3.1) *as*

$$\xi_i = \dot{x}_i - v^d \tag{3.9}$$

and recall from (3.2) *that* $v^d = \begin{bmatrix} v_x^d & v_y^d & v_z^d \end{bmatrix}^T$. *Suppose that* $|\xi_i| \leq \eta_i^m$ *and* $|f_{i,y}^d| + |f_{i,x}^d| \leq f_c^m$ *for some positive bounds* η_i^m *and* f_c^m, *and fix* Γ_i *as a positive definite and diagonal*

matrix. Then, it is straightforward to see that T_i^{des} can be upper bounded as follows

$$\begin{aligned} |T_i^{des}| &= \sqrt{F_i^T F_i} \\ &\leq ||\Gamma_i||\,|\xi_i| + |f_{i,x}^d| + |f_{i,y}^d| + \frac{1}{N} M_c g + m_i g \\ &\leq ||\Gamma_i|| \eta_i^m + f_c^m + \frac{1}{N} M_c g + m_i g. \end{aligned} \tag{3.10}$$

3.1.2 Stability Analysis

The dynamics (2.8) with the proposed control law (3.1) takes the following form:

$$m_i \ddot{x}_i = -\Gamma_i \xi_i + f_i^d - f_i. \tag{3.11}$$

Theorem 3.1 *The decentralized control law* (3.1) *ensures that the closed-loop equilibrium of* (3.11)*, given by,*

$$\mathscr{E} = \{(\dot{x}_i, \dot{x}_c, f_i) | \dot{x}_i = v_i^d, \quad \dot{x}_c = v_i^d, \quad \textit{and} \quad f_i = f_i^d\}, \tag{3.12}$$

is asymptotically stable.

Proof. Consider the following energy-motivated positive definite candidate Lyapunov function

$$\begin{aligned} V = &\sum_{i=1}^{N} \left[P_i(z_i) - P_i(z_i^d) - (f_i^d)^T (z_i - z_i^d) \right] \\ &+ \frac{1}{2} \left(\sum_{i=1}^{N} \xi_i^T m_i \xi_i + \xi_c^T M_c \xi_c \right). \end{aligned} \tag{3.13}$$

From (2.2) and (2.3), the kinematics of z_i is given by

$$\dot{z}_i = \dot{x}_i - \dot{a}_i = \dot{x}_i - \dot{x}_c. \tag{3.14}$$

Define $\xi_c = \dot{x}_c - v^d$. The time derivative of V yields

$$\dot{V} = \sum_{i=1}^{N} (f_i - f_i^d)^T \dot{z}_i + \sum_{i=1}^{N} \xi_i^T m_i \ddot{x}_i + \xi_c^T M_c \ddot{x}_c. \tag{3.15}$$

We rewrite (3.15) from (2.7), (2.8), (2.12) and (3.14) as:

$$\begin{aligned}
\dot{V} &= \sum_{i=1}^{N} (f_i - f_i^d)^T (\dot{x}_i - \dot{x}_c) + \sum_{i=1}^{N} \xi_i^T (F_i - f_i) \\
&\quad + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g_3, \\
&= \sum_{i=1}^{N} (f_i - f_i^d)^T (\xi_i - \xi_c) + \sum_{i=1}^{N} \xi_i^T (-\Gamma_i \xi_i + f_i^d - f_i) \\
&\quad + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g e_3, \\
&= -\sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i \leq 0
\end{aligned} \tag{3.16}$$

which implies the stability of the equilibrium $\mathscr{E}$.

To conclude the asymptotic stability of the system, we apply the LaSalle invariance principle [37] to investigate the largest invariant set M. On M, $\xi_i = 0$ which implies from (3.9) that $\dot{x}_i = v^d$. Since $\dot{x}_i$ is constant, $\ddot{x}_i = 0$, which implies from (3.11) that $f_i^d = f_i$. Since f_i is constant and equal to f_i^d, z_i is constant, which leads to $\dot{x}_c = \dot{x}_i = v^d$ and $\ddot{x}_c = 0$. ■

Remark 3.2 *Since ξ_i is bounded, the existence of a bounded thrust T_i is always guaranteed from the result of Remark 3.1.*

Remark 3.3 *The proof of proposition 3.1 is similar to [38, Chapter 8] except that the dynamics are extended to incorporate the gravity of the agents and the payload.*

3.2 Simulation Results

3.2.1 Simulation Environment

In this section, we present simulation results for two quadcopters with a rigid extension transporting a load. The mass and inertia of each quadcopter are $m_i = 0.75$ kg and $I_i = diag\{0.0820, 0.0845, 0.1377\}\ kgm^2$ respectively. The mass of the payload is $M_c = 1.5$ kg with a radius of 15 cm. We use a linear spring-force model to compute the contact

force f_i in the simulation,

$$f_i = kz_i, \tag{3.17}$$

where k is the spring constant of the payload. If the contact between agents and payload is elastic at a constant pressure, the contact force model in (3.17) is equivalent to the models presented in [28,29]. For all of the results presented in this section we set $\psi_i^{des} = 0$ and $k = 2.5 \times 10^4$ N/m.

3.2.2 Known Payload with a Predesigned v^d with a PD Controller

We choose

$$v^d(t) = (1 - e^{-t}) \begin{bmatrix} 5.0 & 0 & 0 \end{bmatrix}^T,$$
$$f_1^d(t) = (1 - e^{-t}) \begin{bmatrix} 0 & 2.0 & 0.5 M_c g \end{bmatrix}^T,$$
$$f_2^d(t) = (1 - e^{-t}) \begin{bmatrix} 0 & -2.0 & 0.5 M_c g \end{bmatrix}^T,$$
$$\Gamma_i = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}.$$

Note that $v^d(t)$, $f_1^d(t)$, $f_2^d(t)$ converge to constants exponentially. The parameters for the gains for the attitude controller are given in Table 3.1.

K_1^i, K_2^i	K_3	K_{p1}^i, K_{p2}^i	K_{p3}^i	K_{d1}^i, K_{d2}^i	K_{d3}^i
21.93	18	4.65	3.76	0.19	0.15

Table 3.1: Attitude controller gains.

Fig. 3.1 and 3.2 show that the attitude controller implemented in Section 3.2 is successful in tracking the desired attitude angles, ϕ_i^{des} and θ_i^{des} generated from the force controller (3.6) and (3.7). It can be observed that the actual pitch (θ_i) and roll (ϕ_i) angles for each quadcopter converge to steady states within the first 6 seconds of the simulation. Fig. 3.3 shows that the desired thrust force (T_i) converges to a constant value within

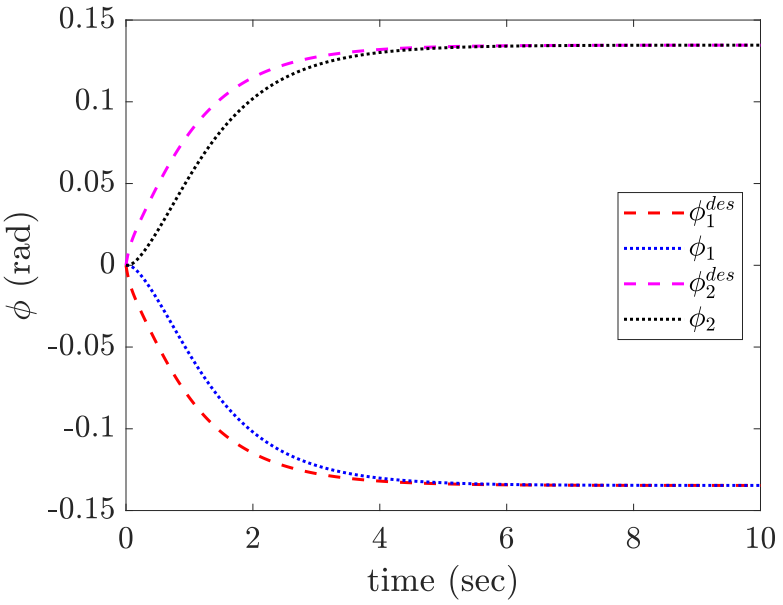


Figure 3.1: Desired and actual roll angles (ϕ) for both agents.

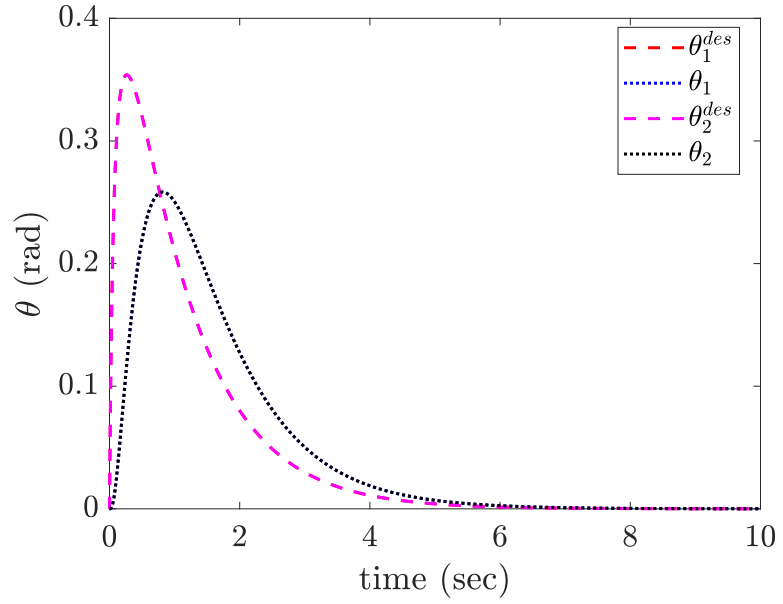


Figure 3.2: Desired and actual pitch angles (θ) for both agents.

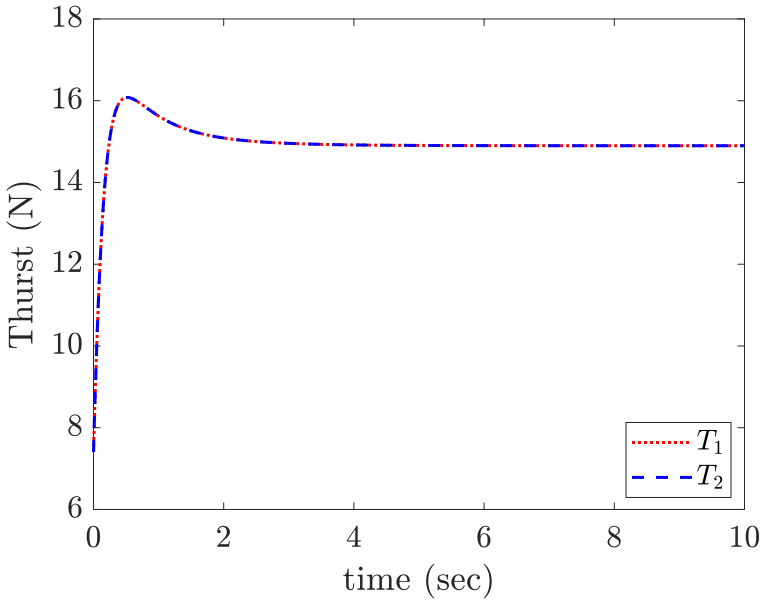


Figure 3.3: The thrusts for both agents are bounded and converge to 14.7 N.

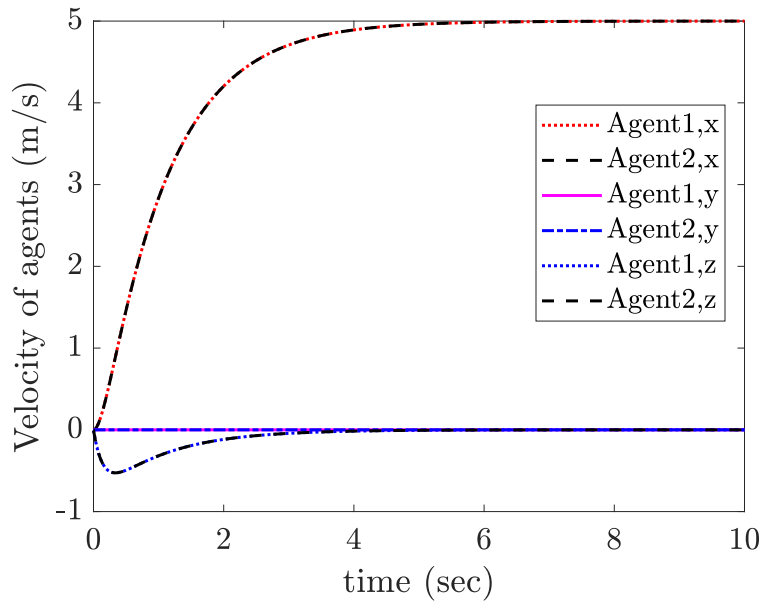


Figure 3.4: Linear velocities for both agents. The x component of the velocity converges to 5.0 m/s and the rest converge to zero.

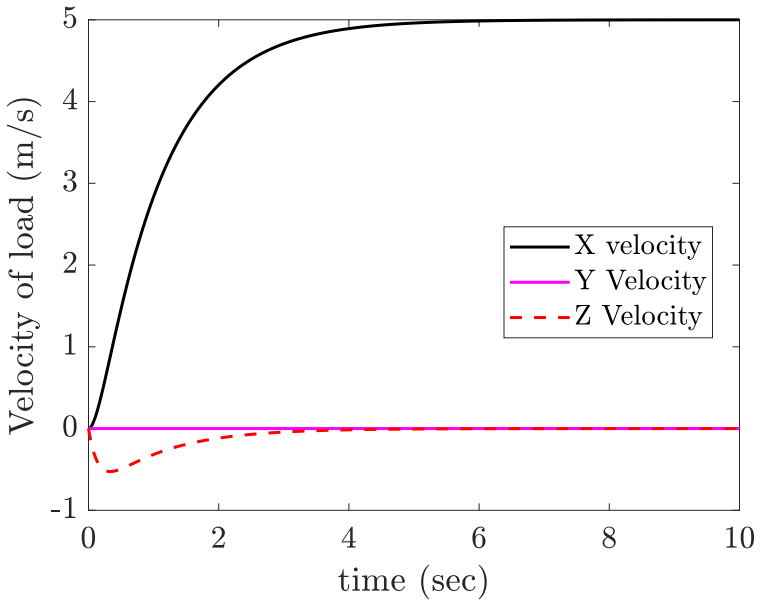


Figure 3.5: Linear velocity for the payload. The x component of the velocity converges to 5.0 m/s and the rest converge to zero.

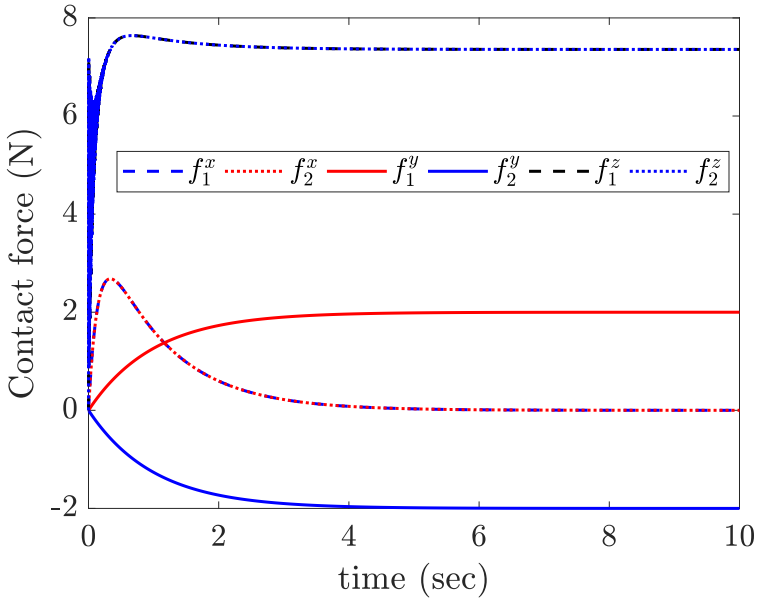


Figure 3.6: Contact forces for both AMs in all 3 direction. The forces in the x direction converge to zero. In the y-direction, they converge to 2.0 N and -2.0 N, respectively. In the z direction, the sum of the force is equal to the weight of the payload.

first 2 seconds and stays constant throughout the simulation. We observe from Fig. 3.4 and 3.5 that the velocities of the agents and the payload converge to v^d. We see from Fig. 3.6 that the contact forces are regulated to the setpoints f_i^d.

3.2.3 Known Payload with a Predesigned v^d with a Geometric Controller

Simulation Parameters:

$$b_i^d = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$$v^d = \begin{bmatrix} 2.0 & 0 & 0 \end{bmatrix}$$

$$f_i^d = \begin{bmatrix} 0 & \pm 2.0 & 0.5 M_c g \end{bmatrix}$$

Gains: $k_R = 1.5$, $\epsilon = 1.0$, $k_\Omega = 1.0$

$$\Gamma_i = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

spring constant $k = 2.5 \times 10^4$ N/m.

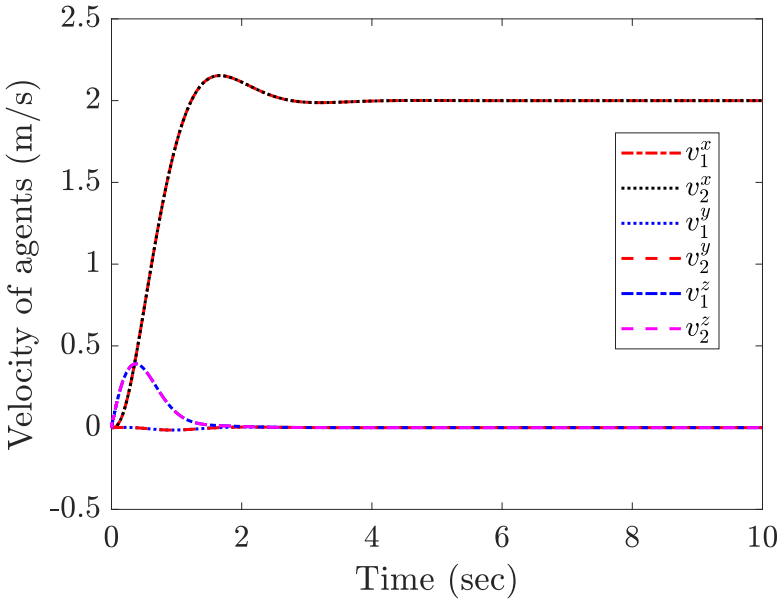


Figure 3.7: Linear velocities for both agents. The x component of the velocity converges to 5.0 m/s and the rest converge to zero.

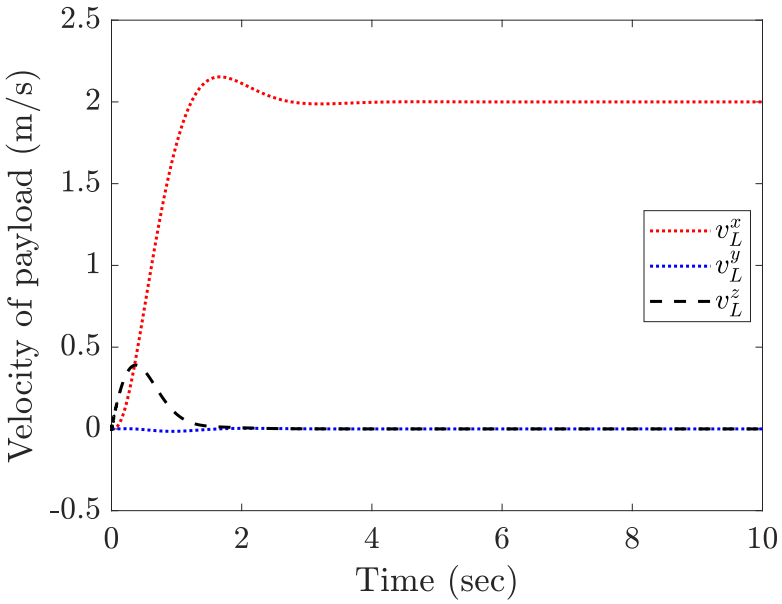


Figure 3.8: Linear velocity for the payload. The x component of the velocity converges to 5.0 m/s and the rest converge to zero.

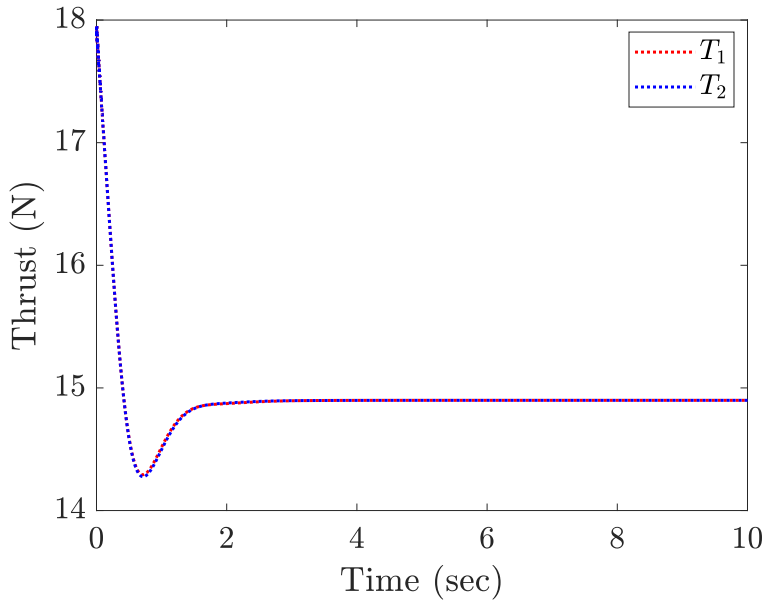


Figure 3.9: Desired thrust input to each quadrotor.

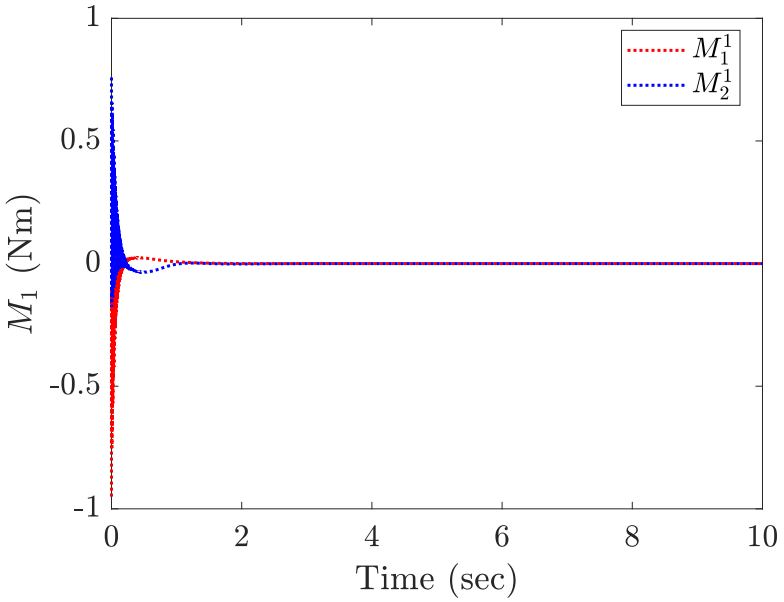


Figure 3.10: Control input M_1

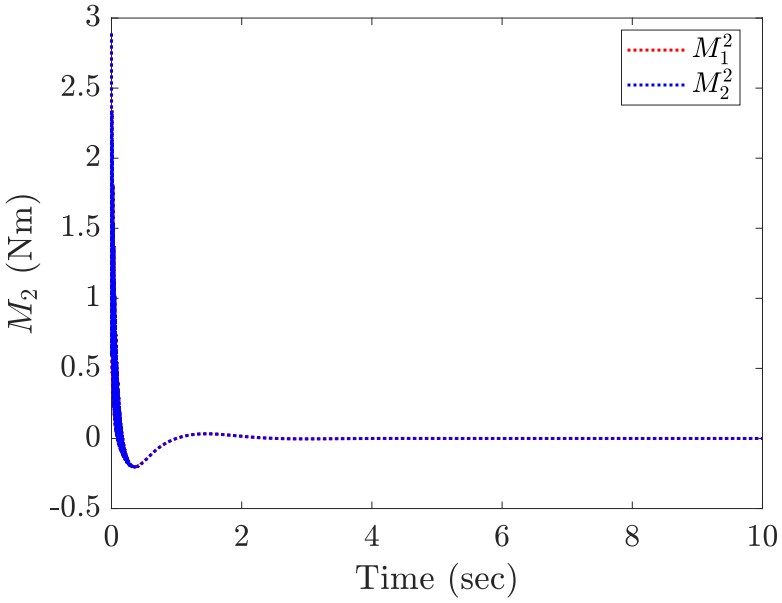


Figure 3.11: Control input M_2

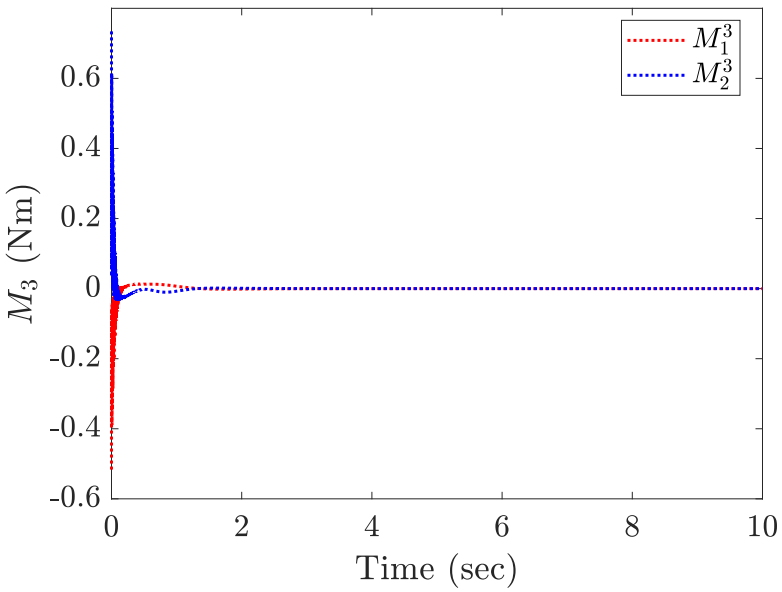


Figure 3.12: Control input M_3

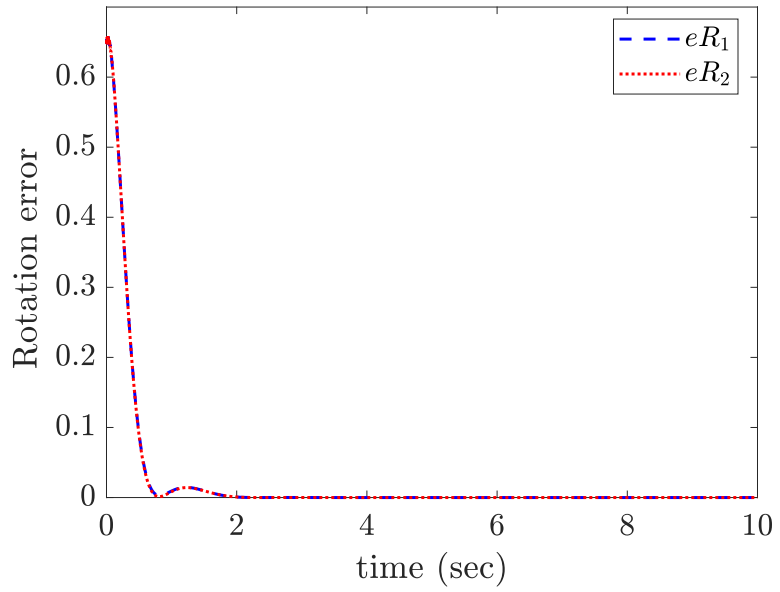


Figure 3.13: Attitude error of the quadrotors given by $\Psi_i = \frac{1}{2}\|R_i - R_I^d\|^2$

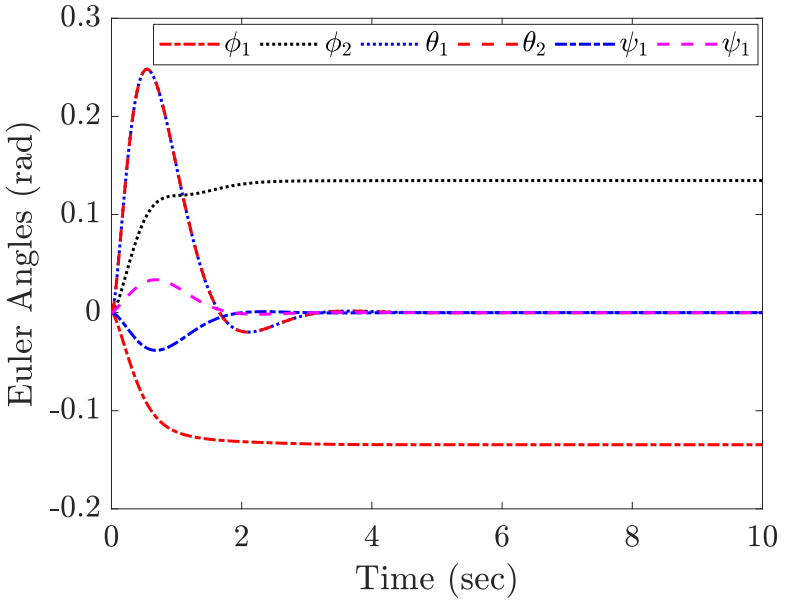


Figure 3.14: Euler Angles for each agent.

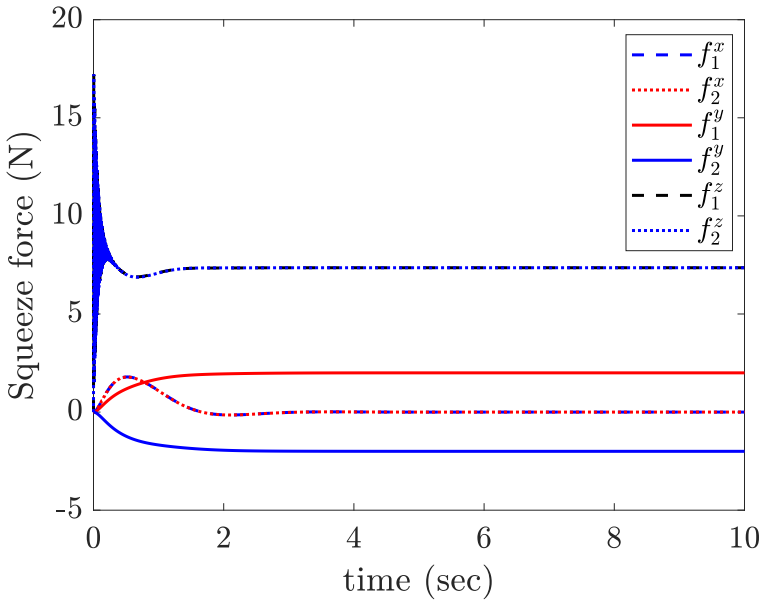


Figure 3.15: Contact forces.

CHAPTER 4

COOPERATIVE AERIAL MANIPULATION OF A PAYLOAD WITH UNKNOWN MASS

In Chapter 3, the mass of the payload is assumed available so that f_i^d in (2.12) can be specified. However, in practical scenarios, an exact knowledge of the mass of the payload may not be available. To address such a scenario, we develop an adaptive controller to estimate M_c.

In this chapter, we consider a group quadcopters with a single rigid link attached to it and develop an adaptive decentralized control law for transporting a payload with an unknown mass without explicit communication between the agents. The algorithm provides desired thrust and attitude angles required for each quadcopter to cooperatively transport a payload. It also guarantees that all the agents converge to a desired velocity and regulates the contact force. The sum of the estimates of the unknown mass from all the agents converge to the true mass. We model the contact force between the agents and the payload as a mass spring model. This assumption is valid when the vehicles are connected to the payload via elastic cables or when the payload is flexible or surrounded by elastic bumper materials.

This chapter also introduces a new force consensus algorithm to ensure that each quadcopter gets an equal share of the payload mass. Such an algorithm is important because without it, some of the quadcopters may provide more lift to the unknown payload than others, and therefore consume more power, which can lead to a shorter life span of the entire team. We provide extended stability analysis for the new algorithm and additional simulation results. To test the performance of the controllers for different conditions, we include simulation results for a time-varying reference velocity.

4.1 Force Control with a Payload of Unknown Mass

In Section 3.1, the mass of the payload is assumed available so that f_i^d in (2.12) can be specified. However, in practical scenarios, an exact knowledge of the mass of the payload may not be available. To address such a scenario, we develop an adaptive controller to estimate M_c.

We define $\widehat{M}_{c,i}$ as an estimate of mass of payload for each agent $i = 1, ..., N$ and propose the following update law for $\widehat{M}_{c,i}$

$$\dot{\widehat{M}}_{c,i} = \gamma_i (\dot{x}_i - v^d)^T e_3, \tag{4.1}$$

where $\gamma_i > 0$. Note that $\widehat{M}_{c,i}$ stops updating when each agent reaches the predesigned velocity v^d.

4.1.1 Computation of T_i^{des} and Φ_i^{des}

We modify the design in (3.1) as:

$$F_i = -\Gamma_i (\dot{x}_i - v^d) + \hat{f}_i^d + m_i g e_3, \tag{4.2}$$

and estimate the z-component of f_i^d as:

$$\hat{f}_{i,z}^d = \widehat{M}_{c,i} g. \tag{4.3}$$

The x and y components of $\hat{f}_i^d$ are the steady state values of the contact forces that are pre-designed to satisfy the following constraints

$$\sum_{i=1}^{N} \hat{f}_{i,x}^d = 0 \quad \text{and} \quad \sum_{i=1}^{N} \hat{f}_{i,y}^d = 0. \tag{4.4}$$

Given (4.2), the corresponding T_i and Φ_i are found based on (3.6)–(3.8).

4.1.2 Stability Analysis

The dynamics (2.8) with the proposed control law (4.1)– (4.3) takes the following form:

$$m_i \ddot{x}_i = -\Gamma_i \xi_i + \hat{f}_i^d - f_i. \tag{4.5}$$

Theorem 4.1 *The decentralized control law* (4.2) *ensures that the equilibrium of* (2.7), (4.1), *and* (4.5), *given by*

$$\begin{aligned}\mathscr{E}^* = \{(\dot{x}_i, f_i, v_i^d, \widehat{M}_{c,i}) | \dot{x}_i = v_i^d, \quad \dot{x}_c = v_i^d, \\ \sum_{i=1}^{N} \widehat{M}_{c,i} = M_c, \quad \textit{and} \quad \hat{f}_i^d = f_i\},\end{aligned} \tag{4.6}$$

is asymptotically stable.

Proof. Consider the following energy-motivated positive definite candidate Lyapunov function

$$\begin{aligned}V_1 = \sum_{i=1}^{N} \left[P_i(z_i) - P_i(z_i^d) - (f_i^d)^T (z_i - z_i^d) \right] \\ + \frac{1}{2} \left(\sum_{i=1}^{N} \xi_i^T m_i \xi_i + \xi_c^T M_c \xi_c \right).\end{aligned} \tag{4.7}$$

The time derivative of V_1 yields

$$\dot{V}_1 = \sum_{i=1}^{N} (f_i - f_i^d)^T \dot{z}_i + \sum_{i=1}^{N} \xi_i^T m_i \ddot{x}_i + \xi_c^T M_c \ddot{x}_c. \tag{4.8}$$

We rewrite (4.8) from (2.7), (2.8), (2.12), (3.14) and (4.2) as:

$$\begin{aligned}\dot{V}_1 &= \sum_{i=1}^{N} (f_i - f_i^d)^T (\dot{x}_i - \dot{x}_c) + \sum_{i=1}^{N} \xi_i^T (F_i - f_i) \\ &\quad + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g_3, \\ &= \sum_{i=1}^{N} (f_i - f_i^d)^T (\xi_i - \xi_c) + \sum_{i=1}^{N} \xi_i^T (-\Gamma_i \xi_i + \hat{f}_i^d - f_i) \\ &\quad + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g e_3, \\ &= \sum_{i=1}^{N} (f_i - f_i^d)^T \xi_i - \sum_{i=1}^{N} (f_i - f_i^d)^T \xi_c - \sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i \\ &\quad + \sum_{i=1}^{N} \xi_i^T (\hat{f}_i^d - f_i) + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g e_3, \\ &= - \sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^{N} \xi_i^T \tilde{f}_i^d,\end{aligned} \tag{4.9}$$

where $\tilde{f}_i^d = \hat{f}_i^d - f_i^d$, i.e.,

$$\tilde{f}_i^d = \begin{pmatrix} 0 \\ 0 \\ (\widehat{M}_{c,i} - M_{c,i})g \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \tilde{M}_{c,i}g \end{pmatrix} = \tilde{M}_{c,i} g e_3, \tag{4.10}$$

where $M_{c,i}$, $i = 1, \cdots, N$, are arbitrary constants that satisfy $\sum_{i=1}^N M_{c,i} = M_c$. From (4.9) and (4.10),

$$\dot{V}_1 = -\sum_{i=1}^N \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^N \xi_i^T \tilde{M}_{c,i} g e_3. \tag{4.11}$$

Consider another positive definite candidate Lyapunov function:

$$V_2 = \frac{1}{2}\sum_{i=1}^N \Lambda_i \big(\tilde{M}_{c,i}\big)^2. \tag{4.12}$$

The time derivative of V_2 yields

$$\dot{V}_2 = \sum_{i=1}^N \Lambda_i \tilde{M}_{c,i} \dot{\tilde{M}}_{c,i}. \tag{4.13}$$

Let $W = V_1 + V_2$. The time derivative of W is given by,

$$\begin{aligned} \dot{W} &= \dot{V}_1 + \dot{V}_2, \\ &= -\sum_{i=1}^N \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^N \xi_i^T \tilde{M}_{c,i} g e_3 + \sum_{i=1}^N \tilde{M}_{c,i} \dot{\tilde{M}}_{c,i}, \\ &= -\sum_{i=1}^N \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^N \xi_i^T \tilde{M}_{c,i} g e_3 - \sum_{i=1}^N \tilde{M}_{c,i} \Lambda_i \big(\gamma_i \xi_i^T e_3\big). \end{aligned} \tag{4.14}$$

Now from (4.1) and (4.14) and choosing $\Lambda_i \gamma_i = g$, we obtain

$$\dot{W} = -\sum_{i=1}^N \xi_i^T \Gamma_i \xi_i \leq 0, \tag{4.15}$$

which implies the stability of the equilibrium $\mathscr{E}^*$.

To conclude the asymptotic stability of the system, we apply the LaSalle invariance principle [37] to investigate the largest invariant set M. On M, $\xi_i = 0$ and (3.9) leads to $\dot{x}_i = v^d$. Since $\dot{x}_i$ is constant, $\ddot{x}_i = 0$, which implies from (4.25) that $\hat{f}_i^d = f_i$. Because $\dot{x}_i = v^d$, $\dot{\widehat{M}}_{c,i} = 0$, which means that $\widehat{M}_{c,i}$ is a constant. It follows from (4.3) that $\hat{f}_i^d$ is constant and equal to f_i. Since f_i is constant, z_i is constant, which leads to $\dot{x}_c = \dot{x}_i = v^d$ and $\ddot{x}_c = 0$. From (2.7), we get $\sum_{i=1}^N f_i = M_c g e_3$, which, together with $f_i = \hat{f}_i^d$, results in $\sum_{i=1}^N \hat{f}_i^d = M_c g$. Given (4.3), we conclude $\sum_{i=1}^N \widehat{M}_{c,i} = M_c$. ■

4.1.3 Simulation Results: Unknown Payload with a Predesigned v^d

In this case, we assume that the mass of the payload is unknown. We keep the other parameters to be the same as in the previous chapter and set $\widehat{M}_{c,1}(0) = 0.1$ kg, $\widehat{M}_{c,2}(0) = 0.4$ kg,

$$\hat{f}_1^d(t) = (1 - e^{-t}) \begin{bmatrix} 0 & 2.0 & \widehat{M}_{c,1} g \end{bmatrix}^T,$$

$$\hat{f}_2^d(t) = (1 - e^{-t}) \begin{bmatrix} 0 & -2.0 & \widehat{M}_{c,2} g \end{bmatrix}^T.$$

$$\Gamma_i = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 25 \end{bmatrix}.$$

Similar to the known mass case, Fig. 4.1 shows that actual roll angle (ϕ_i) converges to the steady state (ϕ_i^{des}) computed from the force controller (using (3.6) for Section. 6.3) within the first 6 seconds of simulation. Likewise, Fig. 4.2 shows that the actual pitch angle (θ_i) converges to its steady state (θ_i^{des}) for each quadrotor within first 6 seconds of the simulation. Fig. 4.3 shows that the desired thrust (T_i) converges to some constant value within first 4 seconds and stays constant throughout the simulation. We observe from Fig. 4.4 and 4.5 that the velocities of the agents and the payload converge to desired v^d. The contact forces is also regulated at the setpoints f_i^d as shown in Fig. 4.6. The update law from (4.1) successfully recovers the actual mass of the payload as seen in Fig. 4.7.

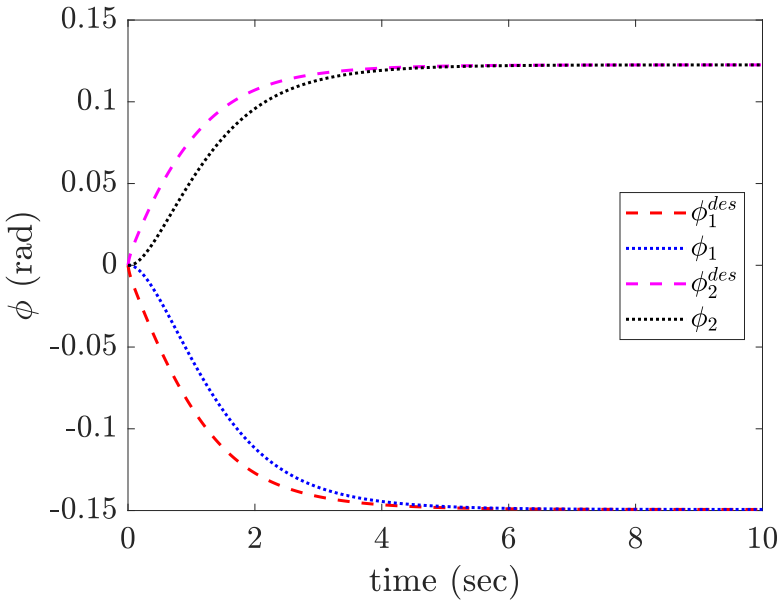


Figure 4.1: Desired and actual roll angles (ϕ) for both agents (unknown M_c case).

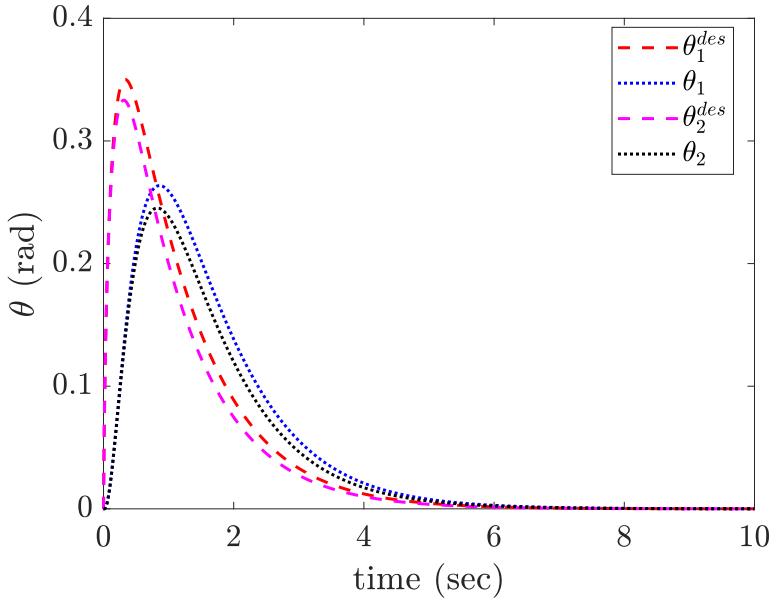


Figure 4.2: Desired and actual pitch angles (θ) for both agents (unknown M_c case).

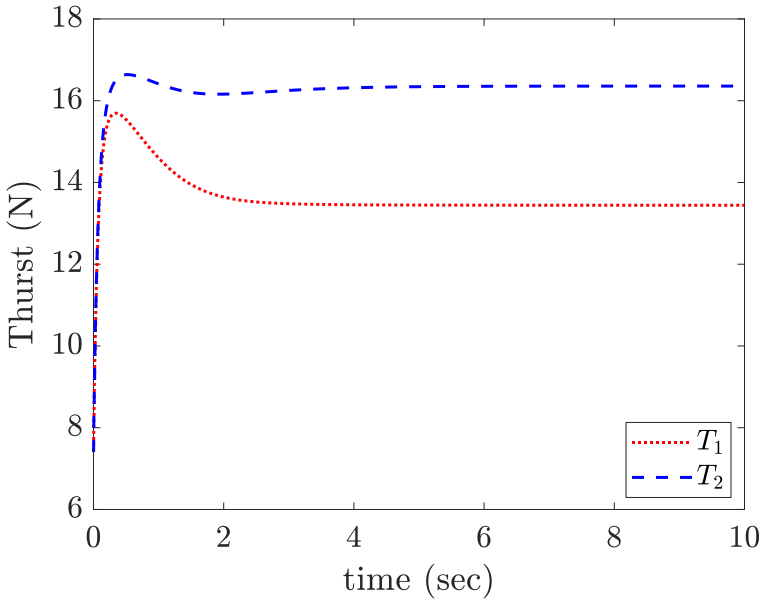


Figure 4.3: The thrusts for both agents converge to constant and remain bounded.

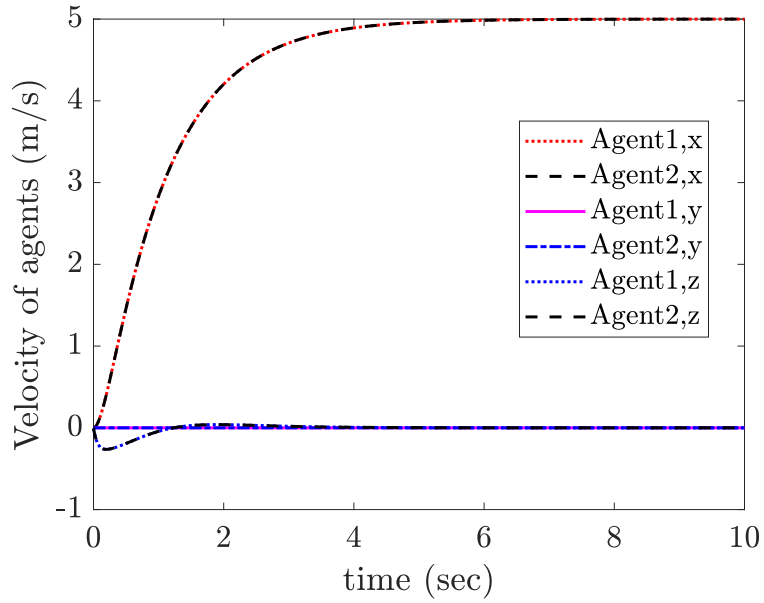


Figure 4.4: Linear velocities for both agents. The x component of the velocity converges to 5.0 m/s and the rest converge to zero.

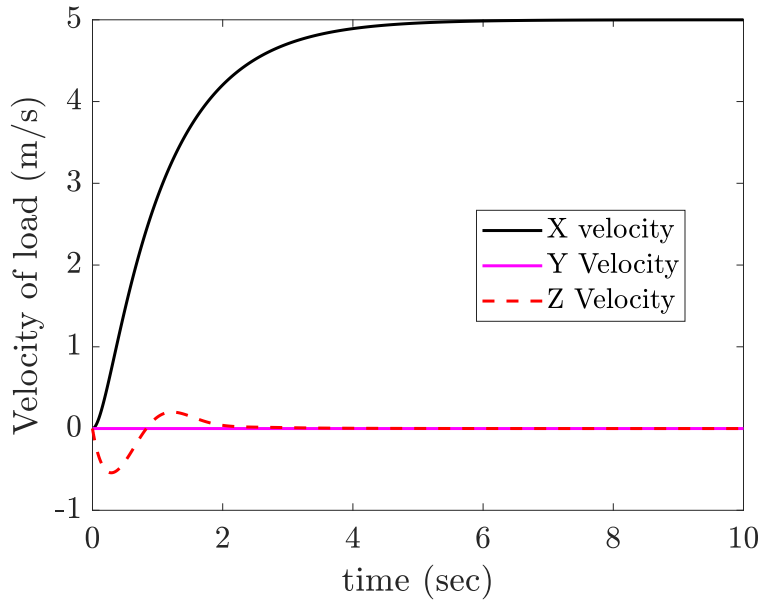


Figure 4.5: Linear velocities for both agents. The x component of the velocity converges to 5.0 m/s and the rest converge to zero.

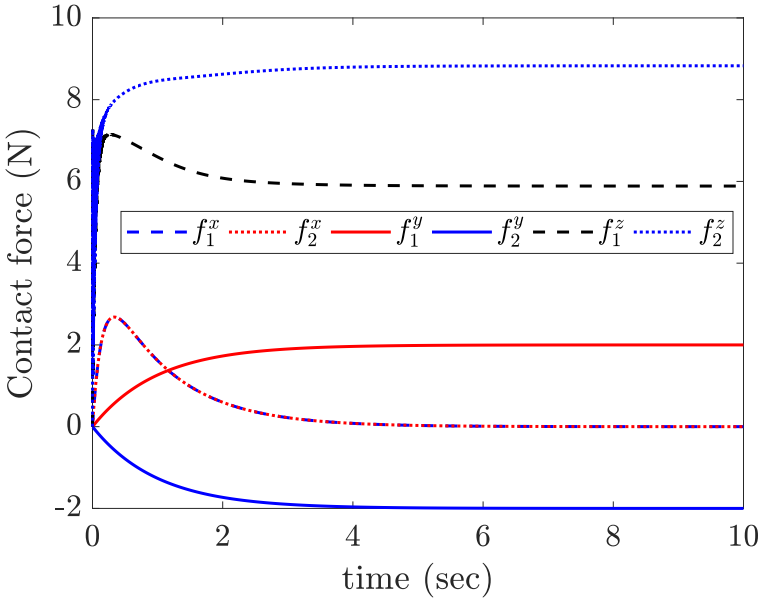


Figure 4.6: Contact forces for both AMs in all 3 direction. The forces in the x direction converge to zero. In the y-direction, they converge to 2.0 N and -2.0 N, respectively. In the z direction, the sum of the force is equal to the weight of the payload.

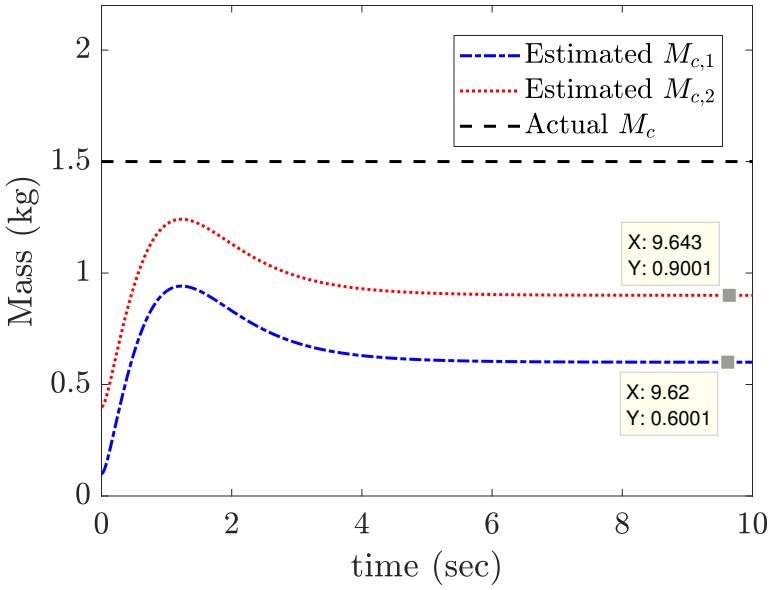


Figure 4.7: Estimation of the unknown mass. The sum of the individual estimates converges to the actual mass of the load which is 1.5 kg.

4.2 Lift Consensus During Cooperative Aerial Load Transport

In Section 4.1, we have developed an adaptive control to estimate the payload when the mass of the payload is not available to the controller. However, the estimates of the mass from the adaptive controller may not converge to the same value. In this section, we enhance the adaptive controller developed in Section 4.1 by incorporating a lift consensus term to ensure that the mass estimate of each agent converges to the same value, i.e., the total mass divided by the number of agents. In particular, we develop a lift consensus algorithm in the z-direction such that convergence of the estimates is achieved.

4.2.1 Adaptation Law

We denote by $\widehat{M}_{c,i}$ the estimate of the payload mass for agent i, $i = 1,...,N$. We propose to update $\widehat{M}_{c,i}$ based on the following equation

$$\dot{\widehat{M}}_{c,i} = \underbrace{-\lambda_i(\dot{x}_i - v^d)^T e_3}_{\widehat{\beta}_i} + \sum_{j\in\mathcal{N}_i} \lambda_i(\widehat{M}_{c,j} - \widehat{M}_{c,i}), \tag{4.16}$$

where $\lambda_i > 0$ and $\mathcal{N}_i$ is the set of neighboring agents that can share their estimates of the payload mass with agent i. Note that $\dot{\widehat{M}}_{c,i}$ becomes zero if each agent attains v^d in the e_3 direction and $\widehat{M}_{c,j} = \widehat{M}_{c,i}$, $\forall\, j \in \mathcal{N}_i$.

Let 1_N be the $N \times 1$ vectors of ones. Define

$$\widehat{M}_c = \begin{bmatrix} \widehat{M}_{c,1}, & \widehat{M}_{c,2}, & \widehat{M}_{c,3}, & \cdots & \widehat{M}_{c.N} \end{bmatrix}^T, \tag{4.17}$$

$$\tilde{M}_c = \widehat{M}_c - \frac{1}{N} M_c 1_N, \tag{4.18}$$

and

$$\widehat{\beta} = \begin{bmatrix} \widehat{\beta}_1, & \widehat{\beta}_2, & \widehat{\beta}_3, & \cdots, & \widehat{\beta}_N \end{bmatrix}^T. \tag{4.19}$$

We can rewrite (4.1) as

$$\dot{\widehat{M}}_c = \widehat{\beta} - \Lambda L \tilde{M}_c, \tag{4.20}$$

where $\Lambda = diag\{\lambda_1.\lambda_2, \cdots, \lambda_n\}$ and $L \in R^{N \times N}$ is the Laplacian matrix defined as

$$\ell_{ij} = \begin{cases} |\mathcal{N}_i| & \text{if } i = j \\ -1 & \text{if } i \in \mathcal{N}_i \\ 0 & \text{otherwise,} \end{cases} \tag{4.21}$$

where $|\mathcal{N}_i|$ is the cardinality of the set $\mathcal{N}_i$. We assume that the communication topology is bidirectional and connected. Therefore, L is symmetric and positive semidefinite and has the only nontrivial null space spanned by 1_N [39].

4.2.2 Force Control Law

Rewrite (4.2)

$$F_i = -\Gamma_i(\dot{x}_i - v^d) + \hat{f}_i^d + m_i g e_3, \tag{4.22}$$

and construct the z-component of $\hat{f}_i^d$ as:

$$\hat{f}_{i,z}^d = \widehat{M}_{c,i} g. \tag{4.23}$$

For the x and y components of $\hat{f}_i^d$, we assume that they are predefined constant values satisfying

$$\sum_{i=1}^{N} \hat{f}_{i,x}^d = 0 \quad \text{and} \quad \sum_{i=1}^{N} \hat{f}_{i,y}^d = 0. \tag{4.24}$$

4.2.3 Computation of T_i^{des} and Φ_i^{des} for a quadcopter

Given (4.22), the corresponding T_i and Φ_i are found based on (3.6)–(3.8).

4.2.4 Main Result

The closed-loop dynamics (2.8) with (4.16), (4.22) and (4.23) can be expressed as

$$m_i \ddot{x}_i = -\Gamma_i \xi_i + \hat{f}_i^d - f_i. \tag{4.25}$$

Theorem 4.2 *The decentralized adaptive control law in* (4.22), (4.23) *and* (4.16) *asymptotically stabilizes the equilibrium of* (2.7), (4.16), *and* (4.25), *given by*

$$\begin{aligned} \mathscr{E}^* = \{(\dot{x}_i, \dot{x}_c, f_i, \widehat{M}_{c,i}) | \dot{x}_i = v^d, \quad \dot{x}_c = v^d, \\ \widehat{M}_{c,i} = \frac{1}{N} M_c, \quad \textit{and} \quad f_i = \hat{f}_i^d \}. \end{aligned} \tag{4.26}$$

Proof. We first consider the following Lyapunov function

$$\begin{aligned} V_1 = \sum_{i=1}^{N} \left[P_i(z_i) - P_i(z_i^d) - (f_i^d)^T (z_i - z_i^d) \right] \\ + \frac{1}{2} \left(\sum_{i=1}^{N} \xi_i^T m_i \xi_i + \xi_c^T M_c \xi_c \right). \end{aligned} \tag{4.27}$$

From (2.2) and (2.3), the kinematics of z_i is given by

$$\dot{z}_i = \dot{x}_i - \dot{a}_i = (\dot{x}_i - \dot{x}_c) \tag{4.28}$$

Define

$$\xi_i = \dot{x}_i - v^d, \quad \xi_c = \dot{x}_c - v^d \tag{4.29}$$

We compute the time derivative of V_1 as

$$\dot{V}_1 = \sum_{i=1}^{N} (f_i - f_i^d)^T \dot{z}_i + \sum_{i=1}^{N} \xi_i^T m_i \ddot{x}_i + \xi_c^T M_c \ddot{x}_c. \tag{4.30}$$

Rewrite (4.30) from (2.7), (2.8), (2.12), (4.22) and (4.28) as:

$$\begin{aligned}
\dot{V}_1 &= \sum_{i=1}^{N} (f_i - f_i^d)^T (\dot{x}_i - \dot{x}_c) + \sum_{i=1}^{N} \xi_i^T (F_i - f_i) \\
&\quad + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g_3, \\
&= \sum_{i=1}^{N} (f_i - f_i^d)^T (\xi_i - \xi_c) + \sum_{i=1}^{N} \xi_i^T (-\Gamma_i \xi_i + \hat{f}_i^d - f_i) \\
&\quad + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g e_3, \\
&= \sum_{i=1}^{N} (f_i - f_i^d)^T \xi_i - \sum_{i=1}^{N} (f_i - f_i^d)^T \xi_c - \sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i \\
&\quad + \sum_{i=1}^{N} \xi_i^T (\hat{f}_i^d - f_i) + \xi_c^T \sum_{i=1}^{N} f_i - \xi_c^T M_c g e_3, \\
&= - \sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^{N} \xi_i^T \tilde{f}_i^d,
\end{aligned} \tag{4.31}$$

where $\tilde{f}_i^d = \hat{f}_i^d - f_i^d$, i.e.,

$$\tilde{f}_i^d = \begin{pmatrix} 0 \\ 0 \\ (\widehat{M}_{c,i} - \frac{1}{N} M_c g \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \tilde{M}_{c,i} g \end{pmatrix} = \tilde{M}_{c,i} g e_3, \tag{4.32}$$

where $\tilde{M}_{c,i} = \widehat{M}_{c,i} - \frac{1}{N} M_c$. From (4.31) and (4.32),

$$\dot{V}_1 = - \sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^{N} \xi_i^T \tilde{M}_{c,i} g e_3. \tag{4.33}$$

We next take another Lyapunov function:

$$V_2 = \frac{g}{2} \tilde{M}_c^T \Lambda^{-1} \tilde{M}_c. \tag{4.34}$$

Let $W = V_1 + V_2$. The time derivative of W is given by,

$$\begin{aligned}
\dot{W} &= \dot{V}_1 + \dot{V}_2, \\
&= -\sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^{N} \xi_i^T \tilde{M}_{c,i} g e_3 + g \tilde{M}_c^T \Lambda^{-1} \dot{\tilde{M}}_c, \\
&= -\sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^{N} \xi_i^T \tilde{M}_{c,i} g e_3 + g \tilde{M}_c^T \Lambda^{-1} (\widehat{\beta} - \Lambda L \tilde{M}_c), \\
&= -\sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^{N} \xi_i^T \tilde{M}_{c,i} g e_3 + g \tilde{M}_c^T \Lambda^{-1} \widehat{\beta} - g \tilde{M}_c^T \Lambda^{-1} \Lambda L \tilde{M}_c, \\
&= -\sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i + \sum_{i=1}^{N} \xi_i^T \tilde{M}_{c,i} g e_3 - \Lambda^{-1} \Lambda g \sum_{i=1}^{N} \tilde{M}_{c,i} \xi_i^T e_3 - g \tilde{M}_c^T L \tilde{M}_c.
\end{aligned} \tag{4.35}$$

Now from (4.16) and (4.35), we obtain

$$\dot{W} = -\sum_{i=1}^{N} \xi_i^T \Gamma_i \xi_i - g \tilde{M}_c^T L \tilde{M}_c \leq 0, \tag{4.36}$$

Since $\dot{W}$ is negative semi-definite, we conclude that the equilibrium $\mathscr{E}^*$ is stable.

We employ the LaSalle invariance principle [37] to complete the analysis for the asymptotic stability of $\mathscr{E}^*$. Let the largest invariant set where $\dot{W} = 0$ be M. On M, $\xi_i = 0$ and $L\tilde{M}_c = 0$. Since the only null space of L is spanned by 1_N, $L\tilde{M}_c = 0$ implies $\tilde{M}_c = 1_N \alpha$, for some scalar α. Because $\tilde{M}_c = \widehat{M}_c - \frac{1}{N} M_c 1_N = 1_N \alpha$, we conclude that $\widehat{M}_{c,i} = \widehat{M}_{c,j}$, $\forall$ i, j. We further note that $\xi_i = 0$ leads to $\dot{x}_i = v^d$, which means that $\ddot{x}_i = 0$ and $\hat{f}_i^d = f_i$ from (4.25). Because $\dot{x}_i = v^d$, $\dot{\widehat{M}}_{c,i} = 0$, which indicates that $\widehat{M}_{c,i}$ is constant. Thus, f_i and z_i are constant, which leads to $\dot{x}_c = \dot{x}_i = v^d$ and $\ddot{x}_c = 0$. From (2.7), we obtain $\sum_{i=1}^{N} f_i = M_c g e_3$, which, together with $f_i = \hat{f}_i^d$, yields $\sum_{i=1}^{N} \hat{f}_i^d = M_c g$. Given (4.23), it follows that $\sum_{i=1}^{N} \widehat{M}_{c,i} = M_c$ and $\widehat{M}_{c,i} = \widehat{M}_{c,j} = \frac{1}{N} M_c$. ■

Remark 4.1 *Theorem 4.2 guarantees that the agents share the weight of the payload equally. If the second term in* (4.16) *is removed, we still ensure that* $\sum_{i=1}^{N} \widehat{M}_{c,i} \to M_c$, $\dot{x}_i \to v^d$, $\dot{x}_c \to v^d$ *and* $f_i \to \hat{f}_i^d$. *However, the estimates* $\widehat{M}_{c,i}$*'s may not converge to the same value, which in turn implies that the agents provide different vertical forces to balance the payload mass.*

4.2.5 Simulation Results: Force control of a payload with an unknown mass for a constant v^d

We consider three quadcopters with rigid extensions transporting a payload. The mass and inertia are kept identical to our previous chapter 2, [4]. We also employ the same PD control as in Chapter 2, to track the desired orientation for each quadcopter.

Throughout this simulation, we assume that agent 1 communicates with agent 2 and 3 but agent 2 and 3 do not communicate with each other. We choose $\widehat{M}_{c,1}(0) = 0.1$ kg,

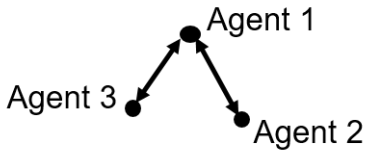


Figure 4.8: Communication topology between each agent.

$\widehat{M}_{c,2}(0) = 0.5$ kg and $\widehat{M}_{c,3}(0) = 1.2$ kg,

$$
\begin{aligned}
v^d &= (1-e^{-t})\begin{bmatrix} 2.0 & 1.0 & 0 \end{bmatrix}^T \text{m/s}, \\
\hat{f}_1^d(t) &= (1-e^{-t})\begin{bmatrix} 0.0 & -2.0 & \widehat{M}_{c,1}g \end{bmatrix}^T, \\
\hat{f}_2^d(t) &= (1-e^{-t})\begin{bmatrix} 2.0 & 1.0 & \widehat{M}_{c,2}g \end{bmatrix}^T, \\
\hat{f}_3^d(t) &= (1-e^{-t})\begin{bmatrix} -2.0 & 1.0 & \widehat{M}_{c,3}g \end{bmatrix}^T, \\
\Gamma_i &= \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 25 \end{bmatrix}, \\
L &= \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}.
\end{aligned}
$$

Note that $v^d(t)$, $f_1^d(t)$, and $f_2^d(t)$ exponentially converge to constants. The term $(1-e^{-t})$ is added to improve performance of the controller during the transition.

Fig. 4.9 and 4.10 illustrate that the velocities of the three agents and the load asymptotically approach v^d. The contact forces f_i's also converge to the desired values f_i^d's as observed in Fig. 4.11. The adaptation law from (4.16) estimates the true mass of the load as illustrated in Fig. 4.12 and the consensus algorithm guarantees that each agent estimates converge to $\frac{1}{3}M_c = 0.5$ kg.

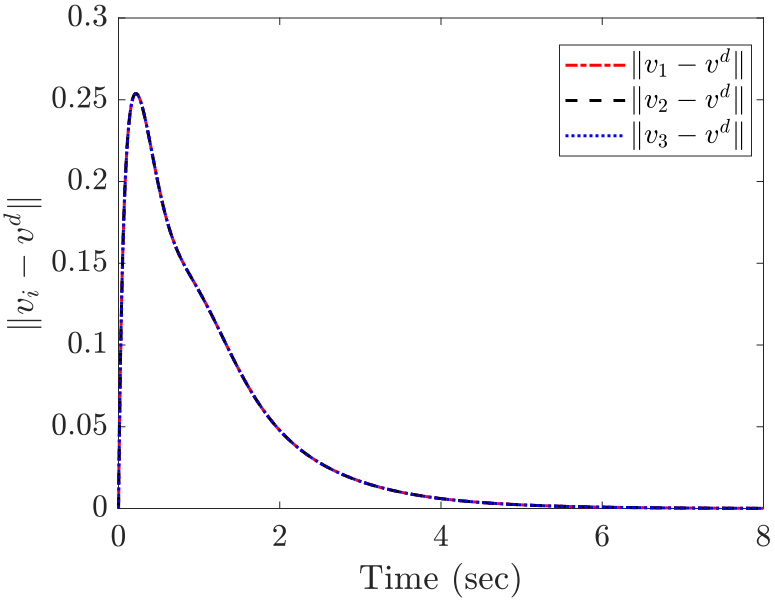


Figure 4.9: The magnitude of the velocity error $(v_i - v^d)$ goes to zero which concludes that v_i converges to predesigned v^d

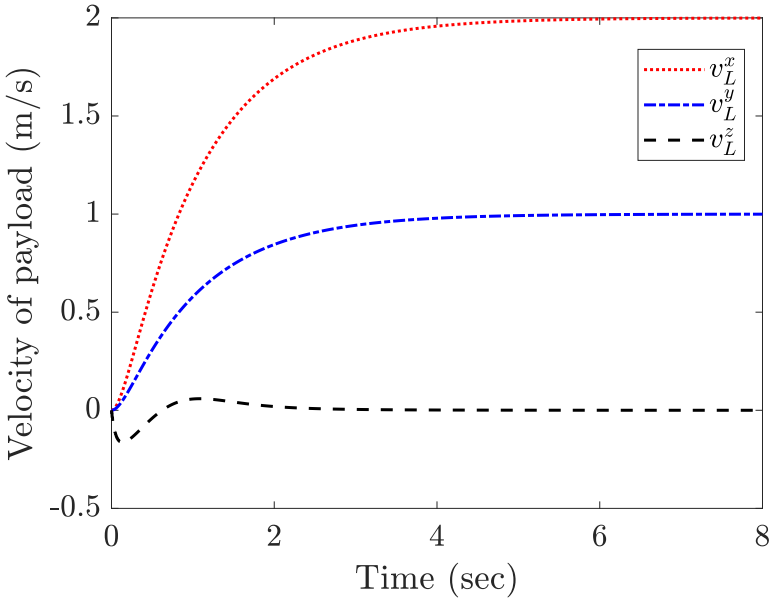


Figure 4.10: Linear velocity plot for the payload. The x and y velocity converges to the desired v^d i.e., 2.0 m/s and 1.0 m/s respectively and the z component converges to zero.

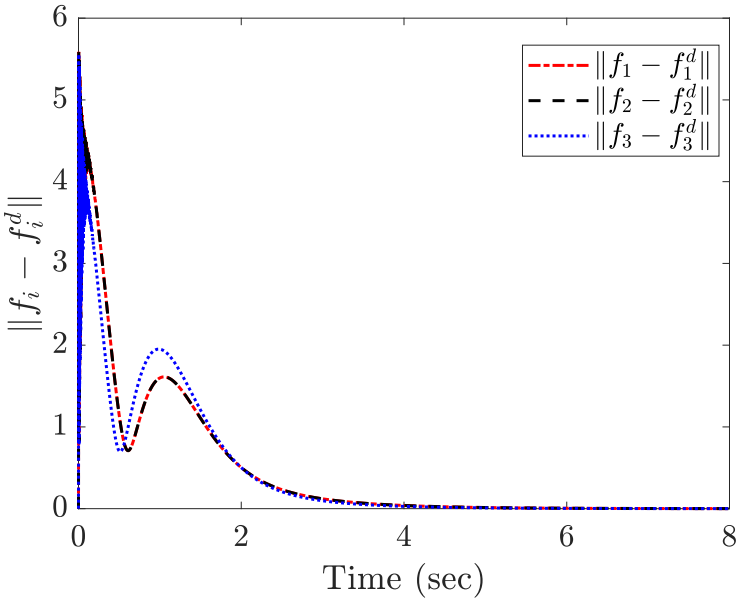


Figure 4.11: The magnitude of $f_i - f_i^d$ goes to zero. All the contact forces f_i converge to f_i^d.

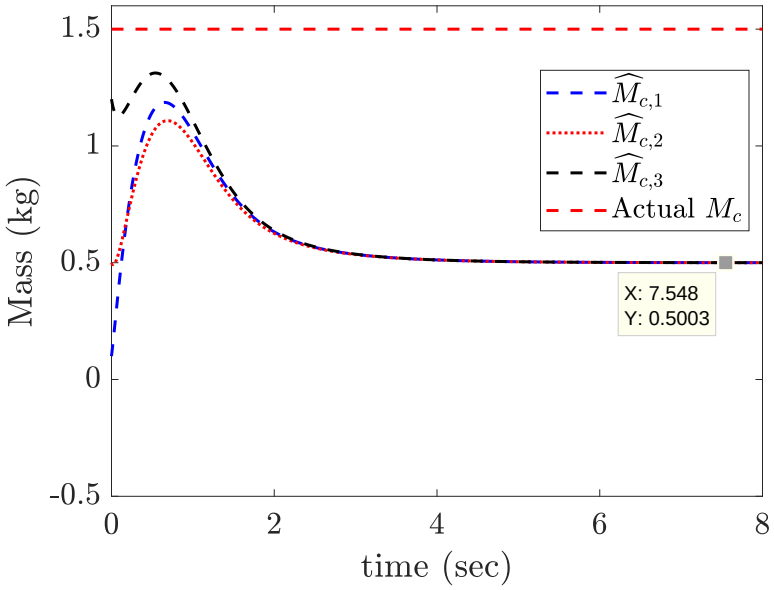


Figure 4.12: Plot of the estimates of the mass of the payload. Each individual estimates converge to $\frac{1}{3}M_c = 0.5$ kg and its sum adds up-to the true mass of the payload i.e., 1.5 kg.

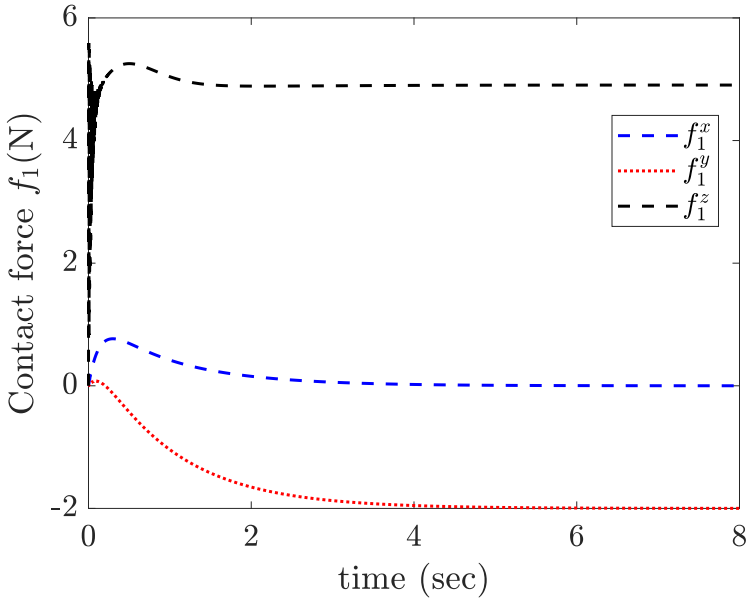


Figure 4.13: Plot for contact forces in all 3 directions for agent 1. It converges to converges 0.0 N, -2.0 N and 0.5 N for each direction respectively. Remaining agents have similar graphs.

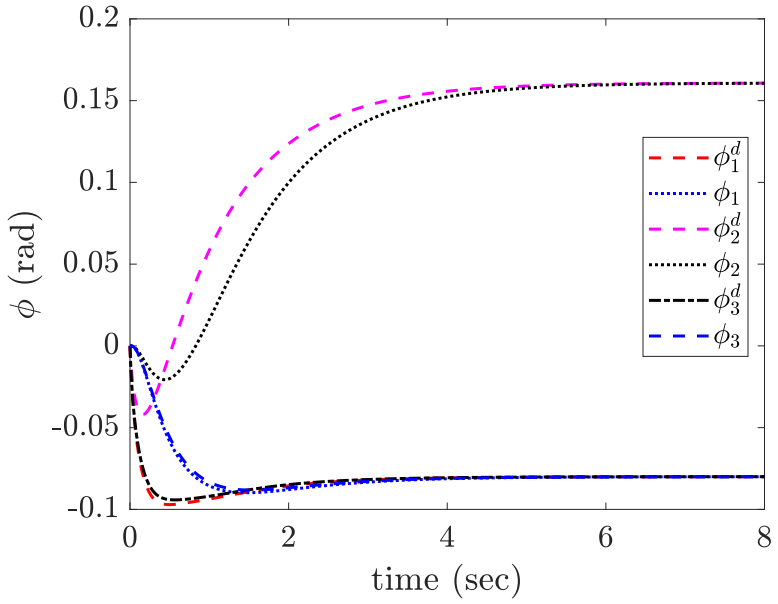


Figure 4.14: Plot for desired and actual orientation angles (ϕ) for three agents in the x-direction.

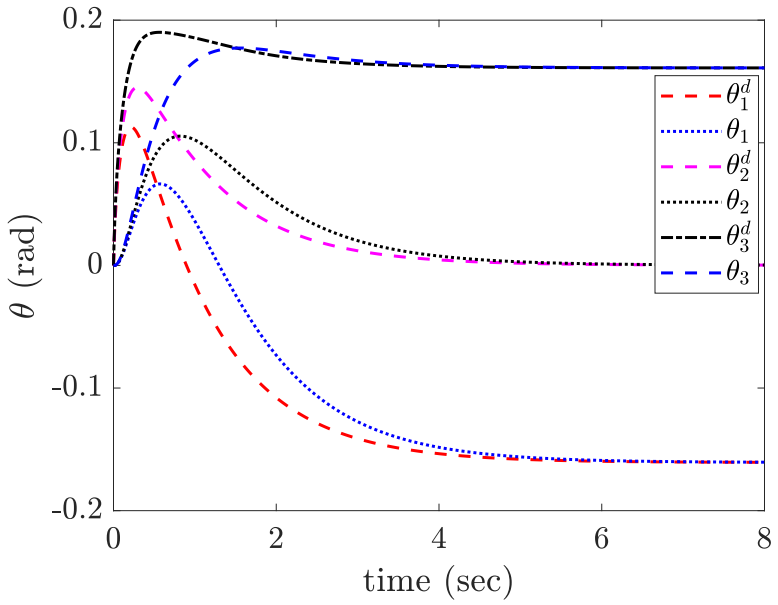


Figure 4.15: Plot for the desired and actual orientation angles (θ) for three agents in the y-direction.

4.2.6 Simulation Results: Force Control of a load with an unknown mass for a time-varying $v^d(t)$

To test the performance of the controllers for different conditions, we illustrate a simulation for a pre-designed time-varying desired velocity. We keep the other parameters identical to the previous constant velocity case and change v^d to the following:

$$v^d(t) = \begin{cases} \begin{bmatrix} 2.0, & 0, & 0 \end{bmatrix}^T, & \text{t} \leq 1.57 \text{ s}, \\ \begin{bmatrix} 2\sin(t), & -2\cos(t), & 0 \end{bmatrix}^T, & 2 < \text{t} \leq 9.42 \text{ s}, \\ \begin{bmatrix} 0, & 2.0, & 0 \end{bmatrix}^T, & \text{t} > 9.42 \text{ s}. \end{cases}$$

Fig. 4.16 and 4.17 illustrate the 2D and 3D trajectories for all three agents and the load respectively. Notice from Fig. 4.18, 4.19 and 4.20 that the velocity errors of the three agents and the load are bounded. The contact force errors are also bounded and stay close to zero as illustrated in Fig. 4.21 and 4.22. The update law in (4.16) recovers the true mass of the load as illustrated in Fig. 4.23 and the consensus algorithm guarantees that each agent estimates converge to $\frac{1}{3}M_c = 0.5$ k even with a time varying velocity in x and y direction. Note that the convergence of the estimates to $\frac{1}{3}M_c = 0.5$ is because the z-component of the desired velocity v^d is still constant. Fig. 4.24 and 4.25 illustrate that the low level PD controller closely tracks the attitude of the quadcopter during the transient and as well as steady state.

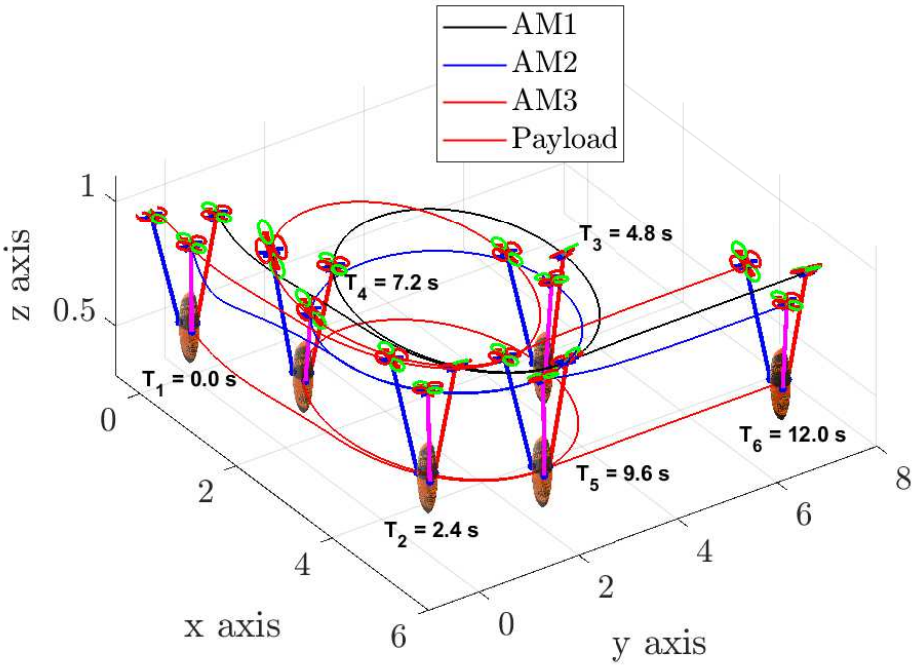


Figure 4.16: Snapshot of 3D position trajectories for all the agents and the payload at different time instance for adaptive design with time varying velocity.

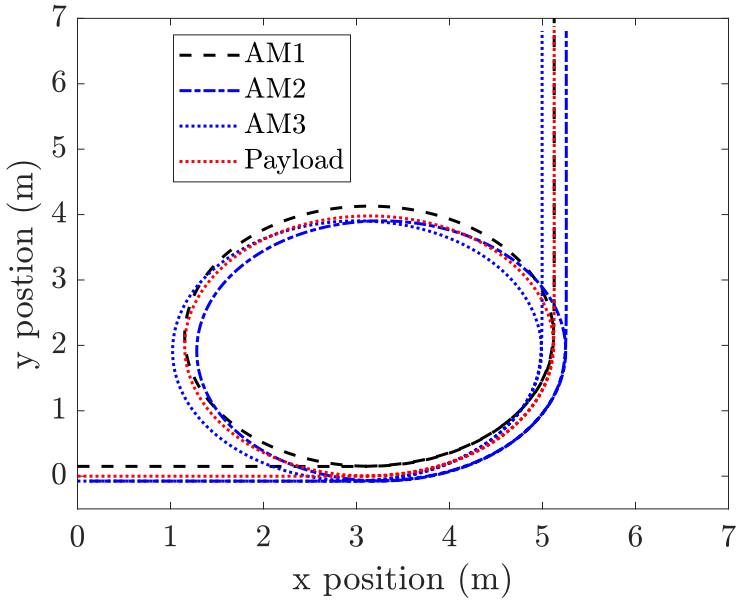


Figure 4.17: Position trajectories for aerial robots and the payload in x and y direction.

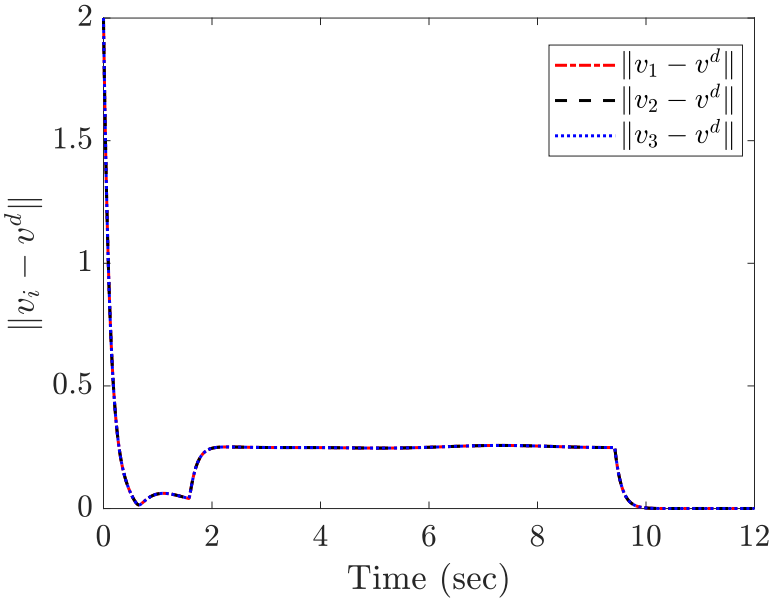


Figure 4.18: The magnitude of the velocity error ($v_i - v^d$) for the agents.

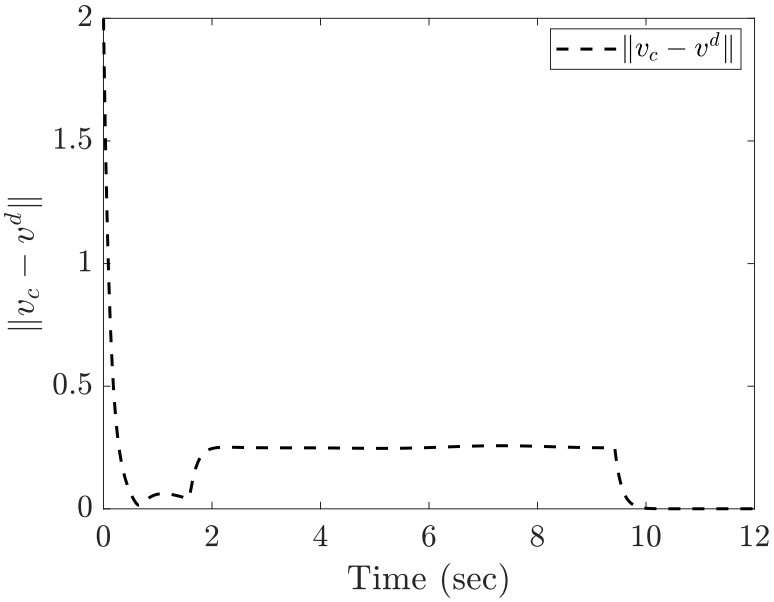


Figure 4.19: The magnitude of the velocity error ($v_c - v^d$) for the payload.

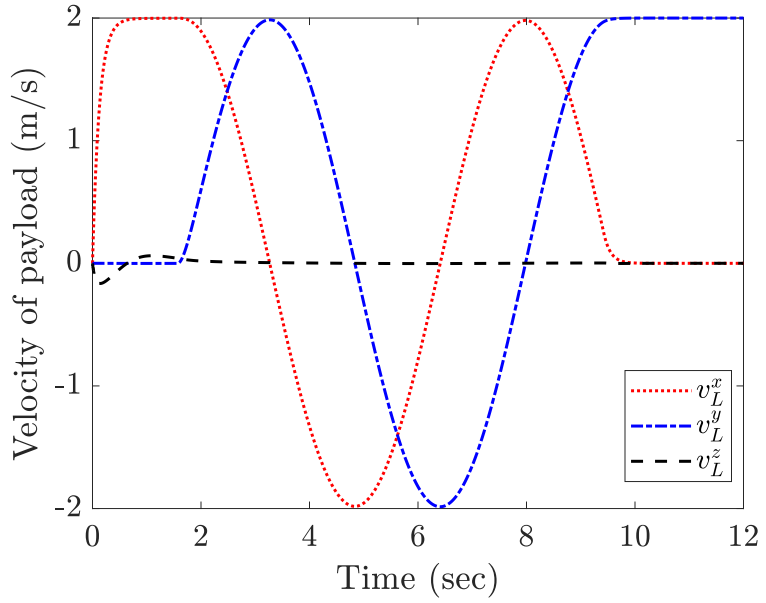


Figure 4.20: Linear velocity for the payload in all 3 directions.

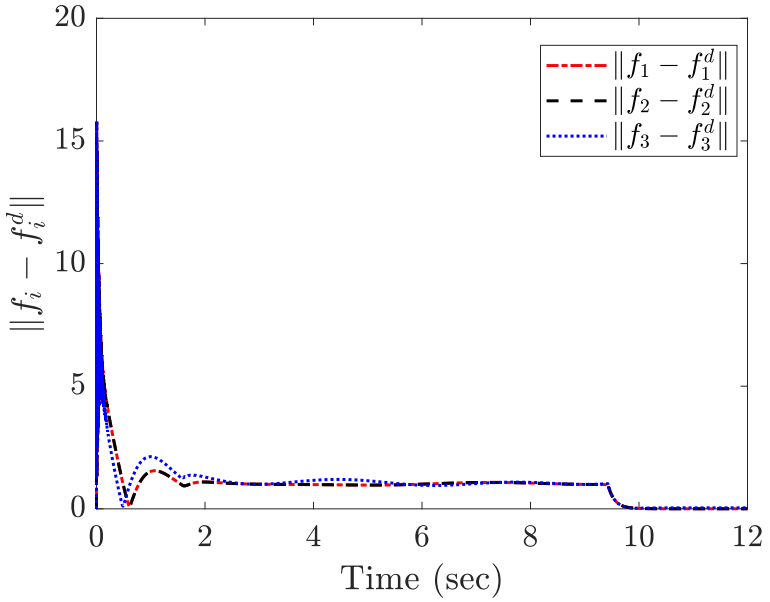


Figure 4.21: The magnitude of $f_i - f_i^d$ stays bounded and eventually converges to zero.

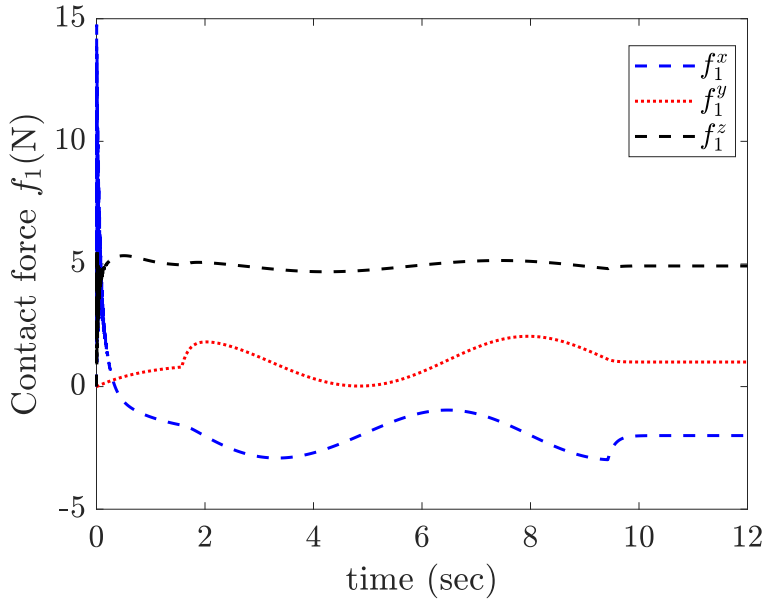


Figure 4.22: Plot of squeeze forces in all 3 directions for agent 1. Note that other agents have similar plot.

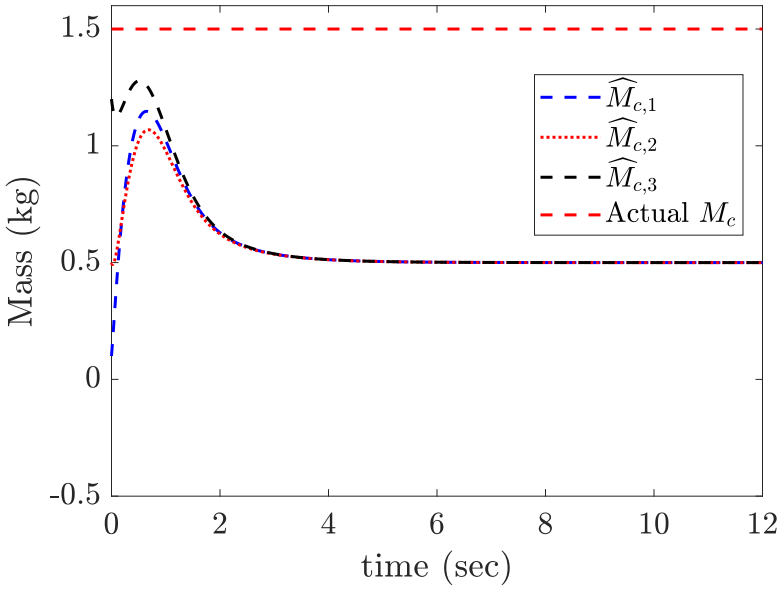


Figure 4.23: Plot of the estimates of the mass of the payload. Eachg individual estimates converges $\frac{1}{3}M_c = 0.5$ kg. Notice that their sum adds up-to the true mass of the payload used in this simulation. The convergence of the estimates to $\frac{1}{3}M_c = 0.5$ is because the z-component of the desired velocity v^d is still constant.

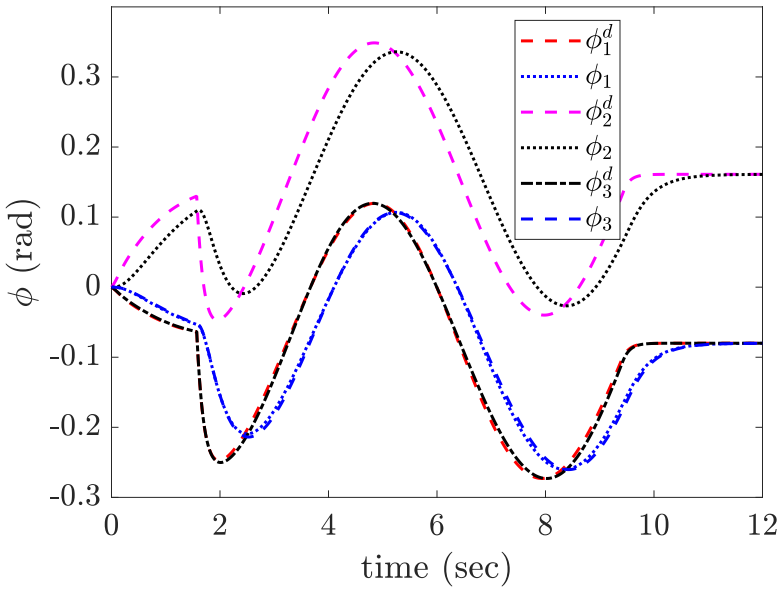


Figure 4.24: Plot for desired and actual orientation angles (ϕ) for three agents in the x-direction.

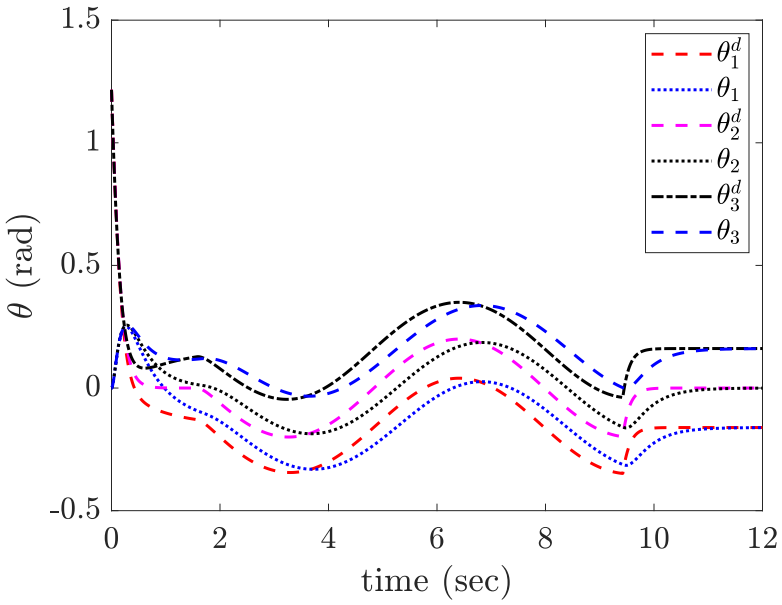


Figure 4.25: Plot for the desired and actual orientation angles (θ) for three agents in the y-direction.

CHAPTER 5

COOPERATIVE AERIAL MANIPULATION WITH ATTITUDE STABILIZATION

In our previous Chapters 3 and 4 we have developed force and motion control algorithms that guarantee force regulation and translational motion of the quadcopters and the payload. For complete motion coordination and path planning, orientation control is important. In this chapter, we extend our work in Chapters 3 and 4, to address both the orientation control of the payload and the force control during cooperative aerial manipulation. In particular, we develop time varying force setpoints to enforce attitude regulation. The algorithm provides desired thrust and attitude angles required for each quadcopter to cooperatively transport the payload. We analyze the stability of the system using singular perturbation theory. We demonstrate the effectiveness of the algorithms in numerical simulations.

5.1 Related Work

Reference [18] uses three or more quadcopters for payload transport to balance the attitude of the payload. Reference [27] considers a ring shaped flexible structure with multiple attached flying vehicles and develops a LQR controller for linearized dynamics to track the desired pose of the payload. It is assumed that the vehicles can exert both forces and moments on the payload. However, we assume that the inputs from the quadcopters to the payload are only contact forces with no moments.

5.2 Problem Formulation

5.2.1 Kinematics

Let the orientation of the load with respect to Σ_I be $R_c \in SO(3)$. Let $q = \begin{bmatrix} q_0 \\ q_v \end{bmatrix}$ be the unit quaternion representation parameterizing R_c. Then R_c is related to q through the Rodriguez formula [40]:

$$R_c = I_3 + 2(\widehat{q_v})^2 + 2q_0\widehat{q_v}, \tag{5.1}$$

where the operator $\widehat{\ }$ converts a given vector $\omega = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^T$ into a skew symmetric matrix as

$$\widehat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \tag{5.2}$$

Note that $\widehat{x}y = x \times y$ for all $x, y \in \mathbb{R}^3$. The kinematic equations of q are given by

$$\frac{dq_0}{dt} = -\frac{1}{2}(\Omega_c)^T q_v, \tag{5.3}$$

$$\frac{dq_v}{dt} = \frac{1}{2}q_0\Omega_c + \frac{1}{2}\widehat{\Omega}_c q_v, \tag{5.4}$$

where Ω_c is the angular velocity of the payload in the inertial frame.

Because the contact points a_i satisfy

$$a_i(t) := x_c(t) + R_c r_i, \tag{5.5}$$

the kinematics of the contact points are given by

$$\dot{a}_i = \dot{x}_c + (\dot{R_c} r_i) = \dot{x}_c + \widehat{\Omega}_c R_c r_i. \tag{5.6}$$

5.2.2 Dynamics

We model the contact force between the agents and the payload as a mass spring model. This assumption is valid when the vehicles are connected to the payload via

elastic cables or when the payload is flexible or surrounded by elastic bumper materials. When the agents move, the payload experiences tension or compression. Therefore, $a_i(t) \neq x_i(t)$. We approximate the deformation as

$$z_i = x_i - a_i, \quad i = 1, \cdots, N. \tag{5.7}$$

In this paper, we consider a linear spring model for the contact force, i.e., $f_i = kz_i$, where k is the spring constant (possibly unknown). If the contact between the agents and the payload is elastic at a constant pressure, the linear spring model is a valid representation of the contact force as shown in [29].

Rewrite the translational dynamics of the load as

$$M_c \ddot{x}_c = \sum_{i=1}^{N} f_i - M_c g e_3, \tag{5.8}$$

where M_c is the mass of the load, g is the gravitational constant, and e_3 is the unit vector $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$. The rotational dynamics of the payload is given by

$$I_c \dot{\Omega}_c = \sum_{i=1}^{N} \widehat{R_c r_i} f_i - \widehat{\Omega}_c I_c \Omega_c, \tag{5.9}$$

where the $\sum_{i=1}^{N} \widehat{R_c r_i} f_i$ term represents the total moment exerted on the payload due to the contact forces. We assume that the agents cannot directly exert moments on the payload.

Rewrite the translational dynamics of the N agents as

$$m_i \ddot{x}_i = F_i - f_i - m_i g e_3, \quad i = 1, ..., N, \tag{5.10}$$

where m_i is the mass of the i-th agent, F_i is the force applied to agent i and f_i is the contact force to agent i. For a quadcopter, the applied force F_i is given by

$$F_i = T_i R_i(\Phi) e_3, \tag{5.11}$$

where T_i is the total thrust from the four rotors, given by

$$T_i = \sum_{i=1}^{4} F_{\text{rotors},i}, \tag{5.12}$$

$\Phi = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T$ are the roll, pitch and yaw angles and $R_i(\Phi) \in SO(3)$ is the rotation matrix between the body frame $O_{c,i}$ and the inertial frame Σ_i given by

$$R_i = \begin{bmatrix} c\psi_i c\theta_i & c\psi_i s\phi_i s\theta_i - c\phi_i s\psi_i & s\phi_i s\psi_i + c\phi_i c\psi_i s\theta_i \\ c\theta_i s\psi_i & c\phi_i c\psi_i + s\phi_i s\psi_i s\theta_i & c\phi_i s\psi_i s\theta_i - c\psi_i s\phi_i \\ -s\theta_i & c\theta_i s\phi_i & c\phi_i c\theta_i \end{bmatrix}, \tag{5.13}$$

in which $c\theta_i = \cos\theta_i$ and $s\theta_i = \sin\theta_i$. The rotational dynamics of the i-th quadcopter is given by

$$I_i \dot{\Omega}_i = M_i - \widehat{\Omega}_i I_i \Omega_i, + \tau_{i,f}, \tag{5.14}$$

where I_i is the inertia matrix, $M_i \in \mathbb{R}^3$ is the moment generated by the ith-quadcopter and $\tau_{i,f}$ is the torque generated due to the contact force f_i.

When f_i is measured, e.g., based on a force sensor or an accelerometer, we may cancel $\tau_{i,f}$ by choosing $M_i = \bar{M}_i - \tau_{i,f}$. Then we can design $\bar{M}_i$ to control the attitude dynamics of the i-th quadcopter.

5.2.3 Control objective

Our control objective is to design F_i such that all the agents and the payload translate with a constant velocity v^d while the orientation of the payload R_c is driven to a desired setpoint R_c^d, i.e., $\dot{x}_i \to v^d$, $\forall i$, $\dot{x}_c \to v^d$, $\Omega_c \to 0$, and $R_c \to R_c^d$. Meanwhile, we also regulate the contact force f_i to desired squeezing force f_i^d, i.e., $f_i \to f_i^d$, $\forall i$.

5.3 Control Design

Compared with Chapters 4 and 6, our control objective consists of an additional goal of stabilizing R_c to R_c^d. To achieve this goal, we design time-varying force setpoints $f_i^d(t)$ such that when substituted in (5.9), the $f_i^d(t)$'s achieve the orientation control. Specifically, we choose $f_i^d(t)$ dynamically to satisfy

$$\sum_{i=1}^{N} f_i^d = M_c g e_3, \tag{5.15}$$

$$\sum_{i=1}^{N} \widehat{R_c r_i} f_i^d = -k_{\Omega_c}\Omega_c - k_{q_v} q_v^e, \tag{5.16}$$

where k_{Ω_c} and k_{q_v} are positive scalars, and q_v^e is the vector part of the quaternion parameterizing $(R_c^d)^T R_c$. Note that the right hand side of (5.16) is an exponentially stable attitude control that ensures $\Omega_c \to 0$ and $q_v^e \to 0$ [40, 41]. Therefore, if $f_i = f_i^d$, it follows from (5.8) and (5.9) and the constraints (5.15) and (5.16) that the payload will move at a constant velocity with a stabilized orientation.

Define

$$u_i = f_i^d(t) - f_i \tag{5.17}$$

and

$$\xi_i := \dot{x}_i - v^d - \Gamma_i u_i \quad \Gamma_i = \Gamma_i^T > 0. \tag{5.18}$$

We next design F_i as

$$F_i = -K_i \xi_i + f_i^d(t) + m_i \Gamma_i \dot{u}_i + m_i g e_3, \tag{5.19}$$

where $K_i = K_i^T > 0$. As we shall see in Section 5.4, the $\dot{u}_i$ term allows us to establish stability of the combined orientation and force control system.

Remark 5.1 *From* (5.15) *and* (5.16)*, we obtain*

$$\underbrace{\begin{bmatrix} I & I & \dots & I \\ \widehat{R_c r_1} & \widehat{R_c r_2} & \dots & \widehat{R_c r_N} \end{bmatrix}}_{\Lambda} \underbrace{\begin{bmatrix} f_1^d \\ f_2^d \\ \vdots \\ f_N^d \end{bmatrix}}_{f^d} = \underbrace{\begin{bmatrix} M_c g e_3 \\ -k_{\Omega_c}\Omega_c - k_{q_v} q_v^e \end{bmatrix}}_{h(\Omega_c, q_v^e)}. \tag{5.20}$$

Thus, f^d can be solved as

$$f^d = \Lambda^+ h(\Omega_c, q_v^e) + f_{N(\Lambda)}^d, \tag{5.21}$$

where $\Lambda^+ = \Lambda^T(\Lambda\Lambda^T)^{-1}$ is the Moore-Penrose generalized inverse of Λ and $f_{N(\Lambda)}^d$ belongs to the null space of Λ, i.e., $\Lambda f_{N(\Lambda)}^d = 0$.

Note that f^d consists of two components: the first one solving the constraints (5.15) *and* (5.16) *and the second one aligned with the squeezing direction (the null space of Λ). The f^d is a function of (Ω_c, q_v^e) since its two components depend only on (Ω_c, q_v^e). At the desired equilibrium where $\Omega_c = 0$ and $q_v^e = 0$, the sum of f_i^d's in the z direction balances the weight of the load and the f_i^d's contribute no moments to the payload.*

The $\dot{f}_i^d$ used in $\dot{u}_i$ can be computed in a similar fashion. The expression of $\dot{f}_i^d$ is given in the Appendix. ■

5.3.1 Computation of T_i^{des} and Φ_i^{des} for a quadcopter

Given (5.11), the corresponding T_i and Φ_i are found based on (3.6)–(3.8). Once θ_i^{des}, ϕ_i^{des} and T_i^{des} are computed, low-level attitude and thrust tracking controllers (e.g., a PD controller) can be implemented to track these desire commands for the i-th quadcopter.

5.4 Stability Analysis

5.4.1 Closed-loop system and equilibria

In this section, we derive the closed-loop dynamics. We note that $u_i = k(z_i^d - z_i)$, where $z_i^d = f_i^d / k$. Define $\tilde{z}_i = z_i^d - z_i$, $\forall i$. Then $u_i = 0$ is equivalent to $\tilde{z}_i = 0$. Therefore, we derive the dynamics for $\tilde{z}_i$. Differentiating (5.7) leads to

$$\dot{z}_i = \dot{x}_i - \dot{a}_i = \xi_i + v^d + \Gamma_i u_i - \dot{x}_c - \widehat{\Omega}_c R_c r_i, \tag{5.22}$$

which can be further written as

$$\begin{aligned} \dot{z}_i &= \xi_i - \xi_c + \Gamma_i u_i - \widehat{\Omega}_c R_c r_i \\ &= \xi_i - \xi_c + K_z \tilde{z}_i - \widehat{\Omega}_c R_c r_i, \end{aligned} \tag{5.23}$$

where $\xi_c = \dot{x}_c - v^d$ and $K_z = \Gamma_i k$. Thus, we obtain

$$\dot{\tilde{z}}_i = -\xi_i + \xi_c - K_z \tilde{z}_i + \widehat{\Omega}_c R_c r_i + \dot{z}_i^d. \tag{5.24}$$

Note from (5.46) that $\dot{z}_i^d$ is a function of $(\Omega_c, q_v^e, \tilde{z}_i)$.

For the translational motion of the payload, we obtain from (5.8) and (5.15) that

$$M_c\dot{\xi}_c = \sum_{i=1}^{N}(f_i - f_i^d) = -k\sum_{i=1}^{N}\tilde{z}_i. \tag{5.25}$$

The derivative of ξ_i defined in (5.18) is given by

$$m_i\dot{\xi}_i = m_i\ddot{x}_i - m_i\Gamma_i\dot{u}_i = F_i - f_i - m_i g e_3 - m_i\Gamma_i\dot{u}_i, \tag{5.26}$$

which, together with the design of F_i in (5.19), leads to

$$\begin{aligned} m_i\dot{\xi}_i &= -K_i\xi_i + f_i^d + m_i\Gamma_i\dot{u}_i + m_i g e_3 \\ &\quad - f_i - m_i g e_3 - m_i\Gamma_i\dot{u}_i \\ m_i\dot{\xi}_i &= -K_i\xi_i + k\tilde{z}_i. \end{aligned} \tag{5.27}$$

The rotational dynamics in (5.9) with f_i^d in (5.16) takes the following form:

$$I_c\dot{\Omega}_c = -\widehat{\Omega}_c I_c\Omega_c - k\sum_{i=1}^{N}\widehat{R_c r_i}\tilde{z}_i - k_{\Omega_c}\Omega_c - k_{q_v}q_v^e, \tag{5.28}$$

where the dynamics of the unit quaternion (q_v^0, q_v^e) is given by

$$\dot{q}_0^e = -\frac{1}{2}\big((R_c^d)^T\Omega_c\big)^T q_v^e, \tag{5.29}$$

$$\dot{q}_v^e = \frac{1}{2}q_0^e\big((R_c^d)^T\Omega_c\big) + \frac{1}{2}\widehat{(R_c^d)^T\Omega_c}\,q_v^e. \tag{5.30}$$

The desired equilibrium of (5.24), (5.25), (5.27), (5.28), and (5.30) is given by the origin of $\{\Xi, \xi_c, \tilde{z}, \Omega_c, q_v^e\}$, where $\Xi = (\xi_1^T, \cdots, \xi_N^T)^T$ and $\tilde{z} = (\tilde{z}_1^T, \cdots, \tilde{z}_N^T)$.

5.4.2 Main result

Theorem 5.1 *Suppose that $K_z = \Gamma_i k = \lambda I > 0$ and λ is sufficiently large. Define $\zeta_i = \dot{\tilde{z}}_i$. The control law* (5.19) *ensures that the equilibria in* (5.31) *below are exponentially stable.*

$$\begin{aligned} \mathscr{E} = \{(\xi_i, \zeta_i, \tilde{z}_i, \Omega_c, q_v^e) | \xi_i = 0,\ \zeta_i = 0, \\ \tilde{z}_i = 0,\ \Omega_c = 0,\ \textit{and}\ q_v^e = 0,\ \forall i\}. \end{aligned} \tag{5.31}$$

In addition, convergence to $\mathscr{E}$ is equivalent to convergence to the origin of $\{\Xi, \xi_c, \tilde{z}, \Omega_c, q_v^e\}$.

■

Proof. From $\zeta_i = \dot{\tilde{z}}_i$ and (5.24), we obtain

$$\zeta_i = \dot{\tilde{z}}_i = -\xi_i + \xi_c - K_z \tilde{z}_i + \widehat{\Omega}_c R_c r_i + \dot{z}_i^d, \forall i. \tag{5.32}$$

We first divide both sides of (5.32) by K_z and obtain

$$\epsilon \dot{\tilde{z}}_i = -\epsilon\xi_i + \epsilon\xi_c - \tilde{z}_i + \epsilon\widehat{\Omega}_c R_c r_i + \epsilon \dot{z}_i^d, \tag{5.33}$$

where $\epsilon = 1/\lambda$ and $\dot{z}_i^d$ is given in (5.46).

We next derive the dynamics of ζ_i. We obtain from (5.32), (5.25), (5.27), and (5.28) that

$$\begin{aligned}
\dot{\zeta}_i &= -\dot{\xi}_i + \dot{\xi}_c - K_z \dot{\tilde{z}}_i + \dot{\widehat{\Omega}}_c R_c r_i + \widehat{\Omega}_c \widehat{\Omega}_c R_c r_i + \ddot{z}_i^d \\
\dot{\zeta}_i &= \frac{1}{m_i} K_i \xi_i - \frac{k}{m_i} \tilde{z}_i - \frac{k}{M_c} \sum_{i=1}^{N} \tilde{z}_i - K_z \zeta_i \\
&- \Big(I_c^{-1} \Big(\sum_{i=1}^{N} \widehat{R_c r_i} k \tilde{z}_i + \widehat{\Omega}_c I_c \Omega_c + k_{\Omega_c} \Omega_c + k_{q_v} q_v^e \Big) \Big) \times R_c r_i \\
&+ (\widehat{\Omega}_c)^2 R_c r_i + \ddot{z}_i^d.
\end{aligned} \tag{5.34}$$

Dividing both sides by K_z yields

$$\begin{aligned}
\epsilon \dot{\zeta}_i = \epsilon &\frac{1}{m_i} K_i \xi_i - \frac{1}{m_i} \Gamma_i^{-1} \tilde{z}_i - \frac{1}{M_c} \Gamma_i^{-1} \sum_{i=1}^{N} \tilde{z}_i \\
&- \zeta_i + \epsilon (\widehat{\Omega}_c)^2 R_c r_i + \epsilon \ddot{z}_i^d + \Gamma_i^{-1} \widehat{R_c r_i} (I_c^{-1} \sum_{i=1}^{N} \widehat{R_c r_i} \tilde{z}_i) \\
&+ \epsilon \widehat{R_c r_i} (I_c^{-1} \widehat{\Omega}_c I_c \Omega_c) + \epsilon \widehat{R_c r_i} (I_c^{-1} k_{\Omega_c} \Omega_c) \\
&+ \epsilon \widehat{R_c r_i} (I_c^{-1} k_{q_v} q_v^e).
\end{aligned} \tag{5.35}$$

Further computation from (5.46) shows that $\ddot{z}_i^d$ is a function of $(\Omega_c, \tilde{z}_i, q_v^e, \zeta_i)$ and does not contain terms multiplied by K_z or k.

For a sufficiently small ϵ, there exists a time-scale separation between the fast dynamics (5.33) and (5.35) and the slow dynamics of ξ_i, Ω_c and q_v^e in (5.27), (5.28) and (5.30), respectively. To analyze such a singularly perturbed system, we first set $\epsilon = 0$ and equate the right hand side of (5.33) and (5.35) to zero to obtain

$$\tilde{z}_i = 0 \quad \text{and} \quad \zeta_i = 0, \ \forall i. \tag{5.36}$$

Substituting (5.36) into (5.27) and (5.28) yields the reduced order system, which consists of

$$\dot{\xi}_i = -\frac{K_i}{m_i}\xi_i, \tag{5.37}$$

$$\dot{\Omega}_c = -I_c^{-1}\Big(k_{\Omega_c}\Omega_c + \widehat{\Omega}_c I_c \Omega_c + k_{qv} q_v^e\Big), \tag{5.38}$$

and (5.30). The origin of the reduced system is exponentially stable because the ξ_i dynamics are exponentially stable and the dynamics of Ω_c and q_v^e are also exponentially stable according to [40, 41].

The boundary layer system of (5.33) and (5.35) is given by

$$\frac{d\tilde{z}_i}{d\tau} = -\tilde{z}_i, \tag{5.39}$$

$$\frac{d\zeta_i}{d\tau} = -\zeta_i - \Gamma_i^{-1}\left(\frac{1}{m_i}\tilde{z}_i + \frac{1}{M_c}\sum_{i=1}^{N}\tilde{z}_i - \widehat{R_c r_i}\big(I_c^{-1}\sum_{i=1}^{N}\widehat{R_c r_i}\tilde{z}_i\big)\right). \tag{5.40}$$

Note that (5.39) and (5.40) is a cascaded system, where the coupling term in (5.40) is linear in $\tilde{z}_i$ with a bounded growth rate. Without the coupling term, (5.39) and (5.40) are both exponentially stable linear systems. Therefore, it is straightforward to employ a Lyapunov function $V = \frac{1}{2}\sum_{i=1}^{N}\tilde{z}_i^T P\tilde{z}_i + \zeta_i^T\zeta_i$ with a sufficiently large P to show that the origin of $(\tilde{z},\zeta)$ is exponentially stable, where $\zeta = (\zeta_1^T,\cdots,\zeta_N^T)$. We then conclude from [37, Theorem 11.4], the origin of the combined system (5.27), (5.28), (5.30), (5.33), and (5.35) given in (5.31) is exponentially stable for a sufficiently large $\Gamma_i k$.

We now show that convergence to $\mathscr{E}$ is equivalent to convergence to the origin of $\{\Xi,\xi_c,\tilde{z},\Omega_c,q_v^e\}$. Specifically, we establish that convergence to $\mathscr{E}$ leads to $\xi_c \to 0$. From (5.24), $\dot{\tilde{z}}_i = \zeta_i \to 0$, $\xi_i \to 0$, $\Omega_c \to 0$, and $\tilde{z} \to 0$, we conclude that $\xi_c + \dot{z}_i^d \to 0$. The expression of $\dot{z}^d$ in (5.46) indicates that $\dot{z}^d - \frac{1}{k}\dot{f}_{N(\Lambda)}^d \to 0$. As $\Omega_c \to 0$ and $q_v^e \to 0$, R_c approaches R_c^d, which means that $f_{N(\Lambda)}^d$ can be chosen to converge to a constant vector and thus $\dot{f}_{N(\Lambda)}^d \to 0$. Therefore, it follows that $\dot{z}_i^d \to 0$ and $\xi_c \to 0$. ■

The proof for Theorem 5.1 illustrates that with a sufficiently large $\Gamma_i k$, the force error dynamics evolves at the fast time-scale while the velocity dynamics of the agents and the rotational dynamics of the payload evolve at the slow time-scale. If K_i in (5.27) is also chosen sufficiently large, the velocity dynamics of the agents ξ_i becomes part of the fast dynamics. We can follow a similar singular perturbation analysis to obtain the same stability and convergence results.

The value of $\Gamma_i k$ depends on both Γ_i and k. For a stiff spring, where k is large, a moderate Γ_i will be sufficient. For a soft spring where k is small, a large Γ_i is needed. Although our analysis assumes a uniform k for all the agents, the analysis can be easily extended to non-uniform cases.

The current implementation of F_i requires f_i, $\dot{f}_i$, f_i^d and $\dot{f}_i^d$. The $\dot{f}_i$ may be estimated using a derivative estimator, such as [42], based on the f_i measurements. As shown in the below, the expression of $\dot{f}_i^d$ is complicated and may be difficult to implement. We are currently investigating the sensitivity and robustness of the proposed algorithm with respect to $\dot{f}_i^d$.

5.4.3 Computation of $\dot{f}_i^d$ and $\dot{z}_i^d$

The time derivative of (2.12) yields

$$\sum_{i=1}^{N} \dot{f}_i^d = 0. \tag{5.41}$$

The time derivative of (5.16) yields

$$\begin{aligned}\sum_{i=1}^{N}\widehat{(R_c r_i)}\dot{f}_i^d &= -k_{\Omega_c}I_c^{-1}\Big(-\widehat{\Omega}_c I_c\Omega_c - \sum_{i=1}^{N}\widehat{R_c r_i}u_i \\ &\quad - k_{\Omega_c}\Omega_c - k_{q_v}q_v^e\Big) \\ &\quad - k_{q_v}\dot{q}_v^e - \sum_{i=1}^{N}(\widehat{{}_cR_c r_i\Omega})f_i^d \\ &= k_{\Omega_c}I_c^{-1}\widehat{\Omega}_c I_c\Omega_c + k_{\Omega_c}\sum_{i=1}^{N}\widehat{R_c r_i}u_i \\ &\quad - k_{\Omega_c}I_c^{-1}k_{q_v}q_v^e - k_{q_v}\dot{q}_v^e + k_{\Omega_c}^2 I_c^{-1}\Omega_c \\ &\quad + k_{qv}\widehat{\Omega}_c q_v^e + \sum_{i=1}^{N}\widehat{R_c r_i}\widehat{\Omega}_c f_i^d.\end{aligned} \tag{5.42}$$

From (5.41) and (5.42), we obtain

$$\begin{aligned}&\begin{bmatrix} I & I & \dots & I \\ \widehat{R_c r_1} & \widehat{R_c r_2} & \dots & \widehat{R_c r_N}\end{bmatrix}\begin{bmatrix}\dot{f}_1^d \\ \dot{f}_2^d \\ \vdots \\ \dot{f}_N^d\end{bmatrix} \\ &= k_{\Omega_c}k\begin{bmatrix} 0 & 0 & \dots & 0 \\ \widehat{R_c r_1} & \widehat{R_c r_2} & \dots & \widehat{R_c r_N}\end{bmatrix}\begin{bmatrix}\tilde{z}_1 \\ \tilde{z}_2 \\ \vdots \\ \tilde{z}_N\end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & 0 & \dots & 0 \\ \widehat{R_c r_1}\widehat{\Omega}_c & \widehat{R_c r_2}\widehat{\Omega}_c & \dots & \widehat{R_c r_N}\widehat{\Omega}_c\end{bmatrix}\begin{bmatrix}f_1^d \\ f_2^d \\ \vdots \\ f_3^d\end{bmatrix} \\ &\quad + \begin{bmatrix} O \\ k_{\Omega_c}I_c^{-1}\widehat{\Omega}_c I_c\Omega_c + k_{\Omega_c}^2 I_c^{-1}\Omega_c\end{bmatrix} \\ &\quad + \begin{bmatrix} O \\ -k_{\Omega_c}I_c^{-1}k_{q_v}q_v^e - k_{q_v}\dot{q}_v^e + k_{q_v}\widehat{\Omega}_c q_v^e\end{bmatrix},\end{aligned} \tag{5.43}$$

which can be written as

$$\Lambda \dot{f}^d = g(\Omega_c, f^d, \tilde{z}, q_v^e) = \bar{g}(\Omega_c, \tilde{z}, q_v^e) \tag{5.44}$$

since f^d in (5.21) is only a function of Ω_c and q_v^e. Then $\dot{f}^d$ can be obtained as

$$\dot{f}^d = \Lambda^+ \bar{g}(\Omega_c, \tilde{z}, q_v^e) + \dot{f}^d_{N(\Lambda)} \tag{5.45}$$

where Λ^+ is the Moore-Penrose generalized inverse of Λ and $\dot{f}^d_{N(\Lambda)}$ belongs to the null space of Λ. Because $\dot{f}^d = k\dot{z}^d$, we obtain

$$\dot{z}^d = \frac{1}{k}(\Lambda^+ \bar{g}(\Omega_c, \tilde{z}, q_v^e) + \dot{f}^d_{N(\Lambda)}). \tag{5.46}$$

5.4.4 Useful identities

$$\hat{\omega}^T = -\hat{\omega} \tag{5.47}$$

$$\hat{\omega}\omega = 0 \tag{5.48}$$

$$\hat{v}\omega = -\hat{\omega}v \implies \vec{v} \times \vec{\omega} = -\vec{\omega} \times \vec{v} \tag{5.49}$$

$$\dot{R} = R\hat{\omega} \tag{5.50}$$

$$\frac{d}{dt}||b|| = \frac{b^T.\dot{b}}{||b||} \tag{5.51}$$

$$\frac{d(r_1 \times r_2)}{dt} = \dot{r}_1 \times r_2 + r_1 \times \dot{r}_2 \tag{5.52}$$

$$\text{Product Rule: } \frac{d}{dt}(f \cdot g) = \dot{f}g + f\dot{g} \tag{5.53}$$

$$\text{Quotient Rule: } \frac{d}{dt}\frac{f}{g} = \frac{\dot{f}g - f\dot{g}}{g^2} \tag{5.54}$$

5.5 Simulation Results

5.5.1 Simulation Environment

In this section, we present a numerical example for three agents transporting a load. The control objective is transport the payload such that it will rotate by 90 degrees and

translate with the constant velocity v^d. The mass and inertia of each quadcopter are $m_i = 0.75$ kg and $I_i = diag\{0.0820, 0.0845, 0.1377\}\ kgm^2$ respectively. The mass and inertia of the payload are $M_c = 1.5$ and $I_c = diag\{0.0135, 0.0135, 0.0135\}\ kgm^2$ kg with a radius of 15 cm. For all of the results presented in this section we set $\psi_i^{des} = 0$ and $k = 1000$ N/m. We set $r_1 = [0, 0.15, 0]^T$, $r_2 = [0.15\sin(\pi/3), -0.15\cos(\pi/3), 0]^T$ and $r_3 = [-0.15\sin(\pi/3), -0.15\cos(\pi/3), 0]^T$. The $f_i^d(t)$'s are chosen online based on (5.21), where $f_{N(\Lambda)}^d$ for each agent is given by $R_c[0.0, -2.0, 0]^T$ N, $R_c[2.0, 1.0, 0]^T$ N, and $R_c[-2.0, 1.0, 0]^T$ N, respectively. As $R_c \to R_d$ and $\Omega_c \to 0$, $f_{N(\Lambda)}^d$ approaches a constant vector and $\dot{f}_{N(\Lambda)}^d$ approaches zero.

5.5.2 Quadcopter Attitude Controller

The gains for the attitude controller are given in Table 5.1.

K_1^i, K_2^i	K_3	K_{p1}^i, K_{p2}^i	K_{p3}^i	K_{d1}^i, K_{d2}^i	K_{d3}^i
21.93	18	4.65	3.76	0.19	0.15

Table 5.1: Attitude controller gains.

5.5.3 Numerical Examples

In this example we illustrate the time-scale separation between the fast dynamics (5.33) and (5.35) and the slow dynamics of ξ_i, Ω_c and q_v^e. We choose

$$v^d = \begin{bmatrix} 2.0, & 0.0, & 0 \end{bmatrix}^T, \quad q_v^d = \begin{bmatrix} 0, & 0, & \frac{1}{\sqrt{2}} \end{bmatrix}^T,$$
$$K_i = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 0.2 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}, \quad \Gamma_i = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}, \tag{5.55}$$
$$k_{q_v} = 0.1, \quad k_{\Omega_c} = 0.2.$$

Note that q_v^d corresponds to a desired vector quaternion for 90 degrees rotation on the $x-y$ plane. The product $\Gamma_i k$ is large compared with other control gains.

As shown in Fig. 5.1, 5.2, 5.3 and 5.4, the convergence of $\tilde{z}_i$, ξ_i, Ω_c and the attitude error q_v^e verifies that there exists a time-scale separation between the fast dynamics (5.33) and (5.35) and the slow dynamics of ξ_i, Ω_c and q_v^e.

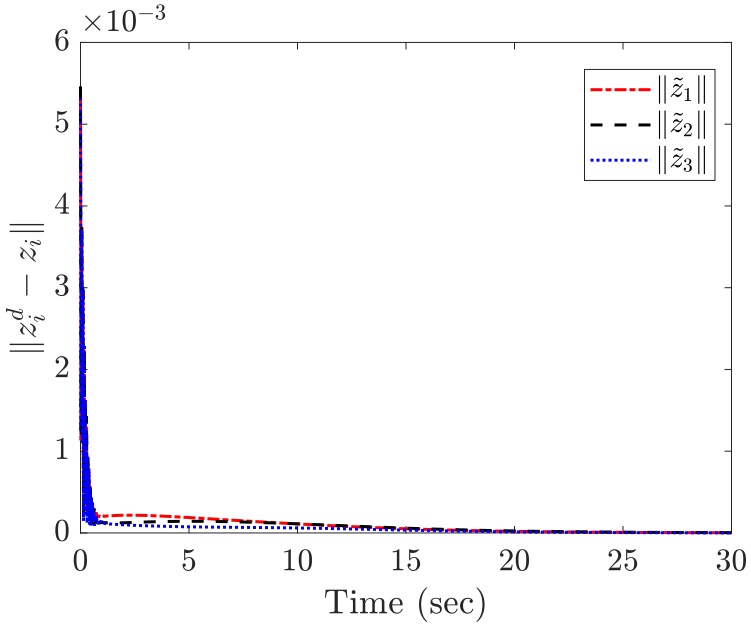


Figure 5.1: The magnitude of $\tilde{z}_i$ for the three agents.

Since singular perturbation and Lyapunov analysis can be conservative, we further tune the gains of the controllers to achieve improved transient performance. We increase K_i, k_{q_v} and k_{Ω_c} to

$$K_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad k_{q_v} = 2, \quad k_{\Omega_c} = 1, \tag{5.56}$$

while keeping other parameters the same as in the previous simulation.

Fig. 5.5 shows that the 3D trajectories of the agents and the payload. It can be observed that the payload is able to make a 90 degree turn and maintain the desired orientation throughout the simulation. Fig. 5.6 shows that the contact force is regulated to

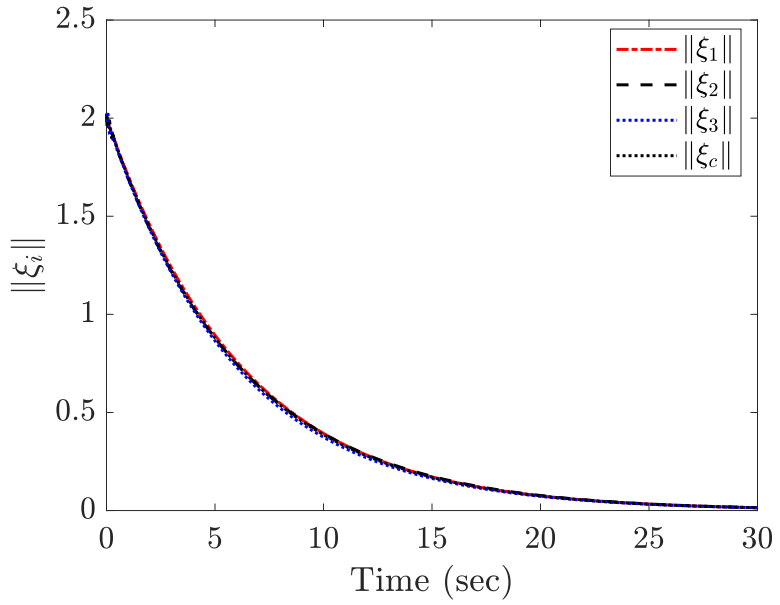


Figure 5.2: The magnitude of the velocity error ξ_i for the three agents and the payload.

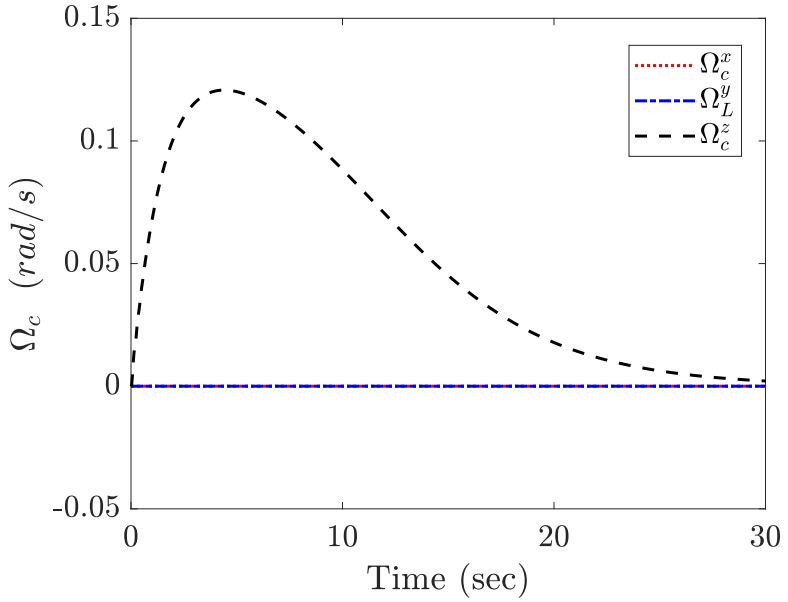


Figure 5.3: The angular velocity of the load Ω_c goes to zero.

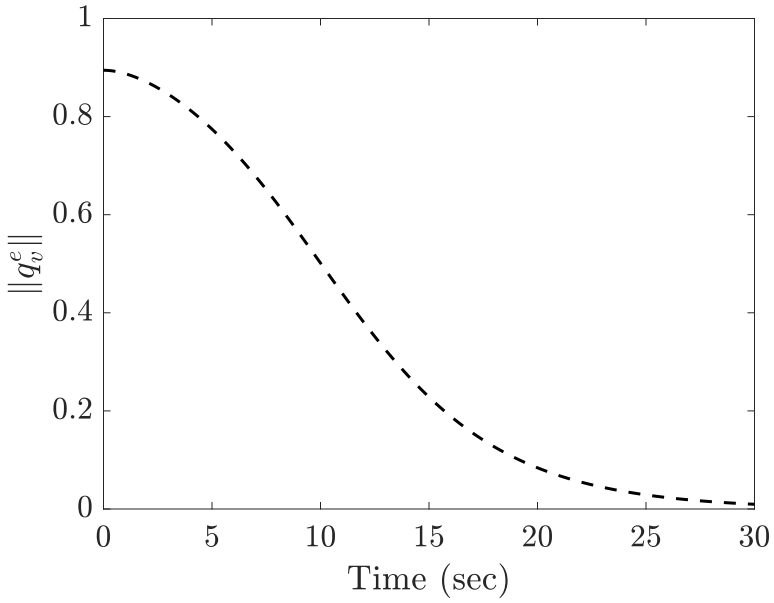


Figure 5.4: The magnitude of the orientation error $||q_v^e||$ for the payload.

the time varying setpoints. Fig. 5.7 shows that the velocity errors of agents and the payload converge to zero. Fig. 5.8 shows that the angular velocity of the payload converges to zero and Fig. 5.9 shows that $\|q_v^e\| \to 0$, which means that R_c converges to R_c^d. Fig. 5.10 and Fig. 5.11 show that the attitude controller tracks the desired attitude angles ϕ_i^{des} and θ_i^{des} generated from the force controller (3.6) and (3.7). Note that the convergence speed for all the variables are improved due to the increased gain parameters in (5.56).

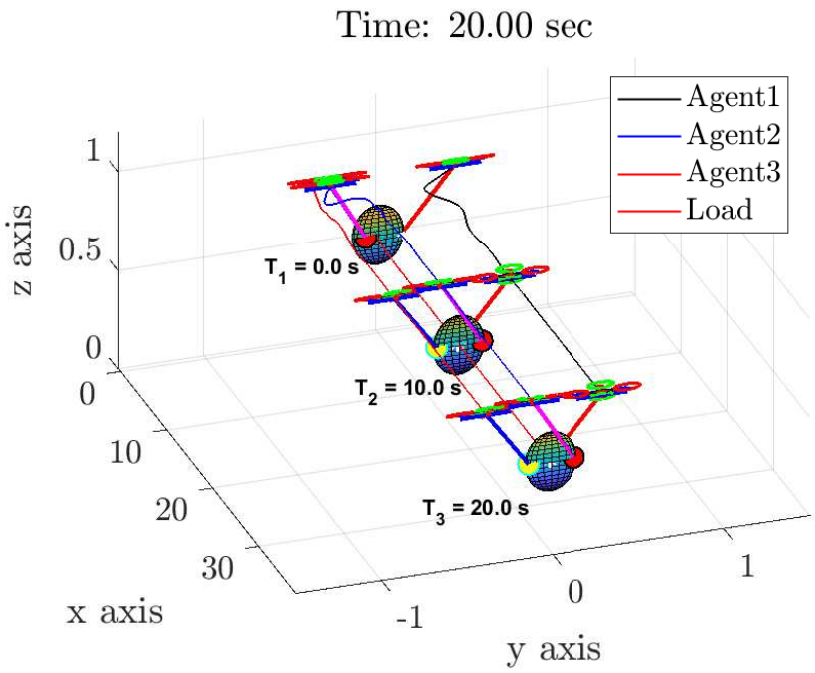


Figure 5.5: 3D position trajectory for three agents and the payload at different time steps for the first 20 seconds. The changes of the position of the red and the light blue dots confirm that the payload is able to achieve the desired rotation.

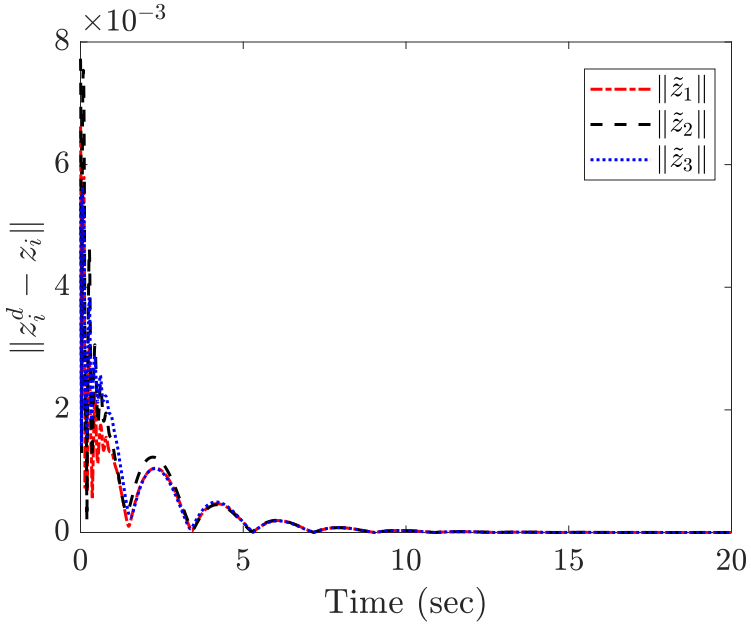


Figure 5.6: The magnitude of $\tilde{z}_i$ goes to zero for all three agents.

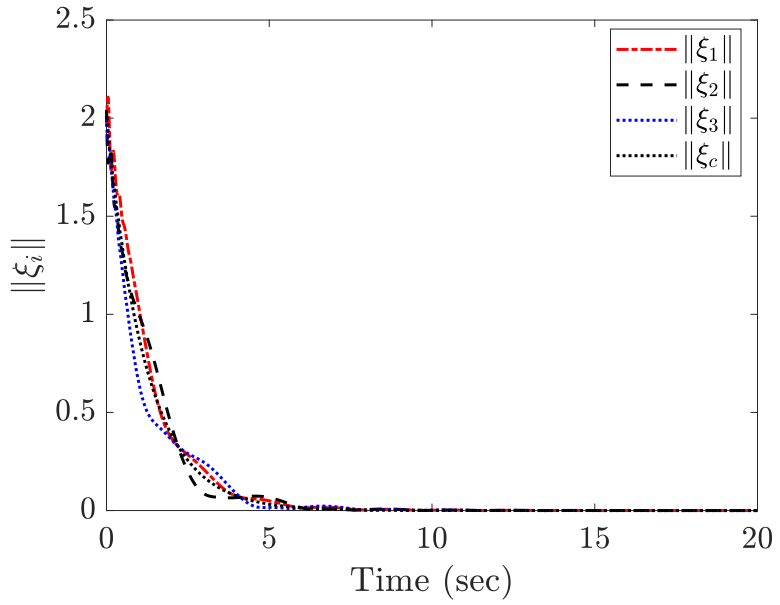


Figure 5.7: The magnitude of the velocity error ξ_i for the three agents and the payload.

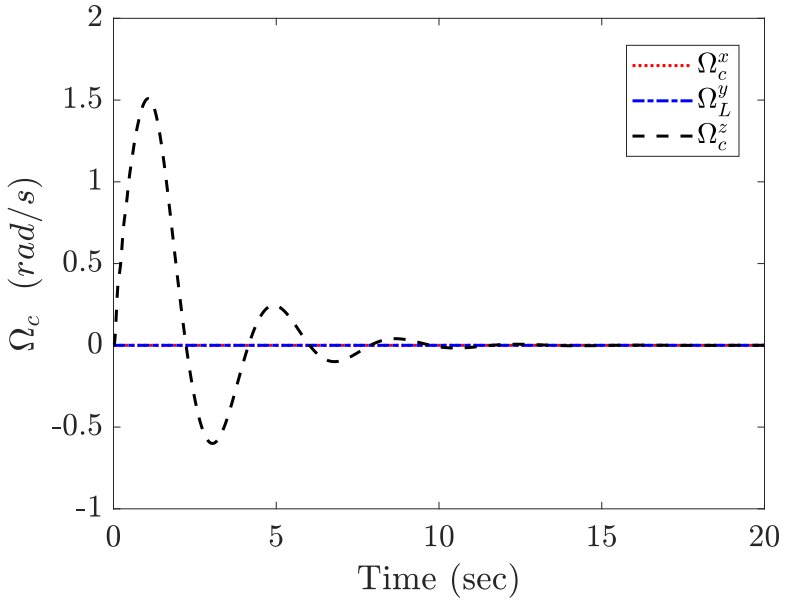


Figure 5.8: The angular velocity of the load Ω_c goes to zero.

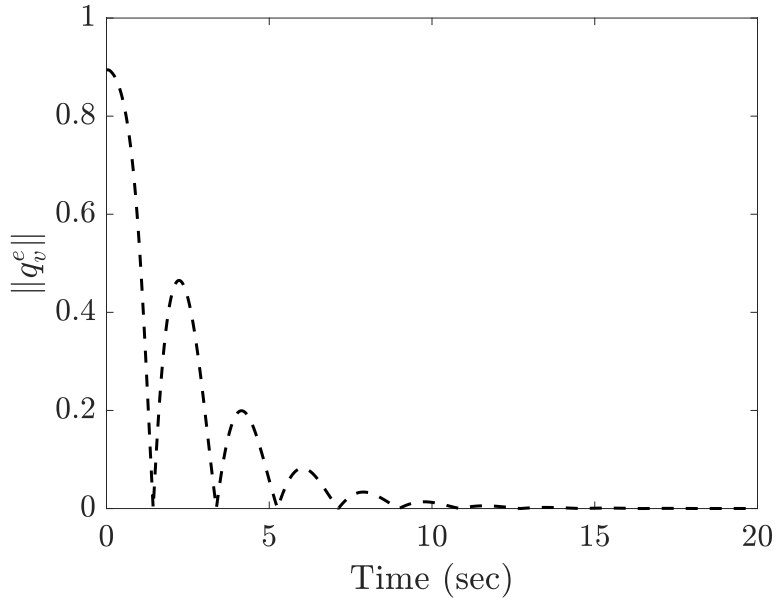


Figure 5.9: The magnitude of the orientation error $\|q_v^e\|$ for the payload.

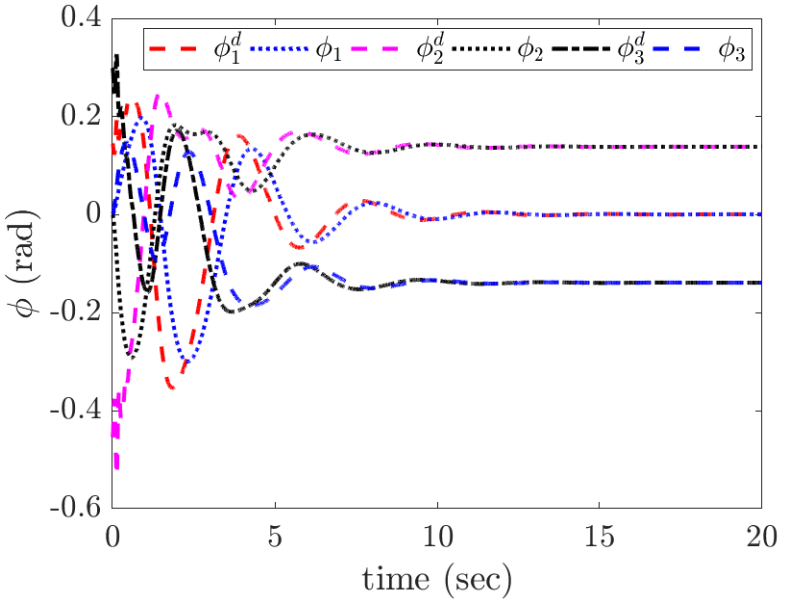


Figure 5.10: Desired and actual roll angles (ϕ) for three agents.

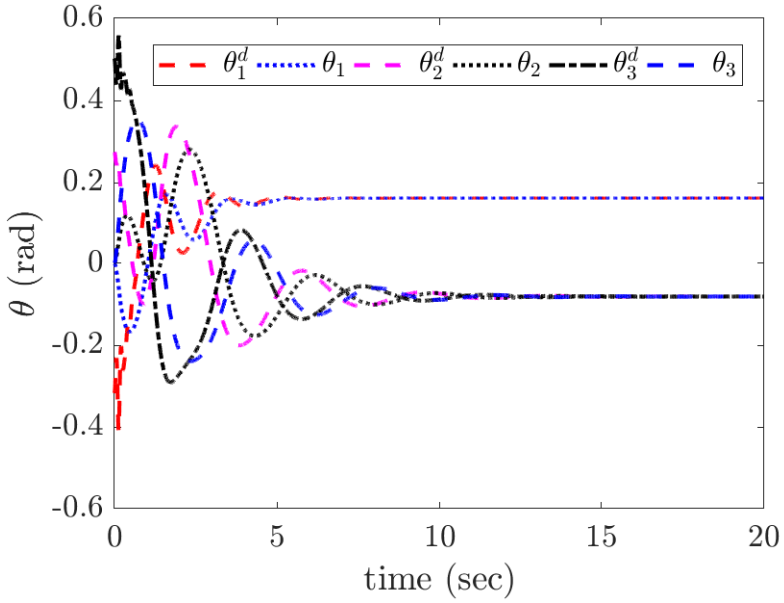


Figure 5.11: Desired and actual pitch angles (θ) for three agents.

Because of the complexity in computation of $\dot{f}_i^d$, we have also experimented with $\dot{f}_i^d$ set to 0 in (5.19). Our simulation results show that $\xi_i \to 0$, $\xi_c \to 0$, $\Omega_c \to 0$, $q_v^e \to 0$ and $\tilde{z}_i \to 0$ are still achieved. The simulation results are not included in this report and we plan to investigate more in our future work.

CHAPTER 6

FORCE CONTROL OF PAYLOAD WITH UNKNOWN MASS USING QUADCOPTERS WITH MULTI-LINK ROBOT

In this chapter, we tested our force control algorithms from Chapter 4 for a group of aerial manipulators (AM) transporting a flexible load. Each AM is a combination of an Unmanned Aerial Vehicle (UAV) with a two-degree-of-freedom robotic manipulator (RM) attached to it. This work is similar to the past work in [36] and we extend it to address cooperative transportation of a payload with an unknown mass. Specifically, we develop decentralized control and estimation algorithms that guarantees force regulation, velocity convergence and convergence of the estimate of the mass of the payload without explicit communication between the AMs. The algorithms are implemented at the kinematic level using the inverse kinematics of the AMs.

6.1 Payload Transportation with an Aerial Manipulator

6.1.1 Kinematics

Each aerial manipulator is a 2-DOF robotic arm mounted at the bottom of a quadrotor. Similar derivations for single aerial manipulator can be found in [43, 44]. Position of the center of mass of the quadrotor in the inertial frame Σ_I is given by, $p^I_{c,i} = \begin{bmatrix} x_{c,i} & y_{c,i} & z_{c,i} \end{bmatrix}^T$, orientation of the quadrotor is described by the triple of ZYX (yaw-pitch-roll) Euler angles, $\Phi_i = \begin{bmatrix} \psi_i & \theta_i & \phi_i \end{bmatrix}^T$ and joint angles of the two-DOF manipulator are given by, $\eta_{ij} = \begin{bmatrix} \eta_{i1} & \eta_{i2} \end{bmatrix}^T$ (see Fig 2.1). Both of the joint angles are defined about positive $x_{c,i}$ axis. The origin of frame for the load and each link of the manipula-

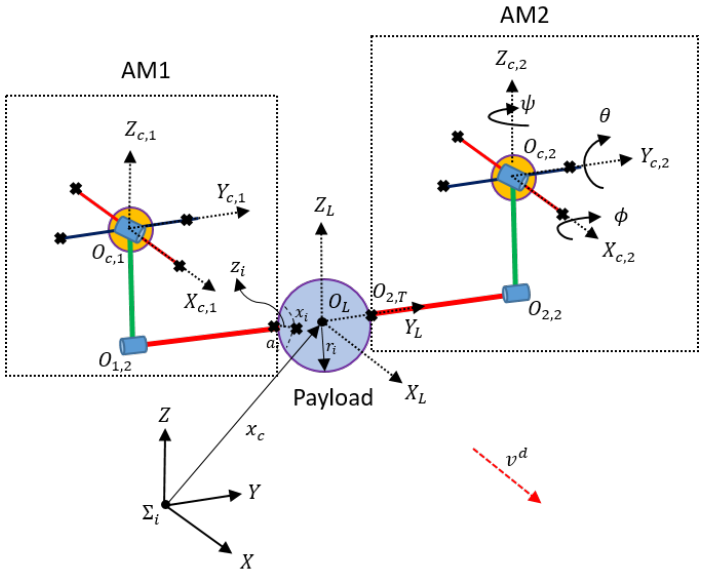


Figure 6.1: Two aerial manipulators transport a common flexible load. Note that body frame $O_{c,i}$ coincides with $O_{1,j}$ (not shown in the figure) and a_i is the initial position of agent i.

tor is placed at its center of mass making the axis coincident with the inertial axes. The vector containing all the generalized coordinates for agent i is given by

$$q_i = \begin{bmatrix} (p^I_{c,i})^T & \Phi_i^T & \eta_i^T \end{bmatrix}^T. \tag{6.1}$$

Let $\omega^I_{c,i}$ and $\omega^b_{i,c}$ be the angular velocity of the quadrotor in the inertial frame and the body-fixed frame respectively. We can map the time derivative of Euler angles $\dot{\Phi}_i$ to $\omega^I_{c,i}$ by a transformation matrix T_i. The following equalities hold:

$$\omega^I_{c,i} = R^I_{c,i}\omega^b_{c,i}, \tag{6.2}$$

$$\omega^I_{c,i} = T_i\dot{\Phi}_i, \tag{6.3}$$

$$\omega^b_{c,i} = (R^I_{c,i})^T T_i\dot{\Phi}_i = Q_i\dot{\Phi}_i, \tag{6.4}$$

where $R^I_{c,i} \in SO(3)$ is the rotation matrix between the body frame $O_{c,i}$ and the inertial frame Σ_I.

Let $p^b_{i,j}$ be the position of center of mass of link $j = 1,2$ of the i-th aerial manipulator

in the body-fixed frame $O_{c,i}$. The position p^I_{ij} of center of mass of link j is given by

$$p^I_{i,j} = p^I_{c,i} + R^I_{c,i} p^b_{i,j}. \tag{6.5}$$

The linear and angular velocity of each link of manipulator are related to the time derivatives of joint angles $\eta_{i,j}$ by

$$\dot{p}^b_{i,j} = J_{v,ij}\dot{\eta}_{i,j}, \tag{6.6}$$

$$\omega^b_{i,j} = J_{\omega,ij}\dot{\eta}_{i,j}, \tag{6.7}$$

where $J_{v,ij} \in \mathbb{R}^{3\times 2}$ and $J_{\omega,ij} \in \mathbb{R}^{3\times 2}$ are the linear and angular Jacobian of the manipulator in the body-fixed frame [45]. We can compute the linear and angular velocity of each link in the inertial frame from the following equations,

$$\dot{p}^I_{i,j} = \dot{p}^I_{c,i} + \dot{R}^I_{c,i} p^b_{i,j} + R^I_{c,i}\dot{p}^b_{i,j}, \tag{6.8}$$

$$\dot{p}^I_{i,j} = \dot{p}^I_{c,i} + \widehat{\omega}^I_{c,i} R^I_{c,i} p^b_{i,j} + R^I_{c,i} J_{v,ij}\dot{\eta}_{i,j}, \tag{6.9}$$

$$\omega^I_{i,j} = \omega^I_{c,i} + R^I_{c,i} J_{\omega,ij}\dot{\eta}_{i,j}, \tag{6.10}$$

where the operator$\widehat{\ }$converts a given vector $\omega = \begin{bmatrix}\omega_1 & \omega_2 & \omega_3\end{bmatrix}^T$, into a skew symmetric matrix as:

$$\widehat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \tag{6.11}$$

Let x^b_i be the position of the end-effector in the body-fixed frame $O_{c,i}$, which is given by

$$x^b_i = P^C_{c1} R^C_{c1} P_{12} + R^C_{c2} P_{2T}, \tag{6.12}$$

where P^C_{c1} is the vector representing the initial distance between the body frame of UAV and the body frame of the 1st link of the manipulator expressed in frame C. R^C_{c1}, and R^C_{c2} are the rotation matrices between the frames of the manipulator and its base frame. P_{12} is the initial distance between the first and the second link, P_{2T} is the initial distance between the second link and the end-effector both expressed in its body-frame.

The position of the end effector x_i^I in the inertial frame for the i-th aerial manipulator is given by,

$$x_i^I = p_{c,i}^I + R_{c,i}^I x_i^b. \tag{6.13}$$

The linear and angular velocities of the end effector can be computed as:

$$\dot{x}_i^I = \dot{p}_{c,i}^I + \dot{R}_{c,i}^I x_i^b + R_{c,i}^I \dot{x}_i^b, \tag{6.14}$$

$$\dot{x}_i^I = \dot{p}_{c,i}^I + \widehat{\omega}_{c,i}^I R_{c,i}^I x_i^b + R_{c,i}^I J_{v,ij} \dot{\eta}_{i,j}, \tag{6.15}$$

$$\omega_i^I = \omega_{c,i}^I + R_{c,i}^I J_{\omega,ij} \dot{\eta}_{i,j}. \tag{6.16}$$

Equations (6.13), (6.15) and (6.16) can be simplified as:

$$\dot{p}_{c,i}^I = \begin{bmatrix} I_{3\times3} & 0_{3\times3} & 0_{3\times2} \end{bmatrix} \dot{q}_i =: J_{t,b} \dot{q}_i, \tag{6.17}$$

$$\omega_{c,i}^I = \begin{bmatrix} 0_{3\times3} & T_i & 0_{3\times2} \end{bmatrix} \dot{q}_i =: J_{r,b} \dot{q}_i, \tag{6.18}$$

$$\dot{x}_i^I = \begin{bmatrix} I_{3\times3} & -\widehat{(R_{c,i}^I x_i^b)} T & R_{c,i}^I J_{v,ij} \end{bmatrix} \dot{q}_i =: J_{t,i} \dot{q}_i, \tag{6.19}$$

$$\omega_i^I = \begin{bmatrix} 0_{3\times3} & T_i & R_{c,i}^I J_{w,ij} \end{bmatrix} \dot{q}_i =: J_{r,i} \dot{q}_i. \tag{6.20}$$

Define $v_i = \begin{bmatrix} (\dot{p}_{c,i}^I)^T & (\omega_{c,i}^I)^T & (\dot{x}_i^I)^T & (\omega_i^I)^T \end{bmatrix}^T$, which contains the linear and angular velocities of the UAV and the end-effector of the i-th aerial manipulator. The time derivative of generalized joints vector q_i can be mapped to v_i by the following equation,

$$v_i = J_i \dot{q}_i, \tag{6.21}$$

where $J_i \in \mathbb{R}^{12\times q_i}$ is the geometric Jacobian of the system ([45]) whose elements is computed as:

$$J_i = \begin{bmatrix} J_{t,b} & J_{r,b} & J_{t,i} & J_{r,i} \end{bmatrix}^T. \tag{6.22}$$

We can represent the position/orientation of the end-effector as the minimal representation ([45]) and express the forward kinematics as:

$$x_i = k(q_i), \tag{6.23}$$

where $k(\cdot)$ is an $m \times 1$ vector function based on the system configuration and x_i is an $m \times 1$ state vector representing the system in minimal configuration. The time derivative of x_i and q_i can be related by,

$$\dot{x}_i = J_{a,i}\dot{q}_i, \tag{6.24}$$

where $J_{a,i} \in \mathbb{R}^{m \times n_{q_i}}$ is the analytical Jacobian of the system which can be derived via differentiation of $k(\cdot)$ ([45]). In this paper, we use the Cartesian position of the end-effector as a minimal representation, which gives us $J_{a,i} = J_{t,i}$.

6.1.2 Dynamics

We derive the dynamic model based on *Euler-Lagrange* formulation.

$$\frac{d}{dt}\frac{\partial \mathscr{L}}{\partial \dot{q}_i} - \frac{\partial \mathscr{L}}{\partial q} = \tau + \tau_{ext} \tag{6.25}$$

$$\mathscr{L} = \mathscr{K} - \mathscr{U} \tag{6.26}$$

where $\mathscr{L}$ is the *Lagrangian* of the system. $\mathscr{K}$ and $\mathscr{U}$ are the total kinetic and potential energy of the i-th aerial manipulator, τ represents generalized forces and τ_{ext} refers to any external disturbance applied to the system.

The total kinetic energy for the i-th aerial manipulator is given by the sum of individual contributions of the aerial vehicle and two links of the manipulator.

$$\mathscr{K}_i = \mathscr{K}_{b,i} + \sum_{i}^{2} \mathscr{K}_{ij} \tag{6.27}$$

where $\mathscr{K}_{b,i}$ is the kinetic energy of the quadrotor and $\mathscr{K}_{ij}$ is the kinetic energy of the link j of the i-th aerial manipulator.

The kinetic energy for the quadrotor $\mathscr{K}_{b,i}$ is given by

$$\mathscr{K}_{b,i} = \frac{1}{2}\dot{p}_{b,i}^T m_{b,i}\dot{p}_{b,i} + \frac{1}{2}\dot{\Phi}_i^T T^T R_{b,i} I_{b,i} R_{b,i}^T T \dot{\Phi}_i \tag{6.28}$$

where $m_{b,i}$ and $I_{b,i}$ is the mass and inertia matrix of the quadrotor respectively of the i-th aerial manipulator. Kinetic energy of each link can be expressed as follows

$$\mathscr{K}_{ij} = \frac{1}{2}\dot{p}_{ij}^T m_{ij}\dot{p}_{ij} + \frac{1}{2} w_{ij}^T (R_{b,i}R_{ij}^b) I_{ij} (R_{b,i}R_{ij}^b)^T w_{ij} \tag{6.29}$$

where R^b_{ij} is the rotation matrix between bodyframe of center of mass of j-th link and O_{bi}, while m_{ij} and I_{ij} is the mass and constant inertia matrix of link j respectively.

By considering (6.4), (6.8), (6.9), (6.10), (6.28) and (6.29) total kinetic energy of i-th aerial manipulator (6.27) can be expressed into following form

$$\mathcal{K}_i = \frac{1}{2}\dot{q}_i^T M_i(q_i)\dot{q}_i \tag{6.30}$$

where $M_i \in \mathbb{R}^{8\times 8}$ is a symmetric and positive definite inertia matrix of aerial manipulator i, which can be defined as a 3×3 block matrix with individual elements as follows:

$$M_{i_{11}} = \left(m_{b,i} + \sum_i^2 m_{ij}\right) I_{3\times 3} \tag{6.31}$$

$$M_{i_{22}} = Q_i^T I_{b,i} Q_i + \sum_i^2 \left(T_i^T (\widehat{R_{b,i} p^b_{ij}})^T m_{ij} (\widehat{R_{b,i} p^b_{ij}}) T_i + Q_i^T R^b_{ij} I_{ij} (R^b_{ij})^T Q_i\right) \tag{6.32}$$

$$M_{i_{33}} = \sum_i^2 \left(J_{v,i}^T m_{ij} J_{v,i} + J_{w,i}^T R^b_{ij} I_{ij} (R^b_{ij})^T J_{w,i}\right) \tag{6.33}$$

$$M_{i_{12}} = M_{i_{21}}^T = -\sum_i^2 \left(m_{ij} (\widehat{R_{b,i} p^b_{i,j}}) T_i\right) \tag{6.34}$$

$$M_{i_{13}} = M_{i_{13}}^T = \sum_i^2 \left(m_{ij} R_{b,i} J_{v,i}\right) \tag{6.35}$$

$$M_{i_{23}} = M_{i_{23}}^T = \sum_i^2 \left(Q_i^T R^b_{ij} I_{ij} (R^b_{ij})^T J_{w,i} - m_{ij} T_i^T (\widehat{R_{b,i} p^b_{ij}})^T R_{b,i} J_{v,i}\right) \tag{6.36}$$

The total potential energy for the i-th aerial manipulator is given by the sum of individual contributions of the aerial vehicle and two links of the manipulator.

$$\mathcal{U}_i = \mathcal{U}_{b,i} + \sum_i^2 \mathcal{U}_{ij} \tag{6.37}$$

The potential energy for the quadrotor $\mathcal{U}_{b,i}$ is given by

$$\mathcal{U}_{b,i} = m_{b,i} g e_3^T p_{b,i} \tag{6.38}$$

where g is acceleration due to gravity and e_3 is the unit vector $\begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$. Potential energy of each links is given by

$$\mathcal{U}_{ij} = m_{ij} g e_3^T (p_{b,i} + R_{b,i} p^b_{ij}) \tag{6.39}$$

The total potential energy for i-th aerial manipulator can be computed as

$$\mathcal{U}_i = m_{b,i} g e_3^T p_{b,i} + \sum_i^2 [m_{ij} g e_3^T (p_{b,i} + R_{b,i} p_{ij}^b)] \tag{6.40}$$

By substituting equations (6.26), (6.30), (6.40), into (6.25), the full dynamics of i-th aerial manipulator can be computed as

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + G_i(q_i) = \tau_i + \tau_{ext,i} \tag{6.41}$$

where $C_i(q_i, \dot{q}_i)$ is the Coriolis matrix and $G_i(q_i)$ is the vector containing gravity effects. The generic element of the Coriolis matrix $C_i(q_i, \dot{q}_i)$ is given by [45, 46],

$$c_{ij} = \sum_1^8 c_{ijk}\dot{q}_k \tag{6.42}$$

where the coefficients c_{ijk} is given by

$$c_{ijk} = \frac{1}{2}\Big(\frac{\partial m_{ij}}{\partial q_k} + \frac{\partial m_{ik}}{\partial q_j} - \frac{\partial m_{jk}}{\partial q_i}\Big) \tag{6.43}$$

which are also termed as *Christoffel symbols of the first type.* $G_i(q_i)$ is computed as

$$G_i(q_i) = \frac{\partial \mathcal{U}_i}{\partial q_i} \tag{6.44}$$

If we neglect the aerodynamic effects and consider low speed displacements [47] for the quadrotor model, the input τ mapped to actuation command as:

$$\tau_i = R_{f,i} N_i f_i = \Xi_i f_i \tag{6.45}$$

where $f_i = \begin{bmatrix} f_{q,i}^T & \tau_i^T \end{bmatrix}^T$ includes $f_{q,i} = \begin{bmatrix} f_{q,1} & f_{q,2} & f_{q,3} & f_{q,4} \end{bmatrix}^T$ as input forces actuated by the quadrotor motors and $\tau_i = \begin{bmatrix} \tau_{i,1} & \tau_{i,2} \end{bmatrix}^T$ for manipulator joint torques. $R_{f,i} = \text{diag}(R_{b,i}, Q^T, I_{n_i \times n_i}) \in \mathbb{R}^{6+n_i \times 6+n_i}$ and $N_i = \text{diag}(\Omega_i, I_{n_i \times n_i}) \in \mathbb{R}^{8 \times 4+n_i}$, in which

$$\Omega = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & d & 0 & -d \\ -d & 0 & d & 0 \\ c & -c & c & -c \end{bmatrix}$$

where d is the distance from a motor to the center of the vehicle and $c = k_m/k_f > 0$, k_m the drag coefficient and k_f the motor thrust coefficient which are obtained from the motor test data using a 6-axis force-torque sensor [17].

6.2 Control Objective

The control objective is to design F_i in a decentralized way such that all the agents and the payload converge to a constant velocity v^d and the contact force f_i is regulated to a setpoint f_i^d, $i = 1, \cdots, N$.

6.3 Payload Transport with an Unknown Mass

Rewrite the update law from Section 4.1

$$\dot{\widehat{M}}_{c,i} = \gamma_i (\dot{x}_i - v^d)^T e_3, \tag{6.46}$$

We modify the design in (2.9) as:

$$F_i = -\Gamma_i (\dot{x}_i - v^d) + \hat{f}_i^d, \tag{6.47}$$

and estimate the z-component of f_i^d as:

$$\hat{f}_{i,z}^d = \widehat{M}_{c,i} g. \tag{6.48}$$

The x and y components of $\hat{f}_i^d$ are the steady state values of the squeeze forces that are pre-designed to satisfy the following constraints

$$\sum_{i=1}^{N} \hat{f}_{i,x}^d = 0 \quad \text{and} \quad \sum_{i=1}^{N} \hat{f}_{i,y}^d = 0. \tag{6.49}$$

The dynamics (2.8) with the proposed control law (6.46), (6.47), takes the following form:

$$m_i \ddot{x}_i = -\Gamma_i (\dot{x}_i - v^d) + \hat{f}_i^d - f_i. \tag{6.50}$$

Theorem 6.1 *Consider the decentralized control law in* (6.47) *together with* (6.46) *and*

(6.48). *The equilibrium of* (2.7), (6.46), *and* (3.11) *given by*

$$\begin{aligned}\mathscr{E} = \{(\dot{x}_i, f_i, v^d, \widehat{M}_{c,i}) | \dot{x}_i = v^d, \quad \dot{x}_c = v^d, \\ \sum_{i=1}^{N} \widehat{M}_{c,i} = M_c, \quad \textit{and} \quad f_i = f_i^d\},\end{aligned} \tag{6.51}$$

is asymptotically stable. ■

Proof. The proof of the closed loop system is exactly same as in the one presented in Section 4.1. ■

6.4 Simulation Results

6.4.1 Simulation Environment

In this section, we present simulation results for two AMs transporting a load. Each quadrotor has a mass of 5 kg. Each link of the robotic manipulator has a length of 20 cm and a mass of 0.5 kg. The mass of the payload is 0.5 kg with a radius of 15 cm. All the joints of the manipulator are assumed to be spherical. The first joint rotates about the x axis and the second joint rotates about the y axis as shown in Fig. 2.1. We select $x_{c,i}$, $y_{c,i}$ and ϕ_i as state variables to be controlled for all the results presented in this section. We use a linear spring-force model to compute the contact force f_i in this simulation.

As a preliminary investigation, we implement the control in Chapter 4 at the kinematic level. This is equivalent to assuming that there exists a sufficiently-fast velocity tracking controller for each AM. To do so, we set right-hand side of (3.11) to 0 and solve for $\dot{x}_i$:

$$\dot{x}_i = K_i(\hat{f}_i^d - f_i) + v^d, \quad i = 1, ..., N, \tag{6.52}$$

where, $K_i > 0$ and x_i is the position of the end-effector of i-th the aerial manipulator.

The end-effector velocity $\dot{x}_i$ is computed from (6.52) and transformed to the generalized velocities using (6.19). Because of the redundancy in the system, we use the partial-inverse of the Jacobian matrix to calculate the joint velocities for 3 desired states.

An example to calculate the joint velocities for 3 states, $x_{c,i}$, $y_{c,i}$ and ϕ_i from the partial-inverse of the Jacobian is shown below. Let

$$J_{(t,i)s} = \begin{bmatrix} J_{t,i}(:,1), & J_{t,i}(:,2), & J_{t,i}(:,6) \end{bmatrix} \tag{6.53}$$

be the partial Jacobian matrix for the $x_{c,i}$, $y_{c,i}$ and ϕ_i states. Then the corresponding joint velocities are computed as

$$\begin{aligned} \dot{q}_{(i)prtl} &= \left(J_{(t,i)s}\right)^{-1} \dot{x}_i, \\ \dot{q}_i &= \begin{bmatrix} \dot{q}_{(i)prtl}(1), & \dot{q}_{(i)prtl}(2), & 0, & 0, & 0, & \dot{q}_{(i)prtl}(3), & 0, & 0 \end{bmatrix}^T. \end{aligned} \tag{6.54}$$

Note that k is not used in the controller (6.52).

6.4.2 Unknown Payload with a Predesigned v^d

We set $v^d = \begin{bmatrix} 0.2 & 0 & 0 \end{bmatrix}^T$, $\widehat{M}_{c,i} = 0.1$ kg, $K_i = 0.1$, $k = 100$ N/m, $f_1^d = \begin{bmatrix} 0 & 2.0 & 0.5\hat{M}_c g \end{bmatrix}^T$ and $f_2^d = \begin{bmatrix} 0 & -2.0 & 0.5\hat{M}_c g \end{bmatrix}^T$. The initial conditions of the two AMs are set as:

$$q_1 = \begin{bmatrix} 0.1 & -0.35 & 0.7 & 0 & 0 & -0.1 & 0 & -\pi/2 \end{bmatrix}^T,$$

$$q_2 = \begin{bmatrix} 0 & 0.35 & 0.68 & 0 & 0 & 0.1 & 0 & \pi/2 \end{bmatrix}^T.$$

We observe from Fig. 6.2 and 6.3 that the velocities of the agents and the payload converge to v^d. The update law from (6.46) successfully recovers the actual mass of the payload as seen in Fig. 6.4. We observe from Fig. 6.5 that the squeeze force is regulated to the setpoint f_i^d.

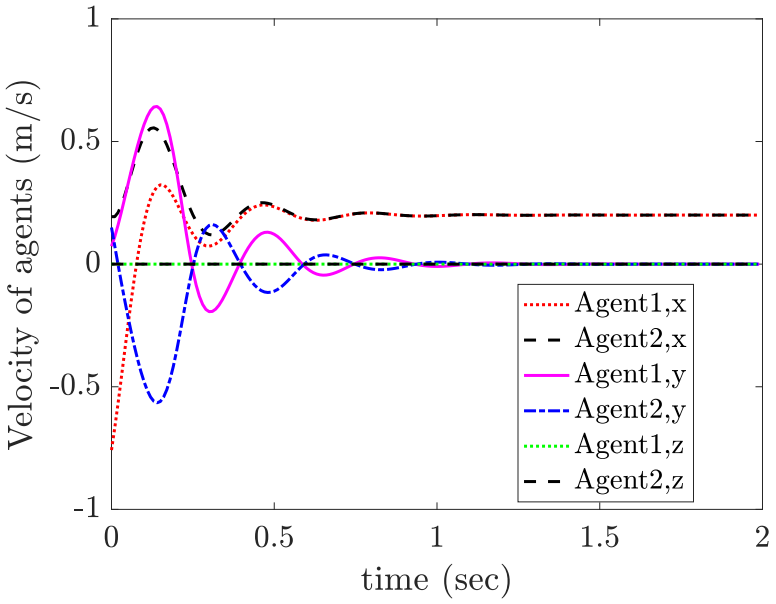


Figure 6.2: Linear velocities for both agents. The x component of the velocity converges to 0.2 m/s and the rest converge to zero.

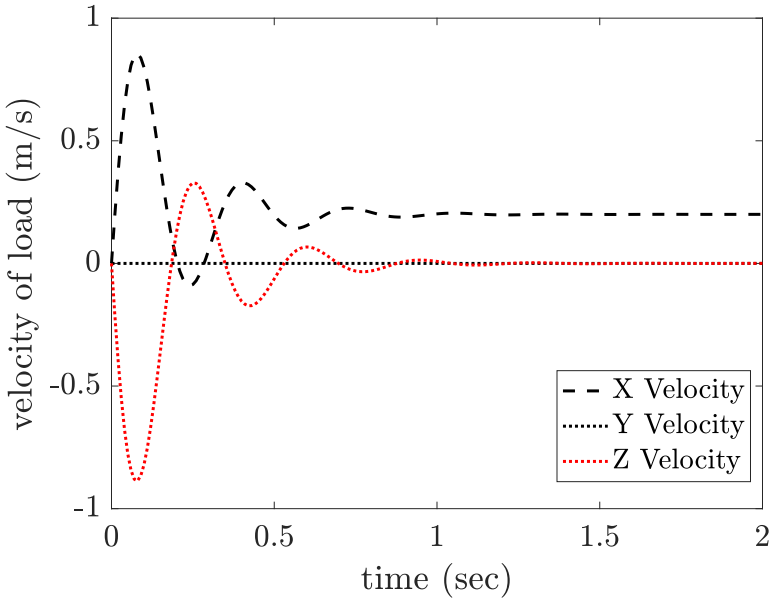


Figure 6.3: Linear velocity for the payload. The x component of the velocity converges to 0.2 m/s and the rest converge to zero.

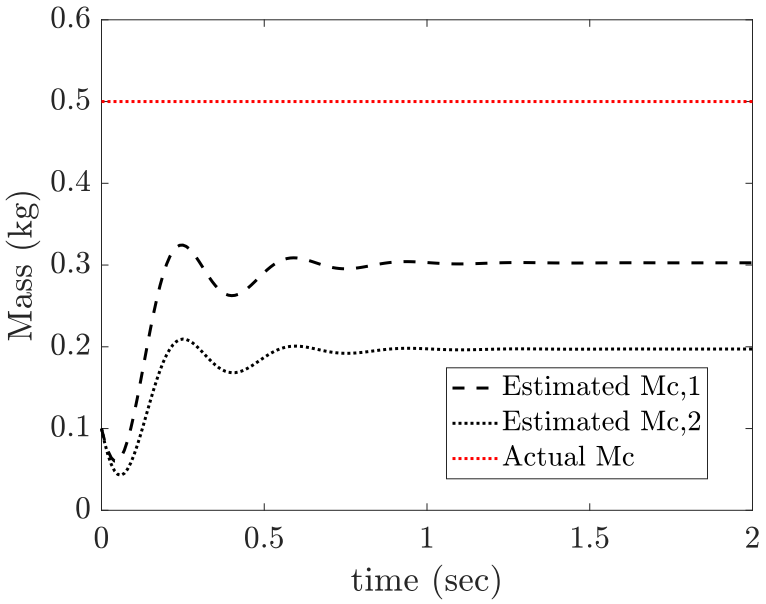


Figure 6.4: Estimation of the unknown mass. The sum of the individual estimates converges to the actual mass of the load.

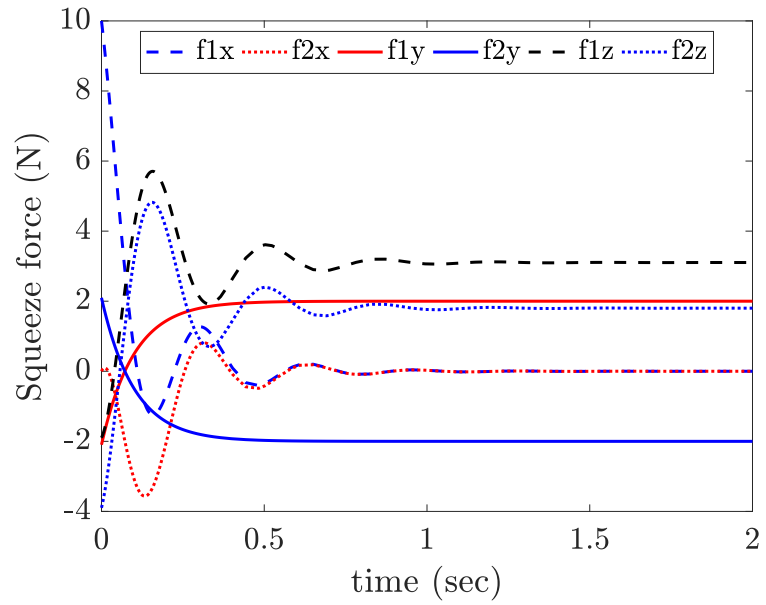


Figure 6.5: Contact forces for both AMs in all 3 direction. The forces in the x direction converge to zero. In the y-direction, they converge to 2.0 N and -2.0 N, respectively. In the z direction, the sum of the force is equal to the weight of the payload.

6.4.3 Unknown Payload with a Time Varying v^d

In this case study, we test the robustness of our controllers to time varying velocities. We keep the other parameters to be the same as in the previous case and change the desired velocity of the agents to

$$v^d(t) = \begin{cases} \begin{bmatrix} 2.0, & 0, & 0 \end{bmatrix}^T, & \text{t} \leq 2 \text{ s}, \\ \begin{bmatrix} 2-2\sin(t-2), & 2\sin(t-2), & 0 \end{bmatrix}^T, & 2 < \text{t} \leq 4 \text{ s}, \\ \begin{bmatrix} 0, & 2.0, & 0 \end{bmatrix}^T, & \text{t} > 4.0 \text{ s}. \end{cases}$$

Fig. 6.6 and 6.7 show the trajectories of the agents and the payload in 2D and 3D, respectively. We also observe from Fig. 6.8 and 6.9 that the velocities of the agents and the payload in both x and y directions closely track v^d. The contact force is also regulated close to the setpoint f_i^d as shown in Fig. 6.11.

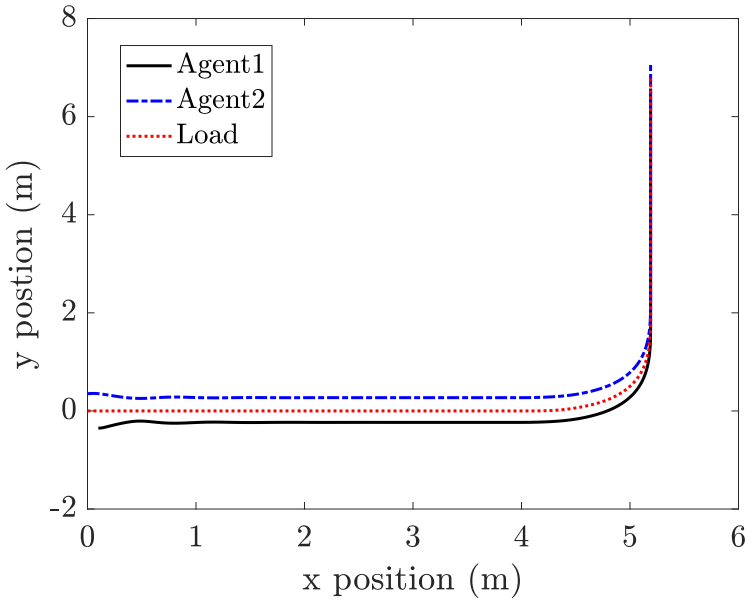


Figure 6.6: X and Y position trajectories for both agents (time varying v^d). Both agents travel in the x-direction until 2 seconds and make a smooth turn towards the y-direction after 2 seconds.

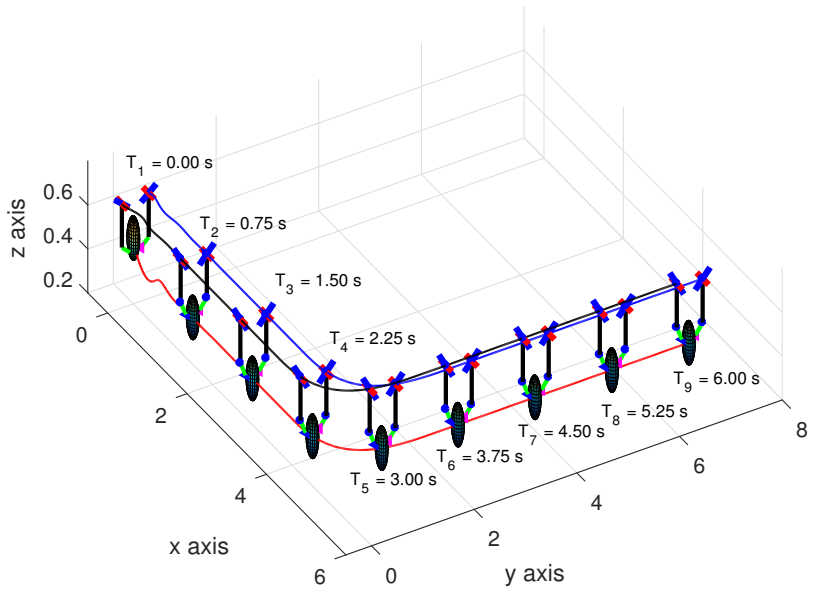


Figure 6.7: 3D position trajectory for both agents and the payload at different time interval. Note that all 3 agents stay at an equal distance and will never collide. They approach the y-axis by being aligned in a straight line as shown in Fig. 6.6. Notice that if the quadrotors are at the same height, they must keep a gap and the manipulator must be large enough to avoid their blades clashing.

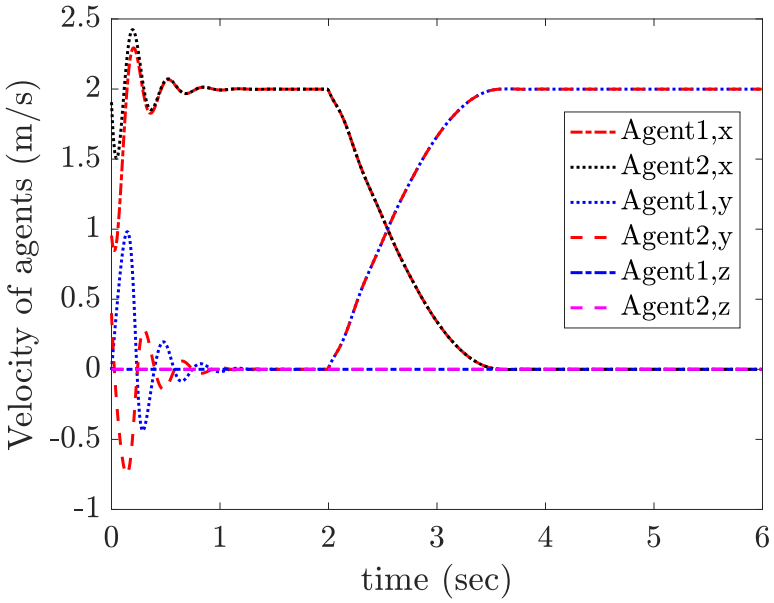


Figure 6.8: Velocities of the 2 agents in all 3 directions. The x and y velocities are smooth during transition and the z velocity is zero.

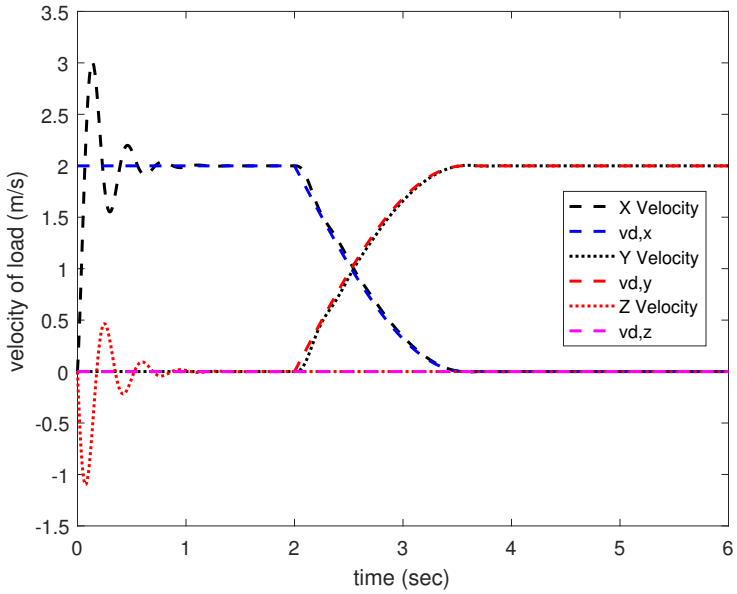


Figure 6.9: Desired and actual velocities of the load in all 3 directions.

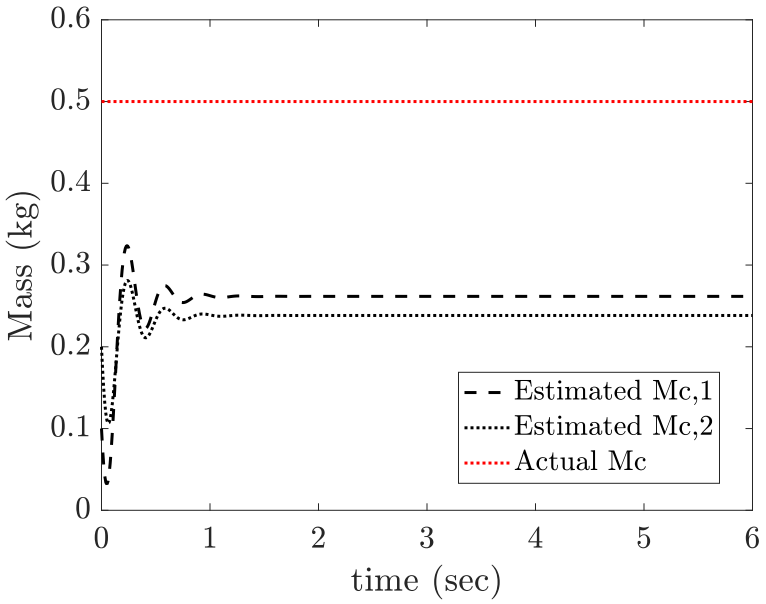


Figure 6.10: Estimation of the unknown mass. The sum of individual estimate converges to the actual mass of the load, which is 0.5 kg.

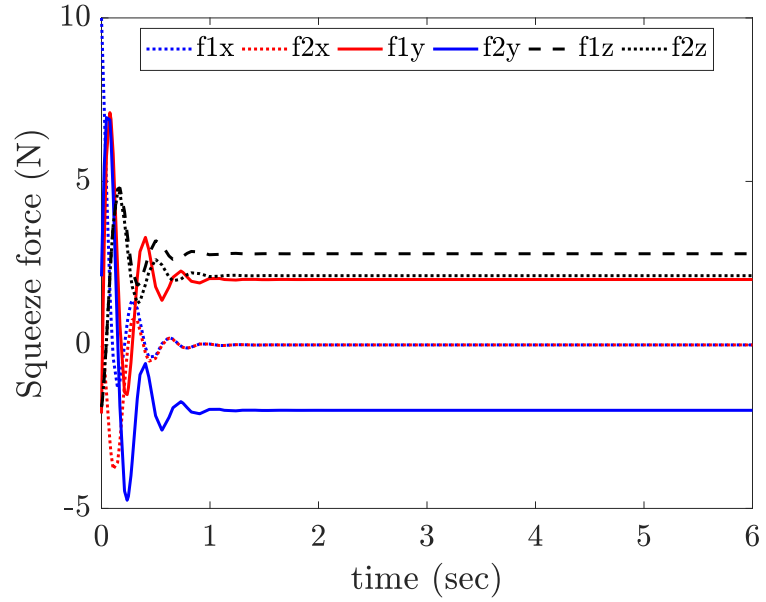


Figure 6.11: Contact forces of both agents in all 3 direction. The forces in the x direction converge to zero. In the y-direction, they converge to 2.0 N and -2.0 N, respectively. In the z direction, the sum of the force is equal to the weight of the payload.

CHAPTER 7

EXPERIMENTAL RESULTS

In this chapter we present our preliminary experimental results.

7.1 Mechanical Design of the Test Bed

There are plenty of quadcopters and robotic arm available. However due to expensive SDK and propriety controllers, we develop our own test bed from custom made quadcopter and 3D printed light weight robotic arm.

7.1.1 Quadcopter

Figure 7.1: Custom made quadcopter in the lab.

7.1.2 Robotic Manipulator Design

We decided to use [48] file and significanty modified it to fit our needs for the quadcopter. This design is simple and can be printed using 3D printers. The motors and

other parts are also inexpensive. The robotic arm is composed a two links which can

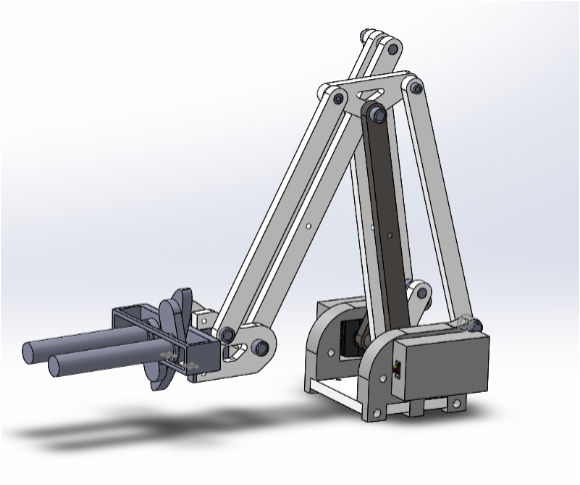


Figure 7.2: CAD model of the tow link robotic arm.

be actuated using high torque servo motors that can be mounted on the bottom of the arm. Another light weight servo motor is mounted close the gripper to enable dynamic gripping.

7.2 Computation Hardware

7.2.1 Microprocessor

We use an Arduino Nano for the processing. The Arduino platform excels at basic input/output operations, such as inputting sensor information and outputting servo commands. It is also has enough computational power to enable the implementation of low-level algorithms. The Arduino is only used as a low-level controller. It inputs user command via serial input, processes it with various sensory information, and outputs the desired angular position of each servo.

7.2.2 Communications

The microprocessor communicates with high-level controllers via serial input. Upon detecting a new character in the serial buffer, the program enters a while() loop to con-

tinuously add incoming characters to char input[]. Once the end of the stream is detected, or a newline character is read, the program proceeds to process input[]. Each character is checked against a menu of options in a switch-case statement. If the character is found, the statement either the remaining characters of input[] are parsed to integers and return for further processing.

7.2.3 Motors

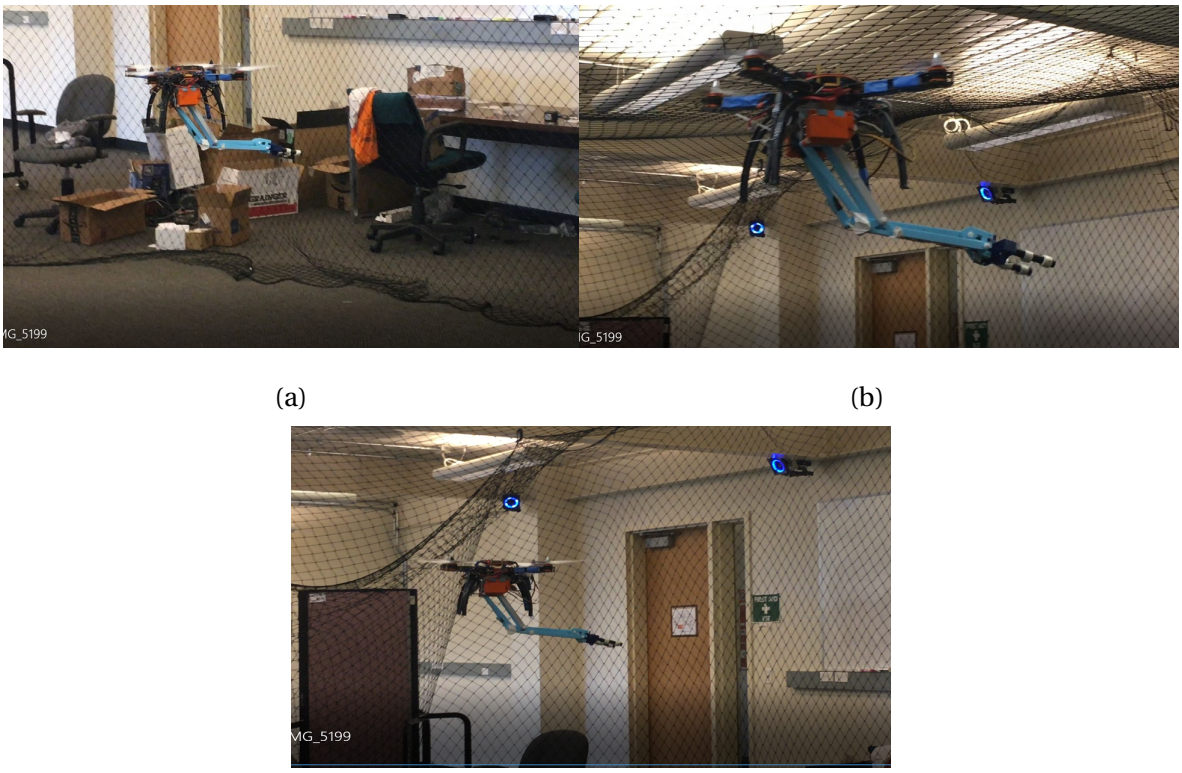
We used two Hitec HS-77BB servo motor in the linkage. The Hitec HS-77BB is a 3-pole brushed servo. At only 35.0 grams and dimensions of 44.0 mm x 23.0 mm x 25.0 mm, the HS-77BB operates at 4.8 – 6.0 V DC and has a maximum torque range of 4.4âĂŞ5.5 kg/cm [49]. When compared with servos of similar volume and mass (such as the SPMSA6050), the HS-77BB has above average torque.

7.3 Initial Integration and test

Figure 7.3: Custom made aerial manipulator.

We also tested the viability of our system by manfully flying the aerial manipulator in

the lab. Currently we are exploring the aerial gripping with a single AM and eventually plan to extend this to multiple AMs.



(a) (b)

(c)

Figure 7.4: Different instances of initial aerial manipulation test.

CHAPTER 8

CONCLUSION AND FUTURE WORK

In this thesis, we considered multiple quadcopters transporting a payload. We developed a force and orientation control algorithm to transport the payload. We developed adaptive decentralized control law for transporting the payload of unknown mass without explicit communication between the agents. Our controller ensures that all quadcopters asymptotically converge to a constant reference velocity. It also ensures that all of the forces applied to the payload converges to desired set-points. Desired thrusts and attitude angles are computed from the control algorithms and a low-level PD controller is implemented to track the desired commands for each quadcopter. The sum of the estimates of the unknown mass from all the agents converges to the true mass.

We also developed a consensus algorithm based on connected graphs to ensure that each agent gets an equal share of the payload mass. Furthermore, we developed orientation control algorithm that guarantees attitude stabilization of the payload. In particular, we developed time varying force setpoints to enforce attitude regulation.

Future work will involve relaxation of the required information in the algorithm and experimental validation of the proposed design. We plan to design and implement robust controllers for disturbance rejection in the presence of wind gusts. Furthermore, we plan to develop a complete aerial manipulation system in the lab.

REFERENCES

[1] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms In MATLAB® Second, Completely Revised.* Springer, 2017, vol. 118.

[2] (2018) Amazon Prime Air. [Online]. Available: https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011

[3] S. Thapa, H. Bai, and J. Acosta, "Cooperative aerial load transport with force control," *IFAC-PapersOnLine*, vol. 51, no. 12, pp. 38–43, 2018.

[4] S. Thapa, H. Bai, and J. A. Acosta, "Force control in cooperative aerial manipulation," in *Unmanned Aircraft Systems (ICUAS), 2018 International Conference on*. IEEE, 2018, pp. 1302–1309.

[5] ——, "Cooperative aerial manipulation with decentralized adaptive force control," *Journal of Intelligent & Robotic Systems (in review)*.

[6] ——, "Cooperative aerial load transport with attitude stabilization," in *2019 American Control Conference (in review)*. IEEE.

[7] K. Sreenath and V. Kumar, "Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots," in *Robotics: Science and Systems (RSS)*, 2013.

[8] I. Maza, K. Kondak, M. Bernard, and A. Ollero, "Multi-UAV cooperation and control for load transportation and deployment," *Journal of Intelligent and Robotic Systems*, vol. 57, no. 1, p. 417, Aug 2009.

[9] J. Fink, N. Michael, S. Kim, and V. Kumar, "Planning and control for cooperative manipulation and transportation with aerial robots," in *Robotics Research*. Springer Berlin Heidelberg, 2011, pp. 643–659.

[10] H. Lee, H. Kim, and H. J. Kim, "Path planning and control of multiple aerial manipulators for a cooperative transportation," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Sept 2015, pp. 2386–2391.

[11] T. Lee, "Geometric control of quadrotor uavs transporting a cable-suspended rigid body," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 1, pp. 255–264, 2018.

[12] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, Sept 2011.

[13] C. Meissen, K. Klausen, M. Arcak, T. I. Fossen, and A. Packard, "Passivity-based formation control for uavs with a suspended load," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 13 150–13 155, 2017.

[14] V. Lippiello and F. Ruggiero, "Exploiting redundancy in cartesian impedance control of uavs equipped with a robotic arm," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 3768–3773.

[15] H. Lee, S. Kim, and H. J. Kim, "Control of an aerial manipulator using on-line parameter estimator for an unknown payload," in *Automation Science and Engineering (CASE), IEEE International Conference on*, 2015, pp. 316–321.

[16] H. Lee, H. Kim, and H. J. Kim, "Planning and control for collision-free cooperative aerial transportation," *IEEE Transactions on Automation Science and Engineering*, 2016.

[17] H. Lee and H. J. Kim, "Estimation, control, and planning for autonomous aerial transportation," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 4, pp. 3369–3379, 2017.

[18] T. Lee, K. Sreenath, and V. Kumar, "Geometric control of cooperating multiple quadrotor UAVs with a suspended payload," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, Dec 2013, pp. 5510–5515.

[19] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, "Cooperative grasping and transport using multiple quadrotors," in *Distributed Autonomous Robotic Systems*. Springer, 2013, pp. 545–558.

[20] S. Kim, H. Seo, J. Shin, and H. J. Kim, "Cooperative aerial manipulation using multirotors with multi-dof robotic arms," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 2, pp. 702–713, 2018.

[21] J. Acosta, M. Sanchez, and A. Ollero, "Robust control of underactuated aerial manipulators via IDA-PBC," in *Decision and Control (CDC), IEEE 53rd Annual Conference on*, Dec 2014, pp. 673–678.

[22] J. Acosta, C. de Cos, and A. Ollero, "A robust decentralised strategy for multi-task control of unmanned aerial systems. application on underactuated aerial manipulator," in *Unmanned Aircraft Systems (ICUAS), International Conference on*, June 2016, pp. 1075–1084.

[23] Z. Wang and M. Schwager, "Multi-robot manipulation without communication," in *Distributed Autonomous Robotic Systems*. Springer, 2016, pp. 135–149.

[24] H. Farivarnejad and S. Berman, "Stability and convergence analysis of a decentralized proportional-integral control strategy for collective transport," in *2018 Annual American Control Conference (ACC)*, June 2018, pp. 2794–2801.

[25] P. Culbertson and M. Schwager, "Decentralized adaptive control for collaborative manipulation," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, 2018.

[26] Z. Wang, S. Singh, M. Pavone, and M. Schwager, "Cooperative object transport in 3D with multiple quadrotors using no peer communication," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1064–1071, 2018.

[27] R. Ritz and R. D'Andrea, "Carrying a flexible payload with multiple flying vehicles," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 3465–3471.

[28] C. F. Babbs, "Biomechanics of heading a soccer ball: implications for player safety," *The Scientific World Journal*, vol. 1, pp. 281–322, 2001.

[29] M. Nagurka and S. Huang, "A mass-spring-damper model of a bouncing ball," in *Proceedings of the 2004 American Control Conference*, vol. 1, June 2004, pp. 499–504 vol.1.

[30] R. Beard, "Quadrotor dynamics and control rev 0.1," 2008.

[31] S. Allison, H. Bai, and B. Jayaraman, "Modeling trajectory performance of quadrotors under wind disturbances," in *AIAA Information Systems-AIAA Infotech@ Aerospace*, Jan 2018, p. 1237.

[32] T. Lee, M. Leoky, and N. H. McClamroch, "Geometric tracking control of a quadrotor uav on se (3)," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 5420–5425.

[33] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The grasp multiple micro-uav testbed," *IEEE Robot. Automat. Mag.*, vol. 17, no. 3, pp. 56–65, 2010.

[34] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2011, pp. 2520–2525.

[35] K. Sreenath, T. Lee, and V. Kumar, "Geometric control and differential flatness of a quadrotor {UAV} with a cable-suspended load." Citeseer.

[36] H. Bai and J. T. Wen, "Cooperative load transport: A formation-control perspective," *IEEE Transactions on Robotics*, vol. 26, no. 4, pp. 742–750, 2010.

[37] H. K. Khalil, *Noninear Systems*. Prentice Hall, 1996.

[38] H. Bai, M. Arcak, and J. Wen, *Cooperative Control Design: A Systematic, Passivity-Based Approach*. Springer Science & Business Media, 2011.

[39] N. Biggs, N. L. Biggs, and E. N. Biggs, *Algebraic graph theory*. Cambridge university press, 1993, vol. 67.

[40] J. T. Wen and K. Kreutz-Delgado, "The attitude control problem," *IEEE Transactions on Automatic control*, vol. 36, no. 10, pp. 1148–1162, 1991.

[41] O.-E. Fjellstad and T. I. Fossen, "Comments on "The attitude control problem"," *IEEE Transactions on Automatic Control*, vol. 39, no. 3, pp. 699–700, 1994.

[42] A. Levant, "Robust exact differentiation via sliding mode technique," *automatica*, vol. 34, no. 3, pp. 379–384, 1998.

[43] V. Lippiello and F. Ruggiero, "Cartesian impedance control of a uav with a robotic arm," *IFAC Proceedings Volumes*, vol. 45, no. 22, pp. 704–709, 2012.

[44] S. Kim, S. Choi, and H. J. Kim, "Aerial manipulation using a quadrotor with a two dof robotic arm," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 4990–4995.

[45] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. Wiley New York, 2006.

[46] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[47] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, and D. Nakazawa, *Autonomous Flying Robots*. Springer Japan, 2010.

[48] Grabcad. (2018). [Online]. Available: https://grabcad.com/library/sea-reach-a-scalable-deep-sea-robotic-arm-1

[49] ServoDatavase. (2018). [Online]. Available: https://servodatabase.com/servo/hitec/hs-77bb

VITA

Sandesh Thapa

Candidate for the Degree of

Master of Science

Thesis: COOPERATIVE AERIAL MANIPULATION WITH FORCE CONTROL AND ATTITUDE STABILIZATION

Major Field: Mechanical & Aerospace Engineering

Biographical:

Education:

Completed the requirements for the Master of Science in Mechanical & Aerospace Engineering at Oklahoma State University, Stillwater, Oklahoma in December, 2018.

Completed the requirements for the Bachelor of Science in Mechanical Engineering at McNeese State University, Lake Charles, Louisiana in 2015.

Publications:

Thapa S., Bai H. and Acosta J.A. Cooperative Aerial Manipulation with Decentralized Adaptive Force Control. Journal of Intelligent & Robotic Systems (in review).

Thapa S., Bai H. and Acosta J.A. Cooperative Aerial Load Transport with Attitude Stabilization. 2018 American Control Conference (ACC) (submitted).

Thapa S., Bai H. and Acosta J.A. Force Control in Cooperative Aerial Manipulation. IEEE International Conference on Unmanned Aircraft Systems (ICUAS), June, 2018.

Thapa S., Bai H. and Acosta J.A. Cooperative Aerial Load Transport with Force Control. IFAC Workshop on Networked & Autonomous Air & Space Systems, June, 2018.