# Improving Leader-Follower Formation Control Performance for Quadrotors

By

Wesam M. Jasim Alrawi

A thesis submitted for the degree of ***Doctor of Philosophy***
School of Computer Science and Electronic Engineering
University of Essex
October 2016

# *Abstract*

This thesis aims to improve the leader-follower team formation flight performance of Unmanned Aerial Vehicles (UAVs) by applying nonlinear robust and optimal techniques, in particular the nonlinear $H_\infty$ and the iterative Linear Quadratic Regulator (iLQR), to stabilisation, path tracking and leader-follower team formation control problems.

Existing solutions for stabilisation, path tracking and leader-follower team formation control have addressed a linear or nonlinear control technique for a linearised system with limited disturbance consideration, or for a nonlinear system with an obstacle-free environment. To cover part of this area of research, in this thesis, some nonlinear terms were included in the quadrotors' dynamic model, and external disturbance and model parameter uncertainties were considered.

Five different controllers were developed. The first and the second controllers, the nonlinear suboptimal $H_\infty$ control technique and the Integral Backstepping (IBS) controller, were based on Lyapunov theory. The $H_\infty$ controller was developed with consideration of external disturbance and model parameter uncertainties. These two controllers were compared for path tracking and leader-follower team formation control. The third controller was the Proportional Derivative square ($PD^2$), which was applied for attitude control and compared with the $H_\infty$ controller. The fourth and the fifth controllers were the Linear Quadratic Regulator (LQR) control technique and the optimal iLQR, which was developed based on the LQR control technique. These were applied for attitude, path tracking and team formation control and there results were compared.

Two features regarding the choice of the control technique were addressed: stability and robustness on the one hand, which were guaranteed using the $H_\infty$ control technique as the disturbance is inherent in its mathematical model, and the improvement in the performance optimisation on the other, which was achieved using the iLQR technique as it is based on the optimal LQR control technique. Moreover, one loop control scheme was used to control each vehicle when these controllers were implemented and a distributed control scheme was proposed for the leader-follower team formation problem.

Each of the above mentioned controllers was tested and verified in simulation for different predefined paths. Then only the nonlinear $H_\infty$ controller was tested in both simulation and real vehicles experiments.

# *Publication List*

The writing of this thesis is based on the papers which have been published in the past years:

1. W. Jasim and D. Gu, “$H_\infty$ control for quadrotor attitude stabilization,” in Control (CONTROL), 2014 UKACC International Conference on, pp. 19–24, July 2014.

2. W. Jasim and D. Gu, “$H_\infty$ path tracking control for quadrotors based on quaternion representation,” in Advances in Autonomous Robotics Systems (M. Mistry, A. Leonardis, M. Witkowski, and C. Melhuish, eds.), vol. 8717 of Lecture Notes in Computer Science, pp. 72–84, Springer International Publishing, 2014.

3. W. Jasim and D. Gu, “Integral backstepping controller for quadrotor path tracking,” in Advanced Robotics, The 17th International Conference on, pp. 50–55, July 2015.

4. Y. Alothman, W. Jasim and D. Gu, ”Quadrotor lifting-transporting cable-suspended payload control”, in 21st International Conference on automation and computing (ICAC2015), (Glasgow,UK), 2015.

5. W. Jasim and D. Gu, “Iterative Linear Quadratic Regulator (iLQR) vs LQR controllers for quadrotor path tracking,” in 18th International Conference on Electrical Systems, Control and Robotics (ICESCR2016), (Paris,France), 2016.

6. W. Jasim and D. Gu, ”Leader-follower formation $H_\infty$ controllers for quadrotors,” in International Journal of Robotics and Automation, Under Revision.

7. W. Jasim and D. Gu, “Robust path tracking control for quadrotors,” in Robotica, Under Revision.

8. W. Jasim and D. Gu, “Rubost team formation control for quadrotors,” in IEEE Transactions on Control Systems Technology, Under Revision.

9. W. Jasim and D. Gu, “Iterative Linear Quadratic Regulator (iLQR) control for quadrotors leader-follower formation flight,” in Journal of Intelligent and Robotic Systems (JINT), Under Revision.

10. W. Jasim and D. Gu, “The Quaternions with an Application of Quadrotors Team Formation,” a Book Chapter in Nova Science Publishers, Abstract was accepted, Under Revision.

*To my loving family . . .*

# *Acknowledgements*

First and foremost I would like to thank Allah, who has given me the strength and patience to carry out this work. My sincere gratitude goes to everyone who has contributed to this thesis in one way or another. To my supervisor Prof. Dongbing Gu for his continuous guidance and technical advices.

To the Iraqi government represented by Ministry of Higher Eduacation and Scientific Research and University of Anbar for the financial support and for giving me the opportunity to undertake this experience. Special thanks to the staff of of the School of Computer Science and Electronic Enginering. Special thanks to the Group of Robotics and particularly to Mr. Robin for his cooperative through all stages of the work.

Many thanks are reserved for all my friends and colleagues at the University of Essex (you know who you are). Every one of you has marked this journey with a contribution. A special thank to you Dr. Zakaria Absi for proofreading the first draft of this thesis and for the valuable comments and suggestions.

To my mother, brothers and sisters who, despite being so far away, have supported me every step of the way. Thank you for your support and immense love. Last but not least I would like to thank my dearest wife, sons and daughters. I won't ever forget your support and enduring patience, words can not describe what that has meant to me.. . .

# Contents

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| AscTec. | Ascending Technology |
| BS | Backstepping |
| FL | Fuzzy Logic |
| HJI | Hamilton-Jacobi Inquality |
| IBS | Integral Backstepping |
| iLQR | iterative Linear Quadratic Regulator |
| LQR | Linear Quadratic Regulator |
| MPC | Model Predictive Controller |
| MRAC | Model Reference Adaptive Controller |
| NNs | Neural Networks |
| PD | Proportional Derivative |
| PI | Proportional Integral |
| PID | Proportional Integral Derivative |
| SM | Sliding Model |
| UAVs | Unmanned Aerial Vehicles |

# Symbols

| | |
|---|---|
| $B$ | Thrust factor |
| $\mathcal{B}$ | Body frame |
| $c$ | Cosine function |
| $C_p$,$C_q$,$C_\zeta$ | Positive constants |
| $D$ | Drag factor |
| $d$ | Desired distance between the leader and the follower |
| $\mathbf{d} = [\mathbf{d}_\mathbf{v}^T, \mathbf{d}_\omega^T]^T$ | External disturbance applied to the leader and the follower |
| $\mathbf{d} = [\mathbf{d}_\mathbf{v}^T, \mathbf{d}_\eta^T]^T$ | External disturbance applied to the leader and the follower |
| $\mathbf{d}_\mathbf{v} = [d_{vx}, d_{vy}, d_{vz}]^T$ | Disturbance vector applied to the leader's and the follower's accelerations |
| $\mathbf{d}_\eta = [d_{\dot\varphi}, d_{\dot\theta}, d_{\dot\psi}]^T$ | Disturbance vector applied to the leader's and the follower's angle accelerations |
| $\mathbf{d}_\omega = [d_{\omega_x}, d_{\omega_y}, d_{\omega_z}]^T$ | Disturbance vector applied to the leader's and the follower's angular velocity rates |
| $\mathbf{e} = [0, 0, 1]^T$ | Vector |
| $f = f_1 + f_2 + f_3 + f_4$ | Total thrust |
| $F'$ | New follower frame |
| $G(\omega)$ | The gyroscopic effect |
| $I$ | Identity matrix |
| $\mathcal{I}$ | Inertial frame |

| | |
|---|---|
| $i = L$ | Leader |
| $i = F$ | Follower |
| $J$ | Moment of inertia matrix |
| $K_p, K_q, K_v, K_\omega, K_\zeta, K_\eta$ | Positive proportional and derivative gains for translation and rotational parameters for the leader and the follower |
| $l$ | Distance from the centre of gravity to the motors of the quadrotor |
| $m$ | Quadrotor mass |
| $\mathbf{p} = [x, y, z]^T$ | Position vector |
| $\tilde{\mathbf{p}} = [\mathbf{p}_d - \mathbf{p}]^T$ | Position tracking error |
| $\mathbf{p}_d$ | Desired position |
| $[q_0, q_1, q_2, q_3]^T = [q_0, \mathbf{q}^T]^T$ | Quaternion parameters |
| $\mathbf{q} = [q_1, q_2, q_3]^T$ | Quaternion vector part |
| $q_0$ | Quaternion scalar part |
| $\tilde{\mathbf{q}}$ | Quaternion error |
| $\mathbf{Q}$ and $\mathbf{Q}_N$ | Symmetric positive semi-definite state cost weighting matrices |
| $R_q$ | Rotation matrix in quaternion form |
| $R_\theta$ | Rotation matrix in Euler angles form |
| $\mathbf{R}$ | Positive definite control cost weighting matrix |
| $s$ | Sine function |
| $S$ | Skew-symmetric cross product matrix |
| $\mathbf{v} = [v_x, v_y, v_z]^T$ | Linear velocity |
| $\dot{\mathbf{v}}$ | Linear velocity derivative |
| $\tilde{\mathbf{v}} = [\mathbf{v}_d - \mathbf{v}]^T$ | Linear velocity error |

| | |
|---|---|
| $\mathbf{v}_d$ | Desired linear velocity |
| $W_1, W_2, W_3, W_4$ | Positive weighting diagonal matrices |
| $\mathbf{x}_N$ | Last step of each state |
| $\mathbf{x}^*$ | Last desired state |
| $\delta\lambda_{k+1}^T$ | Lagrange multiplier |
| $\rho$ | Desired incidence angle |
| $\sigma$ | Desired bearing angle |
| $\mu_1$,$\mu_2$ and $\mu_3$ | Constant diagonal matrices |
| $\tau_q = [\tau_{q_1}, \tau_{q_2}, \tau_{q_2}]^T$ | Torque vector in quaternion parameters representation |
| $\tau_E = [\tau_\varphi, \tau_\theta, \tau_\psi]^T$ | Torque vector in Euler angles representation |
| | using quaternions and Euler angles |
| $\omega = [\omega_x, \omega_y, \omega_z]^T$ | Angular velocity in the body frame $\mathcal{B}$ |
| $\omega'$ | Angular velocity in the inertial frame $\mathcal{I}$ |
| $\dot{\omega}$ | Derivative of the angular velocity |
| $\tilde{\omega} = [\omega_d - \omega]^T$ | Angular velocity error |
| $\omega_d$ | Desired angular velocities |
| $\Omega = [\Omega_1, \Omega_2, \Omega_3, \Omega_4]^T$ | Rotor speed |
| $\zeta = [\varphi, \theta, \psi]^T$ | Roll, Pitch and Yaw angles |
| $\eta = [\dot{\varphi}, \dot{\theta}, \dot{\psi}]^T$ | Roll, Pitch and Yaw angles rate |
| $\tilde{\zeta} = [\zeta_d - \zeta]^T$ | Angles error |
| $\tilde{\eta} = [\eta_d - \eta]^T$ | Angles derivative error |
| $\zeta_d$ | Desired angles |
| $\eta_d$ | Desired angles derivative |

# Chapter 1

# Introduction

## 1.1 Introduction

In recent years, research on the control of Unmanned Aerial Vehicles (UAVs) has been growing due to its simplicity in design and low cost. Quadrotor helicopters have several advantages over fixed-wing air crafts, such as taking off and landing vertically in a limited space and hovering easily over fixed or dynamic targets, which gives them efficiency in applications that fixed-wing air crafts cannot do, in addition to being safer [1–3]. Quadrotor UAVs can be used to perform several tasks in the applications of dangerous areas for a manned aircraft in a high level of accuracy. They can be utilised in different applications, such as inspection of power lines, oil platforms, search and rescue operations, and surveillance [4, 5]. Increasing the applications of quadrotors encourages the growth in their technologies and raises the requirements on autonomous control protocols. Moreover, using swarm robotics has advantages over individual robots in that they perform their tasks faster with high accuracy and use a minimum number of sensors by distributing them to the robots [6]. Researchers are focusing on the design and implementation of many types of controllers to control the take-off, landing and hovering of individual quadrotor UAVs with some applications which require the creation of a trajectory and tracking in three dimensions, benefiting from the wide developments in sensors.

Research in the field of control of individual and multi-robot quadrotor team formation is still facing some challenges. Challenges of individual quadrotor control come from the complexity of modelling its dynamic system because of its complex structure and the design issue. The dynamic model equations present four input forces with six output states, which means that the system is in under-actuated range [5, 6]. Further challenges of multi-robot control come from evaluating the control architecture and communication network limitations.

The formation problem of quadrotors has had a vast area of interesting research in the past few years. Researchers have been motivated to contribute to this field of research by the development of materials, sensors and electronics used in designing quadrotors, which consequently has an effect on minimising their size, weight and cost. Working as a team of quadrotors has many benefits over using a single quadrotor in several applications.

Team formation control includes many problems to be addressed, including communication loss, delay between the robots or packet drop problems [7–10]. Simultaneous localisation and mapping is another problem in team formation control, in which the vehicle builds up its maps and estimates its location precisely at the same time; this problem has also been addressed in [11–13]. The third problem is the collision and obstacle avoidance, which includes avoiding collisions with both other robots and static or moving unknown obstacles while flying to their destination and maintaining their positions. Solutions to this problem have been handled by [11, 14]. Now, team formation control adopts a combination of some functions; the first is to perform the mission between two points, the second is to preserve the comparative positions of the robots over the formation and maintain the shape consequently, the third is to avoid obstacles and the forth is to divide the formation. In this thesis, we focus on designing only a control law for the leader-follower team formation problem with collision avoidance between team members by maintaining the distance between the leader and the follower.

In the leader-follower approach, at least one vehicle performs as a leader and the other robots are followers. The leader vehicle tracks a predefined path, whereas the followers maintain a certain distance with the leader and among themselves to obtain the desired shape. Each robot has its own controller and the robots keep the desired relative distance between themselves. However, two types of control architecture may be used to control the vehicle: one loop control scheme and two loop control scheme. If a two loop control scheme is used to control each vehicle, the outer loop is used for position control and its $x$ and $y$ output is the desired roll and pitch angles. These desired angles with the desired yaw angle are used to calculate the vehicle torques; in other words, they stabilise the quadrotor angles. This type of control is built according to time scale separation, where the attitude dynamics are much faster than translation dynamics. In the one loop control scheme, on the other hand, separation of the vehicle dynamics to attitude and translation is not considered. In this case, the position tracking error is used directly to calculate the vehicle torques to achieve its path tracking. According to these definitions, leader-follower team formation requires attitude stabilisation and path tracking to be achieved.

Abundant literature exists on the subject of attitude stabilisation, path tracking and leader-follower team formation control. Several control techniques have been demonstrated to control a group of quadrotors varying between the linear PID, PD or LQR controllers to more complex nonlinear controllers as neural networks and BS controllers. These controllers achieved good results and some of them guaranteed the performance, such as the LQR controller, and some of them guaranteed their stability. The performance of an individual quadrotor or a group of quadrotors in formation control is often affected by external disturbances such as payload changes (or mass changes), wind disturbance, inaccurate model parameters, etc. Therefore, the controller must be robust enough in order to reject the effect of disturbances and handle the change in model parameter uncertainties. Robust state feedback controllers are very demanding in this case. The $H_\infty$ control approach is able to attenuate the disturbance energy by measuring a ratio between the energy of cost vector and the energy of disturbance signal vector.

For this reason, a robust nonlinear $H_\infty$ state feedback controller is used in this thesis for the quadrotors flight problem. On the other hand, improving the control performance optimality is another aspect. An iLQR controller is addressed based on the optimal LQR control technique to improve the control performance.

Several control techniques have been explored in this thesis in both the simulations and the experiment. First, a robust nonlinear $H_\infty$ state feedback controller was developed to stabilise the quadrotor attitude, track a predefined path and address the team formation control problems with external disturbance consideration by solving Hamilton-Jacobi inequality. The controller stability was analysed via Lyapunov function and robustness conditions were obtained. The main advantages of applying the nonlinear $H_\infty$ optimal control approach are various. Firstly, it is able to attenuate the disturbance energy by measuring a ratio between the energy of cost vector and the energy of disturbance signal vector [15]. Secondly, although it is a nonlinear control approach, the performance criterion can be included in the control objective. Thirdly, robustness of stability and performance is guaranteed. Finally, solving the control action law leads to tuning parameters, which is easy to find better parameters [16]. Due to these reasons, the nonlinear $H_\infty$ optimal control approach was chosen in this thesis. The controller was tested in simulations and real work under different scenarios. On the other hand, an optimal control approach was developed based on the LQR control approach. The iLQR controller was applied for the full quadrotor nonlinear dynamic model, and the dynamic model was linearised at each time step as an equilibrium point. Third, Proportional Derivative square $PD^2$, IBS and LQR controllers were applied for comparison purposes. All these controllers were tested in simulation with several scenarios.

Dynamic model representation of the quadrotors is a major demand for designing these controllers. In this thesis, two techniques were used to represent the quadrotors: Euler angles and unit quaternion methods. The quaternion method was used to overcome the singularity problem which faced the researchers who used the Euler angles representation. A singularity emerges from the so called gimbal lock and it is appearing when dividing the pitch angles $\theta = \pm 90$ by zero.

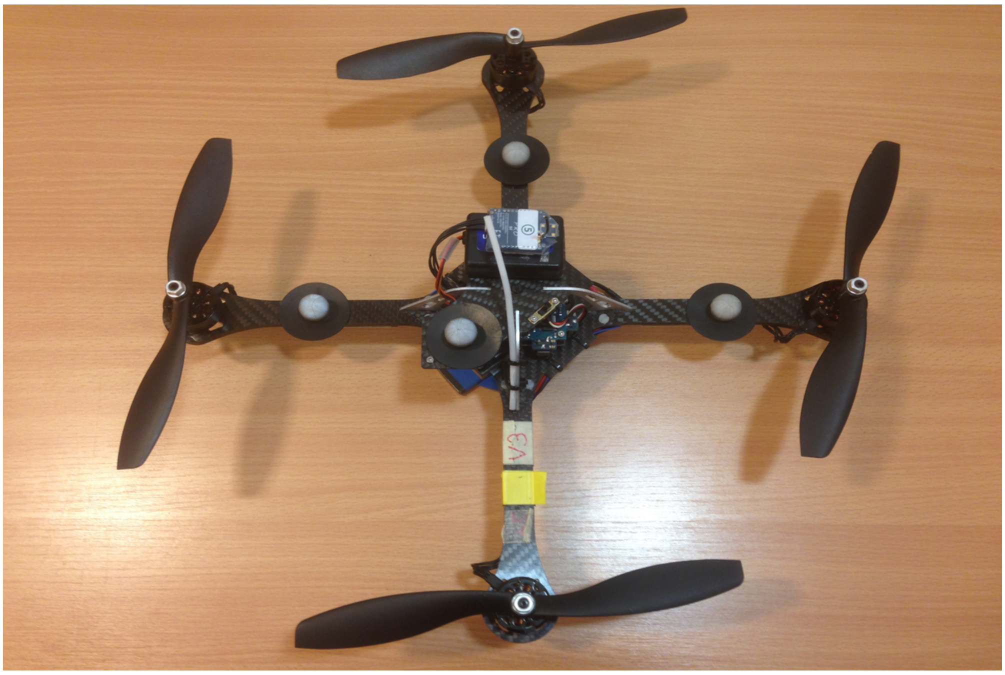

Figure 1.1: AscTec. Hummingbird Quadrotor

Many researchers have started to build their own quadrotors and use them to achieve the required tasks. One of these researcher groups, a small student group from Germany, developed a simple quadrotor toy and, in 2003, established their own company called Ascending Technology (AscTec.). The quadrotor used in our research is from this company (see Figure 1.1). This quadrotor has two micro-controllers: a low level micro-controller for attitude stabilisation and a high level micro-controller for translation control. In the high level micro-controller, the desired roll and pitch are calculated and sent to the low level micro-controller to be used for angles stabilisation control. More details of the vehicle features will be illustrated later in 3.6.1.

## 1.2 Motivations

Nowadays, the demand on the cooperative multi-robot system is widely increasing to handle the vast area of applications that individual robots cannot do, which is forcing researchers to concentrate on controlling a group of UAV quadrotors to improve the overall performance and also achieve their tasks effectively in a cooperative way. One such technique of cooperative problem control is formation control. Formation control is utilised to solve the problem of maintaining a group of vehicles while performing their task, taking advantage of energy saving and sensors sharing.

Although there are different approaches to formation control design, the main approaches are the leader-follower approach, the virtual leader approach and the behavioural approach. In the leader-follower approach, one robot in the group is selected as a leader and it is assumed to manage the path tracking algorithm, while the remainder of the group are chosen as followers. These followers should follow the leader and maintain the relative position and orientation according to the leader's position and orientation. The main drawback of this approach is that the response of the farthest robot from the leader will be very poor compared with the closest one to the leader. In the virtual leader approach, the computer sends the same position information of the path to be tracked to each robot in the team and they are required to track it, which means that no real leader exists in the team. The main disadvantage of this approach is the absence of feedback from the virtual leader to the formation. Finally, in the behavioural approach, the behaviours set for each robot should be determined. Then the control action of each robot for each behaviour is weighted. The behaviour examples are obstacle avoidance, collision avoidance, and formation keeping [17] The drawback of this approach is that the formation stability is not guaranteed and it needs massive information. In this thesis, we focus on the leader-follower approach to maintain the desired distance, the desired bearing angle and the desired incidence angle among the robots. To reach this target, a considerable work should be done as a preface, such as attitude stabilisation and tracking predefined paths.

Successively, designing a controller for an individual quadrotor or a team of quadrotors should consider various vehicles' features, in addition to the nonlinear and multivariate characteristics of its dynamics. Some of these features are the uncertainties of sources in the system, such as external disturbances and communication time delay. The controller is also affected by the aerodynamics of the four rotor blades, gyroscopic effects and inertial change. To overcome these effects, robust controllers were developed in this thesis.

## 1.3 Aims and Objectives

This thesis focuses on flight control of an individual quadrotor UAV and two quadrotors in the leader-follower formation. As mentioned before, the reason for choosing this type of vehicle is that it is still facing some challenges in the control field of individual vehicles and teams of vehicles when some nonlinear parts are included in their dynamic model. Hence, this thesis aims to improve the performance of the vehicle's flight in complex environments and different circumstances. To achieve these tasks the following objectives were carried out:

- The dynamic model of the vehicle is derived including nonlinear parts, and is implemented in MATLAB Simulink.
- A $H_\infty$ control technique is developed considering the external disturbances in its mathematical derivation.
- An optimal iLQR control algorithm is developed to improve the speed of the vehicle behaviour.
- The above two controllers were tested in simulation.
- The IBS, $PD^2$ and LQR controllers were derived and tested in simulation for comparison purposes.
- The $H_\infty$ controller was tested practically in real vehicles.

In conclusion, the $H_\infty$ and iLQR controllers aim to improve the vehicle performance by guaranteeing the rejection of the external disturbances and the speed of target catching.

## 1.4 Thesis Scope

Many authors have proposed to control an individual quadrotor and team of quadrotors to achieve several tasks. Linear and nonlinear dynamic models have been used with several control techniques. Although some of them have addressed a limited disturbance in their tests, it has not been included in the control law derivation. It was obvious that, when a linear controller was applied to a nonlinear dynamic model, it displayed good results in a free environment only; and when a nonlinear controller was applied, the vehicle consumed a long time to catch the target and some times it it was not able to reject high disturbances. It was concluded that including the external disturbances in the control law derivation and improving the speed of the vehicles' response were needed.

The scope of this thesis is to develop control techniques where the external disturbances can be included in the mathematical derivation and improve the vehicles' response speed. These are achieved by developing two control techniques, a nonlinear robust $H_\infty$ controller and an iLQR optimal controller, when they applied to the quadrotors' nonlinear dynamic model.

## 1.5 Contributions

Quadrotor attitude, path tracking and team formation control pose many interesting research problems that have remained open. This thesis concentrates on two controllers, $H_\infty$ and iLQR, to solve these problems taking into consideration disturbances and model uncertainty parameters change. The objective is to ensure the asymptotic stability of each system state of the closed loop nonlinear system. The main contributions of this thesis are the theoretical development and implementation of the $H_\infty$ and iLQR controllers, as well as practically testing the $H_\infty$ controller. These contributions can be summarised as follows:

1. As the quadrotor is an unstable vehicle, a state feedback controller for attitude stabilisation, path tracking and team formation problems of quadrotors

was deducted. A robust controller was synthesised via the $H_\infty$ optimal design approach. Solving the nonlinear $H_\infty$ optimal control problem using state feedback was reduced to finding a solution to a HJI. This control method focused on the stability and robustness when the external disturbances were considered.

2. Sufficient conditions for the stability of (i) attitude (quaternion parameters/Euler angles), (ii) path tracking, and (iii) team formation, considering the disturbance rejection and change recovery in the uncertainty model parameters, were obtained. However, these conditions were used to find out the suitable controllers' parameters.

3. An integral backstepping controller for path tracking and leader-follower team formation problems was derived and implemented in simulation for comparison purposes. Stability analyses were carried out based on Lyapunov functions.

4. A dynamic programming method that uses quadratic approximations to the optimal cost-to-go function – iLQR controller – was developed for both individual quadrotor and leader-follower team formation control problems. The controller is an iterated one based on a linear quadratic regulator approach. The main advantage of using the iLQR controller is its ability to overcome the noise of the nonlinear dynamic system.

5. The LQR controller was also developed and implemented for attitude, path tracking and leader-follower team formation problems of UAVs for comparison purposes with the iLQR controller results.

6. Experimental evaluations of the $H_\infty$ controller for individual quadrotor path tracking and two-quadrotor leader-follower formation was conducted to validate the proposed controller.

## 1.6 Thesis Structure

The rest of the thesis is structured as follows.

**Chapter 2 :** provides a brief explanation about the two controllers – $H_\infty$ and iLQR – as well as their applications in robotics. Then, the advantages of using the unit quaternion over the Euler angles method in the quadrotor dynamic model representation is described. Next, a literature review that surveys various types of controllers used in quadrotor applications are outlined with specific reference to attitude stabilisation, path tracking and leader-follower team formation. At the end, a conclusion of the literature review is provided.

**Chapter 3 :** includes a review on the nonlinear $H_\infty$ optimal control approach followed by the main results when that approach is applied for attitude stabilisation and path tracking problems. The control is built upon a single loop architecture where the four outputs – the thrust and the torques – are achieved, using the state feedback $H_\infty$ controller. The controller is used when the system is represented based on unit quaternion and based on Euler angles in simulation with consideration of external disturbances and model parameter uncertainties. The simulation results are presented to show the performance of the proposed controller. Then, the controller is implemented on a real vehicle and experimental results verify the robustness of the proposed controller.

**Chapter 4 :** describes the leader-follower team formation control problem. Then, the main results and the necessary conditions of the nonlinear $H_\infty$ optimal control approach are provided when the latter is applied to the follower vehicle. The simulation results are obtained when the vehicle dynamic model is represented based on unit quaternion and based on Euler angles with consideration of external disturbances and model parameter uncertainties effect. At the end, experimental results of two real identical vehicles using the proposed $H_\infty$ controller are provided.

**Chapter 5 :** gives the derivation of the IBS control approach based on the derivation of the BS control approach followed by its stability analysis. The controller is used for an individual vehicle's flight control and then for the leader-follower team formation problem. The simulation results of attitude stabilisation using the $PD^2$ controller and of path tracking and leader-follower team formation using IBS are presented for comparison purposes with those of the $H_\infty$ controller.

**Chapter 6 :** presents the optimal iLQR control technique developed based on the LQR control technique to control and stabilise the attitude and altitude of an individual quadrotor. Then, the technique is used to control two quadrotors in leader-follower team formation. The controller is verified using MATLAB Simulink simulation and the results of these simulations are provided and compared with those of the LQR controller.

**Chapter 7 :** summarises our findings and provides a discussion based on the results acquired in the previous chapters. It also includes suggestions for future work.

# Chapter 2

# Literature Review

## 2.1 Research Background

Formation of quadrotor UAVs is considered as a key point for the team of UAVs to perform their task cooperatively. More recently, team formation of multi-robot UAVs have attracted increasing interest in the field of control. Several researchers have investigated many control algorithms to achieve the formation of a team of quadrotors. In this chapter, a literature review is presented for the nonlinear $H_\infty$ control and the iLQR technique. Then the chapter addresses the issue of stabilisation, path tracking and team formation of quadrotors based on Euler angles and quaternion representation.

### 2.1.1 Nonlinear $H_\infty$ Control Technique

The leader-follower formation control performance of a multi-quadrotor system is often affected by external disturbances, such as payload changes (or mass changes), wind disturbance, inaccurate model parameters, etc. Therefore, a robust formation controller is needed in order to reject the effect of disturbances and cover the change in model parameter uncertainties.

A nonlinear $H_\infty$ optimal control approach is hardly ever discussed in the problem of the quadrotors team formation for the difficulties in solving the multivariable partial differential equation (or inequality) of Hamilton-Jacobi. Solving the Hamilton-Jacobi equation to find its generic solution is the main deadlock of the nonlinear $H_\infty$ optimal control approach. Then an approximation approach is used to solve it [15, 16].

To the best of my knowledge, the main previous works in quadrotor control application using the nonlinear $H_\infty$ optimal control approach are [18–20]. Authors in [18, 19] present an $H_\infty$ optimal control technique for stabilisation of quadrotor angles, while a full state control is derived in [20] with quaternion dynamic system representation.

### 2.1.2 iLQR Control Technique

iLQR is a dynamic programming technique developed based on the LQR technique with the use of quadratic approximations to the optimal cost-to-go function [21]. The main idea is that the nonlinear dynamic model is linearised around a nominal predefined path at each time step. Then the proposed iLQR technique calculates the optimal control law based on Reccati equation. The iLQR controller has rarely been used in literature. There are a few examples, however, where it was used to estimate an approximation of the optimal cost to be learned via supervised learning algorithm in [22]. It was also presented in [23] for wheeled mobile robot trajectory tracking. L. Weiwei and T. Emanuel presented the iLQR controller to control a musculo-skeletal arm and calculate the movement optimal energy [24]. In this thesis, the iLQR controller is used to track the attitude quaternion states, the position states of a quadrotor and team formation control.

### 2.1.3 Euler Angles and Unit Quaternion

It has been shown that the rotations of the UAV or any rigid body can be derived using different approaches and are composed of translational and rotational dynamics. Newton-Euler, which is based the Euler angles approach, has been widely used in this status but it has three important drawbacks. Firstly, the Euler angles representation of the attitude suffers from the singularity problem which is also called "gimbal lock". The singularity problem occurs by losing one degree of freedom of the attitude when dividing the pitch angles $\theta = \mp 90$ by zero. Secondly, it is very slow in computation because it has sine and cosine terms. Thirdly, the jacobian cost function of the system states requires a long time in computation because its matrices almost have at least sine or cosine terms in each element, which may lead to crushing the system.

The possible solutions for these problems can be concluded in the following: 1) limiting the Euler angles (the problem of this approach is that it will not be able to reject the external disturbances); 2) using the Direction Cosine Matrix (DCM) approach (the drawbacks of this approach is the orthogonal relationship of the axes to each other and their unit length; and 3) using the quaternion approach (this approach was used to overcome the above drawbacks and its only drawback is that it must be of unit length).

## 2.2 Literature Review

### 2.2.1 Attitude Stabilisation

The first most important task of quadrotor UAV control is the stabilisation in a certain point. Stabilisation means controlling the take-off of a quadrotor UAV to a certain point, staying at this point and then landing with a pitch angle, roll angle, yaw angle, motion in x-direction and motion in y-direction equalling to zero. Various control algorithms have been implemented to address the stabilisation problem with nonlinear or linearised dynamic systems.

#### 2.2.1.1 Linear Controllers

A classical linear Proportional-Integral-Derivative (PID), a Proportional-Integral (PI), a Proportional-Derivative (PD), a Linear Quadratic (LQ) and a LQR controller have been applied for the quadrotor's stabilisation problem. Some of these linear algorithms are as follows: a path tracking strategy was implemented to achieve this task by tracking reference trajectory signals; a quadrotor platform was built and stabilised using a PD controller in indoor environment; this platform was contacted and controlled wirelessly, with a built-in micro-controller, two gyros, a magnetic compass, a mounted wireless camera, and zigbee and accelerometer sensors [1]. Experimental results show a good flight performance stability. The design and control of a quadrotor UAV was implemented by [25], in which PID and LQ controllers were proposed to stabilise the rotation angles of the quadrotor. Experimental results validate the success of the controllers with minor disturbance consideration.

A simple PD controller was tested in a practical test platform using the error quaternion which comes from the estimated rotation and the QUEST algorithm quaternion output. This controller ensures the stability of the quadrotor even when it is affected by the sensors noise, modelling errors and measurement reference frames [26]. A PD controller for vertical take-off and landing of a quadrotor UAV was presented in [3] in normal circumstances with an on-board camera to calculate the quadrotor altitude. The integration of IMU measurement was involved with the PID pose-controllers to recover the Ascending Hummingbird quadrotor behaviour. The platform had a mounted camera in addition to the basic sensors. The PID controllers showed a good performance in terms of stable autonomous take-off and hovering in real time over fixed markers with some oscillations [27].

Schmidt [28] introduced an angle platform control using a MATLAB and Simulink PID controller to achieve the angles stability. The platform was implemented with the substantial parts and sensors. The PID controller was flashed in a processor within the platform. The platform was tested in ideal conditions with no sensors effect and air friction or drags effect. PI and PID controllers were also implemented in [29]. The PI controller was used to control the height of the quadrotor, whereas the three angles and vertical placement were stabilised using the PID controller. The controllers were tested by adding the effect of drags and gravitational forces and the results assert the robustness of the controllers.

Jaromir D. et al. [30] introduced an orientation controller based on LQ-optimal state-feedback combined with an eigenaxis rotation. The controller was tested experimentally to achieve the quadrotor attitude stabilisation and the results show a successful performance with less than one degree errors. A $PD^2$ feedback controller and a PD controller were proposed in [31] to obtain the global exponential attitude stabilisation of the quadrotor. The first controller had three terms: the proportional term affected the quaternion parameters, and the two derivative terms affected the quaternion parameters rate and the angular velocities. The second controller had two terms: the proportional term affected the quaternion parameters while the derivative term affected the angular velocities. Both controllers were tested practically on a quadrotor with small angles and low speed motion. The results prove the efficiency of the controllers against small disturbance.

A single loop PID controller was applied to a linearised dynamic model in [32] for attitude and altitude stabilisation. The simulation results illustrated the stability of the quadrotor after 3 seconds shock adjustment. A comparison between the gain scheduled PID controller and the Model Reference Adaptive Controller (MRAC) was achieved in [33] by testing them on a normal and a faulty vehicle. The result shows that the first controller was easy to implement while the second controller was more robust. Esteves et al. [34] demonstrated several Kalman estimator techniques and a LQR controller for the Quadr-ANT quadrotor stabilisation problem. The indoor test revealed a good stabilisation performance within a 6-10 cm error in altitude and around 1 degree errors in angles. These linear controllers performed well in terms of stability with some oscillations or accepted errors in the absence of high disturbances or dynamic uncertainties.

#### 2.2.1.2 Nonlinear Controllers

Some nonlinear control algorithms have been utilised to achieve the stability of the rotational movement of quadrotor UAVs. In [2], a MRAC was presented with the assistance of a modular simulation environment for a quadrotor nonlinear dynamic system in the presence of actuator failures and loss of control effectiveness. The controller showed a stable performance during the loss of control effectiveness and actuator failures. On the other hand, a nonlinear proportional squared $P^2$ control algorithm was proposed in [35] for attitude control. The controller was implemented to a linear quaternion simulator of the quadrotor and the simulation results indicated a good tracking performance.

A PD sliding mode controller based on quaternion model representation gathering with a filter was implemented and tested practically on a quadrotor to achieve the position and attitude stabilisation [36]. The filter was used to filter the signals of the angular velocities and their rate. The xperimental results showed very good performance and the stability was guaranteed. To guarantee the stability of the controllers, a continuous-time MRAC based on Lyapunov function was applied for the nonlinear system based quaternion model, in addition to implementing a Model Identification Adaptive Controller to a quadrotor in order to overcome the inertia variations [37]. A bounded control law was presented to obtain the attitude stabilisation of the quadrotor and the controller was designed to control the bounded torque using a saturation function which depends on the quaternion parameters. This approach aimed to ensure the effectiveness of the actuators without affecting the stability of the quadrotor [38].

Garcia et al. [39] introduced a nested saturation control technique as the best controller in terms of smooth UAV behaviour and energy consumption compared with Backstepping (BS) and Sliding Mode (SM) techniques when they were tested in real time to stabilise the UAV beneficiary from a mounted camera to measure the quadrotor position and speed. The experimental results showed good stabilisation in using the three controllers. A BS nonlinear control technique was compared with the SM one when it was applied to control the quadrotor UAV [40]. The experimental results showed better performance of rotation angles stabilisation compared with the PID and LQ used in their previous work [25].

A filtered BS controller was implemented to control the attitude of a quadrotor represented using the unit quaternion approach [41]. The filter was a second order quaternion filter and it was used for the quaternion parameters and their rate calculation. It should be noted that the computation of the angular derivative vector depended on the quaternion rate calculated by the filter. The proposed controller was used for attitude stabilisation and path tracking problems, and a stable simulation performance was obtained in disturbance-free state. A simulation of the BS based PID technique combined with PD controllers was presented in [42] to stabilise the take-off and landing of a quadrotor UAV dynamic model derived using the Newton-Euler technique. An optimisation method was introduced here to obtain the parameter values.

Cabecinhas et al. [43] introduced a robust controller to ensure the robustness of the quadrotor UAV's take-off and landing to overcome the problem of dynamics change when the quadrotor contacts the land. In [44], a combination of torque PD and second order SM controllers was proposed to control the position and attitude of a quadrotor, respectively. However, the quadrotor was modelled based on a unit quaternion method and the controllers yielded interesting results in the practical test.

Asymptotic exponential attitude stability was achieved by designing a model independent and model dependent controllers. These robust controllers were tested on a simulation model based on the quaternion quadrotor and they contributed to a stable tracking for the angular velocities and angles [45]. Tayebi et al. [46] suggested a velocity-free attitude stabilisation control scheme for the attitude stabilisation problem of the quadrotor. The approach depended on the measurements of the body vector only, which in turn depends on the inertial measurement vectors. The simulation results proved the effectiveness of the proposed approach in normal circumstances. An event-triggered feedback nonlinear control was practically tested in [47] on a quadrotor to obtain the attitude stabilisation. This type of control was used to minimise both control cost function and communications. The feedback controller was designed relying on a state space based on the quaternion model and it was only valid in bounded state space domain.

#### 2.2.1.3 Intelligent Controllers

Another type of controllers that has been developed in the attitude stabilisation field is intelligent controllers such as, fuzzy logic (FL) and neural networks (NNs) controllers. A FL controller displays some benefits upon other aspects of control laws, particularly in terms of modelling uncertainties and high nonlinearity handling. Furthermore, designing a fuzzy controller might be conjectural with little information about the system. Therefore, some researchers have chosen the FL control system to build the quadrotor UAVs' take-off, flight and landing controller. However, the main challenge of using the FL to control the quadrotor UAVs is that it requires knowledge and experience about the behaviour of the system to find the memberships and build the rules of the controller.

Starting with the BS based FL and BS least mean square controllers, these were presented in [48] to stabilise the attitude of the quadrotor UAV. To guarantee the stability of the controllers a recursive Lyapunov function was applied. The results proved that the fuzzy controller outperforms the other controller in stabilising the rotation angles. A robust adaptive fuzzy control technique was proposed to obtain the stabilisation of quadrotor UAVs' attitude [49]. A set of alternate membership functions were used to avoid the drift in membership centres and a Lyapunov stability theory was applied to ensure the system stability. The simulation results proved the stability and robustness of the controller against a sinusoidal disturbance. Raza and Gueaieb [50] compared two types of FL controller, Mamdani and Takagi Sugeno Kang, when these were applied to control the six outputs of the quadrotor in simulation and real system in normal environment.

Sharma and Barve [51] also compared the simulation results of a PID controller with a FL one in take-off, hovering and landing stabilisation of the quadrotor for a constant angle of each rotor. The simulation results demonstrated a better performance when using the FL controller than when using the PID controller. The benefit of the FL controller was also established in [52] when the controller was tested in a simulation model of the quadrotor. Abeywardena et al. [53] proposed a Mamdani FL controller for stable flying of quadrotor velocities, in which the controller was tested on the MATLAB Simulink of the nonlinear dynamic model with disturbances. The main benefit of fuzzy nonlinearity was demonstrated by using the hybrid fuzzy PID controller together with a conventional PID to control the quadrotor in simulation.

On the other hand, a direct inverse neural control technique was introduced in [54] for a mathematical model of quadrotor while disregarding the impact of gyroscopes and air drag. The main drawbacks of using NNs in quadrotor control is that the controller needs the plant model itself or a perfect identified system to ensure the stability. The application of a mimic NN model to the $PI^{\lambda}D^{\mu}$ controller was applied for the quadrotor UAV. The training technique of Levenberg–Marquardt optimisation was utilised for the training data [55]. The NN was trained to find the parameters of the finite impulse response approximator related to the time varying parameters of the controller.

### 2.2.2 Path Tracking

Path tracking is the second task in controlling a quadrotor in our path to achieve the team formation control. Path tracking control has received a great research attention in the last decay and a significant advances have been obtained in this field. There is a wide range of control algorithms which were proposed for the path tracking problem in different circumstances in the presence or absence of dynamic effects.

#### 2.2.2.1 Linear Controllers

Classical PID and PD control technique were found to be popular techniques in the literature; for example, simple PD and PID controllers were implemented on a linearised model to control the path tracking to the desired point on a desired path for a quadrotor UAV while having wind disturbance and modelling uncertainties [56]. The results revealed that classical controllers, such as PD or PID, were not adequate under wind disturbance and uncertainty conditions. Hoffmann et al. [57] designed a novel path tracking control algorithm focusing on the compound problems of the average of the control input update and path planning in the desired speed; PD and PID controllers were used for this purpose. Mellinger et al. [58] also presented the PID controller to stabilise the orientation of an AscTec. Hummingbird quadrotor, where they designed a gripper added to the vehicle to assist in its perching and landing. The controllers were modified to hold the two and three dimensions trajectory tracking and eventually to achieve the perching and landing purposes.

An explanation of a successful autonomous path generation with obstacle avoidance capability utilising the vision data which originate from Google Earth only in virtual environment was presented in [59]. This autonomous path generation was obtained using a PID controller in the low level mode. A dynamic programming technique was proposed in [60] using a simulation of PD and nonlinear based input-output feedback controllers to address the problem of swing-free suspended load trajectory tracking in a quadrotor UAV. An iterative learning based PD controller was presented in [61]. The dynamic model was linearised and all the dynamic effects were neglected. The controller was tested in simulation and in an off-line tracking experiment. These controllers show a good performance when they were applied to a linearised system in the absence of disturbance and model parameter uncertainties.

LQR control is another linear control technique which was introduced for addressing the path tracking problem in [62]. The D-methodology was integrated with the anti-wind-up technique to obtain a zero steady state error. The linearised model was simulated to track a 3D trimming path, then the controller was tested experimentally to demonstrate the successfulness of the controller with small errors. The authors of [63] implemented and practically tested the LQR controller with and without the Model Free Control algorithm to track a predefined path.

The results of these two approaches showed the effectiveness of using the Model Free Control algorithm. A nested P and PID control structure was presented in [64] to control the quadrotor position. The controller was implemented and tested in simulation and practically in an open source quadrotor.

#### 2.2.2.2 Combination of Two Controllers

It was found that combining two control techniques achieved better flight performance than using one control technique. Raffo et al. [18] introduced a Model Predictive Controller (MPC) for translational control combined with a nonlinear $H_\infty$ controller for attitude stabilisation. Similarly, the authors in [19] reused the nonlinear $H_\infty$ controller for attitude stabilisation combined with a BS controller for translational movement. The simulation results in both studies showed a good performance in the presence of dynamic inertia effect.

A special state space model was presented depending on the gathering of the tracking error of the BS technique and Lyapunov function. The new state space model had some complex parameters and it was controlled using the Sliding Mode (SM) controller to track a way point path in [65]. A switching control methodology between the inertial and imaging sensors was proposed in [66], in which the PID controller and the integral SM controller were used to stabilise the linearised quadrotor UAV model once when the latter was moving in constant velocity over a detected road and another when it was not. The experimental result illustrated the stability of the switching method to reject the external disturbances.

Stabilisation of a quadrotor with onboard sensors was achieved in [67] by using a PID controller to ensure the horizontal position, while three PD controllers were used to cover the control of the three attitude angles. Two mounted cameras were added to the vehicle; the first one was to assist in path tracking and perching control at high speed flying, and the other was used for visual simultaneous localisation and mapping algorithm in unstructured environment navigation. The experimental results of the PD and PID controllers were compared with the results of a nonlinear control algorithm with some assumptions and limitations.

The altitude and attitude stabilisation results of nested PID and LQ controllers applied to the simulation model of a quadrotor UAV showed that the nested PID was better than the LQ controller [68]. While in the hovering case, the result of sensitively mixed $H_\infty$ and $\mu$-synthesis controllers was compared with that of PID and LQ controllers in path tracking and parameter uncertainty.

#### 2.2.2.3 Nonlinear Controllers

Various techniques based on the BS approach were proposed and tested practically in [5, 69–71] to control the attitude and position of different quadrotors' platforms. For instance, the IBS controller was implemented and tested practically on a OS4 quadrotor platform to perform a autonomous take-off, hovering, landing and collision avoidance [5]. A nonlinear adaptive state feedback based on Lyapunov and the BS controller was presented in [69] to follow a time dependent path and constant force disturbances rejection. The proposed controller was tested on a radio controlled Blade mQX quadrotor to verify the controller validation. The controller was developed and retested to reject unknown force disturbances in [70]. However, there are still considerable concerns about the use of external disturbances as these works did not include the latter in the control law derivation. Moreover, the performance was not guaranteed and it had high relative errors. Therefore, there is a need to guarantee and optimise the performance. To this end, the nonlinear $H_\infty$ and iLQR controllers are employed to cover these features.

Choi and Ahn [71] presented a new nonlinear controller based on a BS-like feedback linearisation technique for a fully autonomous quadrotor system. The new controller was compared practically in outdoor take-off with PID and other two nonlinear controllers. The experimental results showed a good performance among the other controllers. A new approach of BS control was proposed to stabilise the quadrotor and track the way point route in the Cartesian mode [72]. The controlled system was divided into three subsets: horizontal position, altitude and yaw angle, and the propeller dynamics. It was found that the controller performed well in normal environment.

Moreover, some researchers addressed controllers based on BS for the path tracking problem for quadrotors based on quaternion representation, as demonstrated, for instance, in [73–75]. In [73] a BS control using a decoupling quaternion parametrisation was proposed and the controller depended on the decoupling of two rotations of quaternion. The stability of the proposed architecture was verified by an experimental test. A global tracking control using a one-step ahead BS controller depending on standard BS and Lyapunov's theorem was implemented and tested in simulation for a quadrotor in [74]. The dynamic model of the quadrotor was represented by a combination of quaternions and Euler angles. The results proved the asymptotic convergence in bounded controls. An et al. [75] proposed a BS based inverse optimal attitude controller for quadrotor representation based on quaternions and the simulation result was compared with that of a PD controller. The controller had two parts: the BS to deal with the quaternion parameters model set and the inverse optimal technique to deal with the Hamilton Jacobi Bellman equation computation. This approach focused on some limitations in the input torque for a large angle flight.

A SM controller approach is another popular technique utilised for path tracking in [44, 76, 77]. Parra-Vega et al. [76] proposed two novel model-free second order SM controllers for attitude stabilisation and path tracking. The work was composed of two theorems: the first one was to ensure semi-global exponential and robust path tracking of a closed-loop system with a zero yaw angle, while the second one was deduced for final stability. The simulation results showed a smooth performance in the free disturbance state and a small spike in the disturbed condition. A SM controller surface based on quaternion representation for time-parametrisation control of a quadrotor was introduced by [77]. The controller was obtained via exponential and terminal stabilisation. Robust performance with small errors was achieved under several simulated scenarios. In [44] a PD gathering with second order SM controllers was implemented to solve the problems of path tracking control and attitude stabilisation, respectively. The controllers were tested in simulation and practically to show their effectiveness in normal circumstances.

Some researchers focused on the other nonlinear control algorithms to solve the path tracking problem. Sorensen [78] compared the simulation and real time stabilisation for a AscTec. quadrotor using two controllers, LQR controller and robust $H_\infty$ controller, when these were tested in tracking of a high speed trajectory. The compared results showed that the $H_\infty$ controller could not track the trajectory in real time while the LQR controller gave a good tracking in both the simulation and real time tests.

A strategy of adaptive scheme with learning reinforcement control was demonstrated experimentally in [79]. The proposed technique showed successful performance when the quadrotor performed a piece's transformation from one point to another. A nonlinear controller was designed for attitude and position control of a quadrotor in [80]. The controller was tested practically with constant disturbance consideration. An adaptive robust controller was proposed in [81] to cover the model uncertainty error and reject the disturbance. The controller was developed via a Lyapunov-like energy function. The results obtained by testing the proposed controller in a real quadrotor showed the validation of this theory.

A MPC controller was used for path tracking control of the quadrotor in [82]. The path destinations were found by nonlinear guidance logic, and the controller was tested for obstacle avoidance as well. The Authors in [83] presented a nonlinear model-based position controller for the quadrotor path tracking problem. The controller was tested in simulation and real phase. Practical results showed a good performance in terms of overshot and settling time. In [84] a differential flatness technique of a quadrotor to follow a vector field as an input to the quadrotor was addressed. The mathematical model was derived and applied in simulation and experiment. In [85] an I/O translational linearisation and reduced quaternion parameter of the attitude was presented for the quadrotor position control problem. The proposed feedback linearisation method was tested successfully on a real quadrotor.

Other nonlinear control techniques were illustrated in [86, 87]. In [86] a nonlinear robust tracking control based on quaternions in a particular Euclidean set was utilised to follow the position and attitude commands. The controller was simulated to reject considerable disturbances when it followed a complicated trajectory. The simulation results showed around 10 cm errors when the disturbance was considered. Moreover, A robust adaptive tracking controller was implemented and practically tested to achieve the attitude command tracking of a quadrotor. The controller was designed to track the attitude without relying on the quadrotor inertia information and ensure the unstructured disturbances rejection with relative high angular velocity error [87].

On the other hand, Elias et al. [88] presented a nonlinear quaternion mathematical model to describe the attitude of the quadrotor. They implemented an LQR gain scheduling simulation controller to obtain the trajectory and attitude stability task. A high relative error in simulation results was shown in the normal state. A path tracking controller using a so-called quasi-static feedback linearisation for the quadrotor dynamics was introduced in [85] to reduce the translational dynamics order. It consisted of two phases: the first relating to the altitude and consequently to the thrust and the second to splitting up the altitude from the other two directions. The attitude was described in two degrees of freedom. The experimental results proved the validation of the proposed approach without observable delay.

#### 2.2.2.4 Intelligent Controllers

Some researchers focused on the other nonlinear control algorithms to solve the path tracking problem. Example of these algorithms were used in [89–95]. Castillo et al. [89] focused on a stabilisation control algorithm based on Lyapunov analysis using nested saturations to control the quadrotor UAV in real time. A satisfactory result was obtained when a real vehicle tracked a 3D path. The results of trajectory tracking of self-tuning PID based on the FL controller were demonstrated and compared with that of a conventional PID controller when they were applied to control the quadrotor UAV with variable payload. The use of the FL controller displayed good results compared with the PID controller in terms of handling the uncertainty of the system, which occurred as a result of the payload variation [90].

Yacef et al. [91] addressed an adaptive fuzzy control scheme for the trajectory tracking problem. The controller was based on the Lyapunov direct method and backstepping techniques. The proposed control method did not require the quadrotor dynamic model. A robust control approach based on the Takagi-Sugeno fuzzy model together with the Linear Matrix Inequality technique was demonstrated in [92]. The use of the fuzzy model leads to a linear model that is valid in several operational points. The proposed controller was designed with the pole placement approach and its simulation results were compared with those of LQR controller.

Zareb et al. [93] presented a fuzzy-PID hybrid control system for the quadrotor autonomous flight. The controller consisted of a Mamdani fuzzy controller for attitude stabilisation and a PID controller for roll, pitch and altitude control. The simulation results showed the stability and robustness of the proposed controller. A smart self-tuning fuzzy PID controller based on Extended Kalman Filter algorithm was addressed for the quadrotor's angles and position control problem in [94]. The smart selection method was used to select the active fuzzy parameters in order to minimise the calculation time, and a Dijkstra's technique was used to find the shortest path and help in obstacle avoidance. The proposed controller showed good performance compared with a traditional PID controller. Authors in [95] proposed a fuzzy radial basis function neural network PID control system for a quadrotor based on particle swarm optimisation to control the quadrotor. The results of the proposed controller were compared with those of PID, fuzzy PID, neural network PID and fuzzy neural network PID controllers. The comparison verified the effectiveness of the proposed controller.

#### 2.2.2.5 Path Generation

Path generation and tracking is another field in quadrotor control and it is beyond the scope of this thesis. Our work is limited to the stabilisation and tracking of a generated path. In [96] a nonlinear BS controller was implemented for quadrotor tracking trajectory generated by three different types of visual servoing, 2D, 3D and 2D1/2 techniques, in with presence of disturbance. The results demonstrated undesirable rotational movement when using 2D visual servoing while using 3D visual servoing led the quadrotor out of the camera scope and using 2D1/2 technique resulted in better performance than the other techniques. Chamseddine et al. [97] presented a flatness-based flight trajectory planning/replanning technique for a quadrotor UAV to solve the problem of trajectory planning in terms of time of the mission and the constraints of the actuators. Control laws for linearised dynamic equations of a quadrotor UAV were presented to decide if there was a necessity to design complicated control techniques for unmodelled UAV or not.

### 2.2.3 Team Formation

In the last decade, the focus on control single unit quadrotors has expanded to controlling a team of quadrotors for these to be able to achieve their tasks in variable weather and complicated environments. Team formation flight also provides advantages over the use of an individual quadrotor in both civil and military applications, such as inspection of an inaccessible area, disaster management, and search and rescue in risky circumstances, etc. Most of these applications demand more than one quadrotor to accomplish the desired objective [81, 98]. The leader-follower approach is one of the main approaches of formation control design.

Distributed and decentralised control techniques were used in the literature to solve the leader-follower control problem. The distributed control technique assumes that not all followers receive the leader's information and there is a kind of cooperation among them [7–9, 13, 99–106], while the decentralised control technique proposes that all followers are able to receive the leader's information [10, 107–115]. Different controllers have been implemented with both distributed and decentralised control techniques.

#### 2.2.3.1 Distributed Control Technique

A robust LQR controller was proposed for individual quadrotors and team formation as well in [7]. The controller was designed for a linearised system around the hovering point. The simulation results indicated the ability of the controller to overcome the changes in communication topology among the robots with no dynamic effects. A NNs controller was presented in [99] for addressing the leader-follower problem. These two studies used Lyapunov theory to analyse the controller stability.

A BS controller was discussed in [102] based on graph theory to maintain the distance among the robots and in [101] with balanced graph and strong connection among the robots. The quadrotors' dynamic systems were linearised around the hovering point and a good performance was obtained in normal circumstances. A distributed cohesive motion control scheme was presented in [103] for 3D motion to maintain the distance among robots. This technique was developed to become a decentralised technique and significant attempts to deal with decentralised control techniques have been made. Three time scale controllers based on the SM controller were proposed in [100] for dealing with the quadrotor formation problem. The controllers were used for the path tracking, attitude tracking and velocity in order to keep the formation and maintain the distance among the robots with the presence of external disturbance affecting the leader robot only. The simulation results proved the effectiveness of the proposed scheme.

A nonlinear control theory was presented to ensure the stability of quadrotors team formation in [8]. The wireless networks communication among the team was obtained via medium access control protocols. Experimental tests verified the proposed algorithm with time delay consideration. In [104] the problem of the leader-follower consensus of a swarm of rigid body space crafts system was analysed based on quaternion representation using a distributed control technique. They assumed that the communication between two neighbouring followers is bidirectional and that all followers can receive the leader information. Stability analysis was obtained via Lyapunov theory and the simulation results proved the attitude and angular velocity tracking stability. In [13] a MPC technique with integrated trajectory planning was analysed with a planning horizon for both team formation and obstacle avoidance. The method showed good simulation results. A distributed coordinated control scheme was proposed by [105] to solve the problem of time-delay in leader-follower team formation communication of quadrotors and the simulation results under sufficient conditions demonstrated the validity of the presented control technique. Xiwang et al. [9] proposed a consensus-based approach for the time varying formation control problem. The simulation and the practical test of five quadrotors demonstrated the validation of the proposed control approach. A vision-based servoing distributed control approach was presented in [106], where the quadrotors equipped cameras to track a moving target which provided the position information to be used for controllers.

#### 2.2.3.2 Decentralised Control Technique

Abdessameud and Tayebi [107] proposed a procedure which depends on a quaternion representation and is split up into translational and rotational control design under the upper bounded translational control input. Analysis of the closed-loop system stability was achieved using Lyapunov theory. The proposed strategy took 8 seconds to catch the desired formation shape. A hybrid supervisory control based on a polar partitioning approach was suggested in [108] for the team formation problem and for collision avoidance as well. The combination of discrete quadrotors dynamic system and the supervisor was achieved using the parallel composition and the simulation results displayed that this method allows the supervisors to achieve a free collision in normal environments. A MPC technique was proposed in [112], where its hierarchical control effectiveness was compared with the potential field technique. The stability of the feedback controller based on fluid dynamic models in [109] was obtained based on smoothed-particle hydrodynamic. The simulation results of the above methods validated the proposed approaches.

Authors in [110] proposed the trajectory planners and feedback controllers for following the planned trajectory. Next they proposed a nonlinear decentralised controller for an aggressive formation problem in the micro quadrotors team in [111]. Communication failures and network time delays impact on team formation efficiency were considered. Local information of neighbour robots in the team was used for individual trajectory planning. Preserving the required form was based on the status estimation of neighbour robots. Then the authors presented two approaches to overcome the problem of concurrent assignment and planning of trajectories (CAPT) for the quadrotors team, a decentralised D-CAPT and centralised C-CAPT in [10]. The decentralised D-CAPT and centralised C-CAPT results were compared in simulation and practice and the experimental results demonstrated a good performance in indoor application.

In [116] a human user for teleoperation with a haptic device was proposed for the quadrotor team formation control problem with the cooperation of a BS controller. The simulation results revealed the ability of the human user to teleoperate in order to perform the formation. A triangle formation control of three quadrotors using optimal control techniques via the Pontryagin maximum principle was presented in [117] and the simulation results showed the effectiveness of using team formation rather than using an individual quadrotor in terms of fuel consumption. In [118] a consensus problem of swarm systems was discussed to obtain the time-varying formation based on double-integrator system modelling. The experimental results of the three quadrotors in formation verified the effectiveness of the proposed approach in dynamic-free conditions.

A new developed framework gathering with a nonlinear MPC technique was presented in [119] to solve the problem of coalition formation. The simulation results showed a zero steady state error in free disturbance and dynamic circumstances. Koksal et al. [113] presented an adaptive formation scheme for quadrotors leader-follower formation. They proposed a distributed control scheme for the kinematic part, an adaptive LQ controller for pitch and roll angles, proportional control for yaw angle and a PID controller for altitude. Several scenarios were implemented in simulation and experiment to validate the algorithm. In [114] a combination of LQR and SM controllers were proposed for a 2D quadrotors leader-follower formation, where the LQR controller was used for position control while two SM controllers were used for the attitude and for maintaining the distance between the robots. The simulation results demonstrated the successfulness of combining the two control techniques. A BS control approach with nonlinear controllers was introduced for handling the team formation problem in [115] and the simulation results proved the effectiveness of the proposed controllers.

The results in most of the previous papers on leader-follower formation control of multi-quadrotor system did not consider the effect of external disturbances, such as payload changes (or mass changes), wind disturbance, inaccurate model parameters, etc., which often affected the quadrotors' control performance. Therefore, a quadrotor controller must be robust enough in order to reject the effect of disturbances and cover the change in model parameter uncertainties and external disturbances. Robust state feedback controllers are very demanding when dealing with the quadrotor control problem. The $H_\infty$ control approach was able to attenuate the disturbance energy by measuring the ratio between the energy of cost vector and the energy of disturbance signal vector [15].

## 2.3 Discussions

Stabilisation, path tracking and team formation of the quadrotor were achieved using various types of controllers based on Euler angles and quaternion representations. Most of these control approaches were tested in practice and some of them in simulation. However, the majority of these control approaches have some drawbacks such as the requirement of a great deal of system information in order to build the controller and the discount of the effect of disturbances and parameter model uncertainties. Moreover, most of the above control approaches (PID, LQR, Backstepping, Fuzzy system and Neural Networks) were tested on the quadrotor with a linearised system.

On the other hand, most of the approaches described above side-step the vehicle control problem by either technique assuming that the vehicle dynamics are extremely simple or that an inner-loop controller that solves the problem exists. As UAV designs become smaller, lighter, and more agile, these assumptions no longer hold. While the approach described in [44] includes adaptation in the outer-loop control of the vehicle kinematics, this approach does not accommodate local errors in the vehicle dynamics, or global errors, such as the error in the overall configuration of the vehicles.

This thesis addresses the problem of control of multi-vehicle UAVs in the presence of disturbance and uncertainty by including adaptation in the control loop and adapting to both local and global errors. Moreover, the iterative method is also discussed to achieve fast track and minimum steady state errors. In addition, as most of the work mentioned in the literature used a two loops control scheme, an inner loop to control the attitude and altitude and an outer loop to control $x$ and $y$ directions, a single loop control technique is used in this thesis.

# Chapter 3

# Attitude Stabilization and Path Tracking Using $H_\infty$ Controller

As mentioned before, quadrotors are underactuated and unstable nonlinear systems; each quadrotor has 6 DOFs and four inputs only to be controlled. Therefore, designing a robust controller is a substantial task to guarantee the quadrotor stability and robustness. In addition, quadrotors are probably affected by external disturbances such as wind disturbance and model parameters uncertainties. The contribution of this chapter is the design of a state feedback controller for attitude stabilisation and the path tracking problem of a UAV quadrotor. The quadrotor attitude is represented by unit quaternion once and then by Euler angles, and external disturbances and model parameter uncertainties are taken into consideration. A robust controller is synthesised via the $H_\infty$ optimal design approach. Solving the nonlinear $H_\infty$ optimal control problem using state feedback is melted down to finding a solution to a HJI. Based on the quadrotor attitude and translation dynamics, appropriate parametrised Lyapunov functions are selected and the corresponding state feedback controllers are derived. Then the parameters are found from a HJI. The resultant state feedback controllers can lead to a closed-loop nonlinear system having $L_2$-gain less than or equal to a constant $\gamma$, and establish the asymptotic stability of the closed-loop nonlinear system without external disturbance.

The controllers were implemented and tested in a MATLAB quadrotor simulater. In simulation, the controllers were tested for various disturbances, including the model parameter uncertainties (mass and inertia) and torque disturbance **d**. The simulation results of the proposed $H_\infty$ controller are discussed later in this Chapter.

For comparison purposes, a $PD^2$ controller obtained in [31] was also tested for attitude stabilisation. For path tracking, an IBS controller was derived, implemented and tested in a MATLAB quadrotor simulater and its stability analysis was achieved based on a selected Lyapunov function. The simulation results are demonstrated in Chapter 5.

## 3.1 $H_\infty$ Suboptimal Control Approach

The major task in the $H_\infty$ optimal control of a nonlinear system is to find the solution of the nonlinear state feedback $H_\infty$ control problem with the presence of disturbance which affects the dynamic feedback measurement. Then the next step is the relation of $L_2$-gain of a nonlinear system with the $H_\infty$ norm. This is because the $H_\infty$ norm is a norm on transfer matrices and only the $L_2$-induced norm if it is translated into time domain (from the input to the output time functions with initial state zero). This norm is appropriate for a nonlinear system and it is called $L_2$-gain of the nonlinear system.

In this section, an overview on the $H_\infty$ suboptimal control approach is summarised for affine nonlinear systems of the form:

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} + k(\mathbf{x})\mathbf{d} \\ \mathbf{y} &= h(\mathbf{x}) \end{aligned} \tag{3.1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is a state vector, $\mathbf{u} \in \mathbb{R}^m$ is an input vector, $\mathbf{y} \in \mathbb{R}^p$ is an output vector, and $\mathbf{d} \in \mathbb{R}^q$ is a disturbance vector. Detailed information on $H_\infty$ control approach can be found in [15].

We assume the existence of an equilibrium $\mathbf{x}_*$, i.e. $f(\mathbf{x}_*) = 0$, and we also assume $h(\mathbf{x}_*) = 0$. Given a smooth state feedback controller,

$$\begin{cases} \mathbf{u} = l(\mathbf{x}) \\ l(\mathbf{x}_*) = 0. \end{cases} \tag{3.2}$$

The $H_\infty$ suboptimal control problem considers the $L_2$-gain from the disturbance $\mathbf{d}$ to the vector of $\mathbf{z} = [\mathbf{y}^T, \mathbf{u}^T]^T$. This problem is defined below.

*Problem* 1. Let $\gamma$ be a fixed nonnegative constant. The closed loop system consisting of the nonlinear system (3.1) and the state feedback controller (3.2) is said to have $L_2$-gain less than or equal to $\gamma$ from $\mathbf{d}$ to $\mathbf{z}$ if

$$\int_0^T \|\mathbf{z}(t)\|^2 dt \leq \gamma^2 \int_0^T \|\mathbf{d}(t)\|^2 dt + K(\mathbf{x}(0)) \tag{3.3}$$

for all $T \geq 0$ and all $\mathbf{d} \in L_2(0, T)$ with initial condition $\mathbf{x}(0)$, where $0 \leq K(\mathbf{x}) < \infty$ and $K(\mathbf{x}_*) = 0$.

For the nonlinear system (3.1) and $\gamma > 0$, define the Hamiltonian $H_\gamma(\mathbf{x}, V(\mathbf{x}))$ as below:

$$\begin{aligned} H_\gamma(\mathbf{x}, V(\mathbf{x})) =& \frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} f(\mathbf{x}) + \frac{1}{2}\frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} \left[\frac{1}{\gamma^2} k(\mathbf{x}) k^T(\mathbf{x}) - g(\mathbf{x}) g^T(\mathbf{x})\right] \frac{\partial^T V(\mathbf{x})}{\partial \mathbf{x}} \\ &+ \frac{1}{2} h^T(\mathbf{x}) h(\mathbf{x}). \end{aligned} \tag{3.4}$$

where $g(\mathbf{x})$ and $k(\mathbf{x})$ are the input and disturbance matrices respectively, and it is obtained in the nonlinear system (3.1).

**Theorem 3.1.** *[15] If there exists a smooth solution $V \geq 0$ to the Hamilton-Jacobi inequality*

$$\begin{aligned} H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq 0 \\ V(\mathbf{x}_*) = 0, \end{aligned} \tag{3.5}$$

*then the closed-loop system for the state feedback controller*

$$\mathbf{u} = -g^T(\mathbf{x})\frac{\partial^T V(\mathbf{x})}{\partial \mathbf{x}} \tag{3.6}$$

*has $L_2$-gain less than or equal to $\gamma$, and $K(\mathbf{x}) = 2V(\mathbf{x})$.*

The nonlinear system (3.1) is called zero-state observable if for any trajectory $\mathbf{x}(t)$ such that $\mathbf{y}(t) = 0, \mathbf{u}(t) = 0, \mathbf{d}(t) = 0$ implies $\mathbf{x}(t) = \mathbf{x}_*$.

*Proposition* 1. [15] If the nonlinear system (3.1) is zero-state observable and there exists a proper solution $V \geq 0$ to the HJI (3.5), then $V(\mathbf{x}) > 0$ for $\mathbf{x}(t) \neq \mathbf{x}_*$ and the closed loop system (3.1), (3.6) with $\mathbf{d} = 0$ is globally asymptotically stable.

## 3.2 Flight Control Based On Quaternion Representation

### 3.2.1 Attitude Stabilisation

The attitude or orientation of a quadrotor is described by a rotation with an angle $\alpha$ above the axis k $\in \mathbb{R}^3$. The corresponding rotation matrix is $R$, which falls in the special orthogonal group of degree three $SO(3) = \{R \in \mathbb{R}^{3\times3} | R^T R = RR^T = I, \det(R) = 1\}$. To describe the orientation of a quadrotor, the quaternion representation is used in this section, which is able to alleviate the singularity problem caused by the Euler angles representation. The dynamic model of a quadrotor including the gyroscope effects $G(\omega)$ can be written as:

$$\begin{cases} \dot{\mathbf{p}} = \mathbf{v} \\ \dot{\mathbf{v}} = -g\mathbf{e} + \frac{f}{m} R_q \mathbf{e} \\ \begin{bmatrix} \dot{q}_0 \\ \dot{\mathbf{q}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\mathbf{q}^T\omega \\ (q_0 I + S(\mathbf{q}))\,\omega \end{bmatrix} \\ J\dot{\omega} = -S(\omega)J\omega - G(\omega) + \tau_q \end{cases} . \tag{3.7}$$

where $m$ is the quadrotor mass, $\omega = [\omega_x, \omega_y, \omega_z]^T$ is the angular velocity in the body frame, $J$ is the $3 \times 3$ diagonal matrix representing three inertial moments in the body frame, $\tau_q$ is the torque vector applied on the quadrotor, the unit quaternion $[q_0, q_1, q_2, q_3]^T = [q_0, \mathbf{q}^T]^T$ where $\mathbf{q} = [q_1, q_2, q_3]^T$ is the vector part and $q_0$ is the scalar part of the quaternion, $\mathbf{v} = [v_x, v_y, v_z]^T$ is the linear velocity, $\mathbf{p} = [x, y, z]^T$ is the position vector, the vector $\mathbf{e} = [0, 0, 1]^T$, and $I$ is the $3 \times 3$ unit matrix. The rotation matrix $R_q$ is related to the unit quaternion through the Rodrigues formula

$$R_q = ({q_0}^2 - \mathbf{q}^T\mathbf{q})I + 2\mathbf{q}\mathbf{q}^T + 2q_0 S(\mathbf{q})$$

and $S$ is the skew-symmetric cross product matrix

$$S(\mathbf{q}) = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}.$$

More details of the quadrotor model derivation based on unit quaternion representation are described in Appendix B.

The essential purpose of the attitude stabilisation is to design a controller $\tau$ which can asymptotically drive the quadrotor to an equilibrium point from an initial attitude and the effect of added disturbances tends to disappear. The equilibrium point for the attitude stabilisation problem is the point with $R_q = I, \omega = 0$ or the two equilibrium points (desired point) in quaternion representation ($q_{0d} = \pm 1, \mathbf{q}_d = [0, 0, 0]^T, \omega_d = [0, 0, 0]^T$), which represent the same physical point. In mathematical form, the attitude stabilisation controller should satisfy the following conditions:

$$\begin{cases} \lim_{t\to\infty} \tilde{q}_0 = \lim_{t\to\infty}(q_{0d} - q_0) = 0 \\ \lim_{t\to\infty} \tilde{\mathbf{q}} = \lim_{t\to\infty}(\mathbf{q}_d - \mathbf{q}) = 0 \\ \lim_{t\to\infty} \tilde{\omega} = \lim_{t\to\infty}(\omega_d - \omega) = 0 \end{cases} . \tag{3.8}$$

The total reference thrust from the propellers is considered to be

$$f = mg$$

and the rotational part of the quadrotor dynamic model is

$$\left\{ \begin{array}{l} \begin{bmatrix} \dot{q}_0 \\ \dot{\mathbf{q}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\mathbf{q}^T \omega \\ (q_0 I + S(\mathbf{q}))\,\omega \end{bmatrix} \\ J\dot{\omega} = -S(\omega)J\omega - G(\omega) + \tau_q \end{array} \right. . \tag{3.9}$$

Now we consider a robust control approach to the attitude stabilisation problem. When considering $\mathbf{d}$ as the torque disturbance, then $\mathbf{d} = [d_{\omega_x}, d_{\omega_y}, d_{\omega_z}]^T$ is applied to the nonlinear system (3.9). Let $\mathbf{x} = [q_0, \mathbf{q}^T, \omega^T]^T$ and $\mathbf{u} = G(\tilde{\omega}) - \tau_q$. The attitude system (3.9) with the disturbance $\mathbf{d}$ can be written into an affine nonlinear form:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} + k(\mathbf{x})\mathbf{d} \tag{3.10}$$

where

$$\begin{aligned} f(\mathbf{x}) &= \begin{bmatrix} \frac{1}{2}\tilde{\mathbf{q}}^T\tilde{\omega} \\ -\frac{1}{2}(\tilde{q}_0 I + S(\tilde{\mathbf{q}}))\,\tilde{\omega} \\ J^{-1}S(\tilde{\omega})J\tilde{\omega} \end{bmatrix} \\ g(\mathbf{x}) &= \begin{bmatrix} 0_{1\times 3} \\ 0_{3\times 3} \\ J^{-1} \end{bmatrix} \\ k(\mathbf{x}) &= \begin{bmatrix} 0_{1\times 3} \\ 0_{3\times 3} \\ J^{-1} \end{bmatrix} . \end{aligned}$$

### 3.2.2 $H_\infty$ Suboptimal Controller

The $H_\infty$ suboptimal controller is designed for the attitude stabilisation problem in this section. The following form of $V$ is suggested for the attitude stabilisation model (3.9):

$$V(\mathbf{x}) = \frac{1}{2}\left[\begin{array}{cc} \tilde{\mathbf{q}}^T & \tilde{\omega}^T \end{array}\right]\left[\begin{array}{cc} 0_{3\times3} & JK_q \\ JK_q & JK_\omega \end{array}\right]\left[\begin{array}{c} \tilde{\mathbf{q}} \\ \tilde{\omega} \end{array}\right] + 2C_q(1-\tilde{q}_0) \tag{3.11}$$

where diagonal matrices $K_q > 0$ and $K_\omega > 0$ are the proportional and derivative gains for translational and rotational parts. $C_q > 0$ is constant, and its partial derivative is

$$\frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} = [-2C_q \quad JK_q\tilde{\omega} \quad JK_q\tilde{\mathbf{q}} + JK_\omega\tilde{\omega}].$$

Accordingly the controller is

$$\begin{aligned} \mathbf{u} &= -g^T(\mathbf{x})\frac{\partial^T V(\mathbf{x})}{\partial \mathbf{x}} \\ &= -\left[\begin{array}{c} K_q\tilde{\mathbf{q}} + K_\omega\tilde{\omega} \end{array}\right]. \end{aligned} \tag{3.12}$$

For the equilibrium points $\mathbf{x}_* = [q_0, \mathbf{q}^T, \omega]^T = [\pm1, 0_{1\times3}, 0_{1\times3}]^T$, the following diagonal weighting matrices are chosen $W_2 = W_2^T > 0, W_4 = W_4^T > 0$;

$$h(\mathbf{x}) = [\sqrt{W_2}\tilde{\mathbf{q}}^T \quad \sqrt{W_4}\tilde{\omega}^T]^T$$

which satisfies $h(\mathbf{x}_*) = 0$. When the rotational angle $\alpha$ is taken between $-\pi$ and $\pi$, we have $0 \le q_0 \le 1$ and this will exclude $[-1, 0_{1\times3}, 0_{1\times3}]^T$ from the equilibrium points. For the equilibrium point, $\mathbf{x}_* = [1, 0_{1\times3}, 0_{1\times3}]^T$. And we know

$$V(\mathbf{x}_*) = 0.$$

Now the attitude stabilisation problem of the quadrotor under the disturbance $\mathbf{d}$ is defined below.

*Problem* 2. Find the parameters $K_q, K_\omega, C_q$ in order to enable the closed-loop system (3.10) with the above controller $\mathbf{u}$ (3.12) to have $L_2$-gain less than or equal to $\gamma$.

Next we want to show our main result in the following theorem.

**Theorem 3.2.** *If the following conditions are satisfied, the closed-loop system* (3.10) *with the above controller* $\mathbf{u}$ (3.12) *has* $L_2$*-gain less than or equal to* $\gamma$*. And the closed loop system* (3.10),(3.12) *with* $\mathbf{d} = 0$ *is globally asymptotically stable.*

$$\begin{aligned}
C_q K_\omega &\geq J K_q^2 \\
C_q &= K_q K_\omega \left( \frac{1}{\gamma^2} - 1 \right) \\
\|K_q\|^2 &\geq \frac{\gamma^2 \|W_2\|}{\gamma^2 - 1} && (3.13) \\
\|K_\omega\|^2 &\geq \frac{\gamma^2 (\|W_4\| - \sqrt{3}\|J\|\|K_q\|)}{\gamma^2 - 1} && (3.14) \\
\|W_2\| &> 0; \ \|W_4\| > 0
\end{aligned}$$

*Proof.* With the given conditions, we need to show (1) $V(\mathbf{x}) \geq 0$ and (2) the Hamiltonian $H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq 0$. Then the first part of the theorem can be proved by using Theorem 3.1.

(1) Since

$$\begin{aligned}
2(1 - \tilde{q}_0) &= (1 - \tilde{q}_0)^2 + \tilde{\mathbf{q}}^T \tilde{\mathbf{q}} \\
&\geq \tilde{\mathbf{q}}^T \tilde{\mathbf{q}},
\end{aligned}$$

then

$$\begin{aligned}
V(\mathbf{x}) &= \frac{1}{2}\left[\begin{array}{cc} \tilde{\mathbf{q}}^T & \tilde{\omega}^T \end{array}\right]\left[\begin{array}{cc} 0_{3\times 3} & JK_q \\ JK_q & JK_\omega \end{array}\right]\left[\begin{array}{c} \tilde{\mathbf{q}} \\ \tilde{\omega} \end{array}\right] + 2C_q(1-\tilde{q}_0) \\
&\geq \frac{1}{2}\left[\begin{array}{cc} \tilde{\mathbf{q}}^T & \tilde{\omega}^T \end{array}\right]\left[\begin{array}{cc} 0_{3\times 3} & JK_q \\ JK_q & JK_\omega \end{array}\right]\left[\begin{array}{c} \tilde{\mathbf{q}} \\ \tilde{\omega} \end{array}\right] + C_q\tilde{\mathbf{q}}^T\tilde{\mathbf{q}} \\
&= \frac{1}{2}\left[\begin{array}{cc} \tilde{\mathbf{q}}^T & \tilde{\omega}^T \end{array}\right]\left[\begin{array}{cc} C_qI & JK_q \\ JK_q & JK_\omega \end{array}\right]\left[\begin{array}{c} \tilde{\mathbf{q}} \\ \tilde{\omega} \end{array}\right].
\end{aligned}$$

Thus the condition for $V(\mathbf{x}) \geq 0$ is

$$C_qK_\omega \geq JK_q^2.$$

(2)

$$\begin{aligned}
H_\gamma(\mathbf{x}, V(\mathbf{x})) =& -\tilde{\mathbf{q}}^TC_q\tilde{\omega} - \frac{1}{2}\tilde{\omega}^TJK_q(\tilde{q}_0I + S(\tilde{\mathbf{q}}))\tilde{\omega} + \tilde{\mathbf{q}}^TK_qS(\tilde{\omega})J\tilde{\omega} + \tilde{\omega}^TK_\omega S(\tilde{\omega})J\tilde{\omega} \\
&+ \frac{1}{2}\left(\frac{1}{\gamma^2}-1\right)\|K_q\tilde{\mathbf{q}} + K_\omega\tilde{\omega}\|^2 + \frac{1}{2}\|W_2\|\|\tilde{\mathbf{q}}\|^2 + \frac{1}{2}\|W_4\|\|\tilde{\omega}\|^2.
\end{aligned}$$

By choosing

$$C_q = K_qK_\omega\left(\frac{1}{\gamma^2}-1\right),$$

then

$$\begin{aligned}
H_\gamma(\mathbf{x}, V(\mathbf{x})) =& -\frac{1}{2}\tilde{\omega}^TJK_q(\tilde{q}_0I + S(\tilde{\mathbf{q}}))\tilde{\omega} + \tilde{\mathbf{q}}^TK_qS(\tilde{\omega})J\tilde{\omega} + \tilde{\omega}^TK_\omega S(\tilde{\omega})J\tilde{\omega} \\
&+ \frac{1}{2}\left(\frac{1}{\gamma^2}-1\right)\left(\|K_q\|^2\|\tilde{\mathbf{q}}\|^2 + \|K_\omega\|^2\|\tilde{\omega}\|^2\right) + \frac{1}{2}\|W_2\|\|\tilde{\mathbf{q}}\|^2 \\
&+ \frac{1}{2}\|W_4\|\|\tilde{\omega}\|^2.
\end{aligned}$$

By using $\|S(\tilde{\omega}))\| = \|\tilde{\omega}\|$, $\|(\tilde{q}_0I+S(\tilde{\mathbf{q}}))\| \leq \sqrt{3}$, $|\tilde{\omega}^TJK_q(\tilde{q}_0I+S(\tilde{\mathbf{q}}))\tilde{\omega}| \leq \|K_q\|\|J\|\|\tilde{\omega}\|^2\|(\tilde{q}_0I+S(\tilde{\mathbf{q}}))\|$, $\tilde{\mathbf{q}}^TK_qS(\tilde{\omega})J\tilde{\omega} = 0$ and $\tilde{\omega}^TK_\omega S(\tilde{\omega})J\tilde{\omega} = 0$, we have

$$\begin{aligned}H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq & \frac{-\sqrt{3}}{2}\|K_q\|\|J\|\|\tilde{\omega}\|^2 + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)(\|K_q\|^2\|\tilde{\mathbf{q}}\|^2 + \|K_\omega\|^2\|\tilde{\omega}\|^2) \\ & + \frac{1}{2}\|W_2\|\|\tilde{\mathbf{q}}\|^2 + \frac{1}{2}\|W_4\|\|\tilde{\omega}\|^2.\end{aligned}$$

Thus, the conditions for $H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq 0$ are

$$\begin{aligned}\frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_q\|^2 + \frac{1}{2}\|W_2\| \leq 0 \\ \frac{-\sqrt{3}}{2}\|J\|\|K_q\| + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_\omega\|^2 + \frac{1}{2}\|W_4\| \leq 0,\end{aligned}$$

i.e.

$$\begin{aligned}\|K_q\|^2 &\geq \frac{\gamma^2\|W_2\|}{\gamma^2 - 1} \\ \|K_\omega\|^2 &\geq \frac{\gamma^2(\|W_4\| - \sqrt{3}\|J\|\|K_q\|)}{\gamma^2 - 1}.\end{aligned}$$

It is trivial to show that the nonlinear system (3.10) is zero-state observable. Further, due to the fact that $V(\mathbf{x}) \geq 0$ and it is a proper function (i.e. for each $\beta > 0$ the set $\{x : 0 \leq V(x) \leq \beta\}$ is compact), the closed loop system (3.10),(3.12) with $\mathbf{d} = 0$ is globally asymptotically stable according to Proposition 1. This proves the second part of the theorem. □

Finally from $\mathbf{u}$, we can find $\tau_q$:

$$\begin{aligned}\tau_q &= -\mathbf{u} + G(\tilde{\omega}) \\ &= K_q\tilde{\mathbf{q}} + K_\omega\tilde{\omega} + G(\tilde{\omega}).\end{aligned} \tag{3.15}$$

### 3.2.3 Path Tracking Control

The full mathematical model of the quadrotor (3.7) is used to control the quadrotor to track 3D paths. In what follows, an individual controller is designed using $H_\infty$

to track various scenarios by using tracking errors as inputs and providing the propellers speed as outputs.

Then the tracking errors can be written as:

$$\begin{aligned}
\tilde{\mathbf{p}} &= \mathbf{p}_d - \mathbf{p} \\
\tilde{\mathbf{v}} &= \mathbf{v}_d - \mathbf{v} \\
\begin{bmatrix} \tilde{q}_0 \\ \tilde{\mathbf{q}} \end{bmatrix} &= \begin{bmatrix} q_{0d} - q_0 \\ \mathbf{q}_d - \mathbf{q} \end{bmatrix} \\
\tilde{\omega} &= \omega_d - \omega.
\end{aligned}$$

Then equation (3.7) can be rewritten in an error form as:

$$\begin{cases}
\dot{\tilde{\mathbf{p}}} = \tilde{\mathbf{v}} \\
\dot{\tilde{\mathbf{v}}} = g\mathbf{e} - \frac{f}{m} R_q \mathbf{e} \\
\begin{bmatrix} \dot{\tilde{q}}_0 \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \tilde{\mathbf{q}}^T \tilde{\omega}_F \\ -\left(\tilde{q}_0 I + S(\tilde{\mathbf{q}})\right) \tilde{\omega} \end{bmatrix} \\
J\dot{\tilde{\omega}} = S(\tilde{\omega}) J \tilde{\omega} + G(\tilde{\omega}) - \tau_q
\end{cases} . \tag{3.16}$$

The control aim is to asymptotically drive the quadrotor towards the desired position $\mathbf{p}_d$ from an initial position with the effect of added disturbances tending to disappear and changed parameters tending to be recovered by satisfying the following conditions:

$$\begin{cases}
\lim_{t\to\infty} \tilde{\mathbf{p}} = \lim_{t\to\infty} (\mathbf{p}_d - \mathbf{p}) = 0 \\
\lim_{t\to\infty} \tilde{q}_0 = \lim_{t\to\infty} (q_{0d} - q_0) = 0 \\
\lim_{t\to\infty} \tilde{q}_3 = \lim_{t\to\infty} (q_{3d} - q_3) = 0.
\end{cases} \tag{3.17}$$

Now we consider the robust control approach to the path tracking problem when considering $\mathbf{d} = [\mathbf{d}_\mathbf{v}^T, \mathbf{d}_\omega^T]^T = [d_{vx}, d_{vy}, d_{vz}, d_{\omega_x}, d_{\omega_y}, d_{\omega_z}]^T$ as the disturbance applied to the nonlinear system (3.16). Those disturbances are used here to model the changes of mass and moment, and the wind disturbances.

Let

$$\mathbf{x} = \begin{bmatrix} \tilde{\mathbf{p}} \\ \tilde{q}_0 \\ \tilde{\mathbf{q}} \\ \tilde{\mathbf{v}} \\ \tilde{\omega} \end{bmatrix}$$

$$\mathbf{u} = \begin{bmatrix} g\mathbf{e} - \frac{f}{m} R_q \mathbf{e} \\ G(\tilde{\omega}) - \tau_q \end{bmatrix}.$$

The dynamic system (3.16) with the disturbance **d** can be written into an affine nonlinear form:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} + k(\mathbf{x})\mathbf{d} \tag{3.18}$$

where

$$f(\mathbf{x}) = \begin{bmatrix} \tilde{\mathbf{v}} \\ \frac{1}{2}\tilde{\mathbf{q}}^T \tilde{\omega} \\ -\frac{1}{2}\left(\tilde{q}_0 I + S(\tilde{\mathbf{q}})\right)\tilde{\omega} \\ 0_{3\times 1} \\ J^{-1} S(\tilde{\omega}) J \tilde{\omega} \end{bmatrix}$$

$$g(\mathbf{x}) = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} \\ 0_{1\times 3} & 0_{1\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \\ I & 0_{3\times 3} \\ 0_{3\times 3} & J^{-1} \end{bmatrix}$$

$$k(\mathbf{x}) = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} \\ 0_{1\times 3} & 0_{1\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \\ I & 0_{3\times 3} \\ 0_{3\times 3} & J^{-1} \end{bmatrix}.$$

### 3.2.4 $H_\infty$ Suboptimal Path Tracking Controller

The $H_\infty$ suboptimal controller is designed for the path tracking problem in this section. The following form of $V$ is suggested for the dynamic model (3.18):

$$V(\mathbf{x}) = \frac{1}{2}\begin{bmatrix} \tilde{\mathbf{p}}^T & \tilde{\mathbf{q}}^T & \tilde{\mathbf{v}}^T & \tilde{\omega}^T \end{bmatrix} \begin{bmatrix} C_pI & 0_{3\times3} & K_p & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & JK_q \\ K_p & 0_{3\times3} & K_v & 0_{3\times3} \\ 0_{3\times3} & JK_q & 0_{3\times3} & JK_\omega \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}} \\ \tilde{\mathbf{q}} \\ \tilde{\mathbf{v}} \\ \tilde{\omega} \end{bmatrix} + 2C_q(1-\tilde{q}_0). \tag{3.19}$$

where diagonal matrices $K_p > 0, K_q > 0, K_v > 0, K_\omega > 0$ are the proportional and derivative gains for translational and rotational parts. $C_p > 0, C_q > 0$ are constants. And

$$\frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} C_p\tilde{\mathbf{p}} + K_p\tilde{\mathbf{v}} & -2C_q & JK_q\tilde{\omega} & K_p\tilde{\mathbf{p}} + K_v\tilde{\mathbf{v}} & JK_q\tilde{\mathbf{q}} + JK_\omega\tilde{\omega} \end{bmatrix}.$$

Accordingly the controller is

$$\begin{aligned} \mathbf{u} &= -g^T(\mathbf{x})\frac{\partial^T V(\mathbf{x})}{\partial \mathbf{x}} \\ &= -\begin{bmatrix} K_p\tilde{\mathbf{p}} + K_v\tilde{\mathbf{v}} \\ K_q\tilde{\mathbf{q}} + K_\omega\tilde{\omega} \end{bmatrix}. \end{aligned} \tag{3.20}$$

The following diagonal weighting matrices are chosen $W_1 > 0$, $W_2 > 0$, $W_3 > 0$ and $W_4 > 0$;

$$h(\mathbf{x}) = [\sqrt{W_1}\tilde{\mathbf{p}}^T \quad \sqrt{W_2}\tilde{\mathbf{q}}^T \quad \sqrt{W_3}\tilde{\mathbf{v}}^T \quad \sqrt{W_4}\tilde{\omega}^T\ ]^T$$

which satisfies $h(\mathbf{x}_*) = 0$, where the equilibrium point $\mathbf{x}_* = [0_{1\times3}, 1, 0_{1\times3}, 0_{1\times3}, 0_{1\times3}]^T$. And we know

$$V(\mathbf{x}_*) = 0.$$

Now the path tracking problem of the quadrotor under the disturbance $\mathbf{d}$ is defined below.

*Problem* 3. Given the equilibrium point $\mathbf{x}_*$, find the parameters $K_p$, $K_q$, $K_v$, $K_\omega$, $C_p$, $C_q$ in order to enable the closed-loop system (3.18) with the above controller $\mathbf{u}$ (3.20) to have $L_2$-gain less than or equal to $\gamma$.

Next we want to show our main result in the following theorem.

**Theorem 3.3.** *If the following conditions are satisfied, the closed-loop system* (3.18) *with the above controller* $\mathbf{u}$ (3.20) *has* $L_2$*-gain less than or equal to* $\gamma$*. And the closed loop system* (3.18), (3.20) *with* $\mathbf{d} = 0$ *is asymptotically locally stable for the equilibrium point* $\mathbf{x}_*$.

$$\begin{aligned}
C_pC_q &\geq 0 \\
C_pK_v &\geq K_p^2 \\
C_pC_qK_vK_\omega &\geq C_pJK_q^2K_v - JK_q^2K_p^2 + C_qK_p^2K_\omega \\
C_p &= K_pK_v\left(1 - \frac{1}{\gamma^2}\right) \\
C_q &= K_qK_\omega\left(\frac{1}{\gamma^2} - 1\right) \\
\|K_p\|^2 &\geq \frac{\gamma^2\|W_1\|}{\gamma^2 - 1} && (3.21) \\
\|K_q\|^2 &\geq \frac{\gamma^2\|W_2\|}{\gamma^2 - 1} && (3.22) \\
\|K_v\|^2 &\geq \frac{\gamma^2(\|W_3\| + 2\|K_p\|)}{\gamma^2 - 1} && (3.23) \\
\|K_\omega\|^2 &\geq \frac{\gamma^2(\|W_4\| - \sqrt{3}\|J\|\|K_q\|)}{\gamma^2 - 1} && (3.24) \\
\|W_1\| &> 0; \|W_2\| > 0; \|W_3\| > 0; \|W_4\| > 0.
\end{aligned}$$

*Proof.* With the given conditions, we need to show (1) $V(\mathbf{x}) \geq 0$ and (2) the Hamiltonian $H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq 0$. Then the first part of the theorem can be proved by using Theorem 3.1.

(1) Since

$$\begin{aligned} 2(1-\tilde{q}_0) &= (1-\tilde{q}_0)^2 + \tilde{\mathbf{q}}^T\tilde{\mathbf{q}} \\ &\geq \tilde{\mathbf{q}}^T\tilde{\mathbf{q}}, \end{aligned}$$

then

$$V(\mathbf{x}) \geq \frac{1}{2}\begin{bmatrix} \tilde{\mathbf{p}}^T & \tilde{\mathbf{q}}^T & \tilde{\mathbf{v}}^T & \tilde{\omega}^T \end{bmatrix} \begin{bmatrix} C_pI & 0_{3\times3} & K_p & 0_{3\times3} \\ 0_{3\times3} & C_qI & 0_{3\times3} & JK_q \\ K_p & 0_{3\times3} & K_v & 0_{3\times3} \\ 0_{3\times3} & JK_q & 0_{3\times3} & JK_\omega \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}} \\ \tilde{\mathbf{q}} \\ \tilde{\mathbf{v}} \\ \tilde{\omega} \end{bmatrix}.$$

Thus the conditions for $V(\mathbf{x}) \geq 0$ are

$$\begin{aligned} C_pC_q &\geq 0 \\ C_pK_v &\geq K_p^2 \\ C_pC_qK_vK_\omega &\geq C_pJK_q^2K_v - JK_q^2K_p^2 + C_qK_p^2K_\omega \end{aligned}$$

(2)

$$\begin{aligned} H_\gamma(\mathbf{x}, V(\mathbf{x})) &= \tilde{\mathbf{p}}^TC_p\tilde{\mathbf{v}} - \tilde{\mathbf{q}}^TC_q\tilde{\omega} + \tilde{\mathbf{v}}^TK_p\tilde{\mathbf{v}} - \frac{1}{2}\tilde{\omega}^TJK_q(\tilde{q}_0I + S(\tilde{\mathbf{q}}))\tilde{\omega} \\ &+ \tilde{\mathbf{q}}^TK_qS(\tilde{\omega})J\tilde{\omega} + \tilde{\omega}^TK_\omega S(\tilde{\omega})J\tilde{\omega} + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_p\tilde{\mathbf{p}} + K_v\tilde{\mathbf{v}}\|^2 \\ &+ \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_q\tilde{\mathbf{q}} + K_\omega\tilde{\omega}\|^2 + \frac{1}{2}\|W_1\|\|\tilde{\mathbf{p}}\|^2 + \frac{1}{2}\|W_2\|\|\tilde{\mathbf{q}}\|^2 \\ &+ \frac{1}{2}\|W_3\|\|\tilde{\mathbf{v}}\|^2 + \frac{1}{2}\|W_4\|\|\tilde{\omega}\|^2. \end{aligned}$$

By choosing

$$\begin{aligned} C_p &= K_pK_v\left(1 - \frac{1}{\gamma^2}\right) \\ C_q &= K_qK_\omega\left(\frac{1}{\gamma^2} - 1\right), \end{aligned}$$

then

$$\begin{aligned}H_\gamma(\mathbf{x}, V(\mathbf{x})) = {} & \tilde{\mathbf{v}}^T K_p \tilde{\mathbf{v}} + \tilde{\mathbf{q}}^T K_q S(\tilde{\omega}) J \tilde{\omega} - \frac{1}{2}\tilde{\omega}^T J K_q (\tilde{q}_0 I + S(\tilde{\mathbf{q}}))\tilde{\omega} \\ & + \tilde{\omega}^T K_\omega S(\tilde{\omega}) J \tilde{\omega} + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)(\|K_p\|^2\|\tilde{\mathbf{p}}\|^2 + \|K_v\|^2\|\tilde{\mathbf{v}}\|^2) \\ & + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)(\|K_q\|^2\|\tilde{\mathbf{q}}\|^2 + \|K_\omega\|^2\|\tilde{\omega}\|^2) + \frac{1}{2}\|W_1\|\|\tilde{\mathbf{p}}\|^2 \\ & + \frac{1}{2}\|W_2\|\|\tilde{\mathbf{q}}\|^2 + \frac{1}{2}\|W_3\|\|\tilde{\mathbf{v}}\|^2 + \frac{1}{2}\|W_4\|\|\tilde{\omega}\|^2.\end{aligned}$$

By using $\|S(\tilde{\omega}))\| = \|\tilde{\omega}\|$, $|\tilde{\mathbf{v}}^T K_p \tilde{\mathbf{v}}| \leq \|K_p\|\|\tilde{\mathbf{v}}\|^2$, $\|(\tilde{q}_0 I + S(\tilde{\mathbf{q}}))\| \leq \sqrt{3}$, $|\tilde{\omega}^T J K_q (\tilde{q}_0 I + S(\tilde{\mathbf{q}}))\tilde{\omega}| \leq \|K_q\|\|J\|\|\tilde{\omega}\|^2\|(\tilde{q}_0 I + S(\tilde{\mathbf{q}}))\|$, $\tilde{\mathbf{q}}^T K_q S(\tilde{\omega}) J \tilde{\omega} = 0$ and $\tilde{\omega}^T K_\omega S(\tilde{\omega}) J \tilde{\omega} = 0$, we have

$$\begin{aligned}H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq {} & \frac{-\sqrt{3}}{2}\|K_q\|\|J\|\|\tilde{\omega}\|^2 + \|K_p\|\|\tilde{\mathbf{v}}\|^2 + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right) \\ & (\|K_p\|^2\|\tilde{\mathbf{p}}\|^2 + \|K_v\|^2\|\tilde{\mathbf{v}}\|^2) + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)(\|K_q\|^2\|\tilde{\mathbf{q}}\|^2 \\ & + \|K_\omega\|^2\|\tilde{\omega}\|^2) + \frac{1}{2}\|W_1\|\|\tilde{\mathbf{p}}\|^2 + \frac{1}{2}\|W_2\|\|\tilde{\mathbf{q}}\|^2 + \frac{1}{2}\|W_3\|\|\tilde{\mathbf{v}}\|^2 \\ & + \frac{1}{2}\|W_4\|\|\tilde{\omega}\|^2.\end{aligned}$$

Thus, the conditions for $H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq 0$ are

$$\begin{aligned}& \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_p\|^2 + \frac{1}{2}\|W_1\| \leq 0 \\ & \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_q\|^2 + \frac{1}{2}\|W_2\| \leq 0 \\ & \|K_p\| + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_v\|^2 + \frac{1}{2}\|W_3\| \leq 0 \\ & \frac{-\sqrt{3}}{2}\|J\|\|K_q\| + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_\omega\|^2 + \frac{1}{2}\|W_4\| \leq 0;\end{aligned}$$

i.e.

$$\begin{aligned}
\|K_p\|^2 &\geq \frac{\gamma^2\|W_1\|}{\gamma^2-1} \\
\|K_q\|^2 &\geq \frac{\gamma^2\|W_2\|}{\gamma^2-1} \\
\|K_v\|^2 &\geq \frac{\gamma^2(\|W_3\|+2\|K_p\|)}{\gamma^2-1} \\
\|K_\omega\|^2 &\geq \frac{\gamma^2(\|W_4\|-\sqrt{3}\|J\|\|K_q\|)}{\gamma^2-1}.
\end{aligned}$$

It is trivial to show that the nonlinear system (3.18) is zero-state observable for the equilibrium point $\mathbf{x}_*$. Further, due to the fact that $V(\mathbf{x}) \geq 0$ and it is a proper function (i.e. for each $\beta > 0$ the set $\{x : 0 \leq V(x) \leq \beta\}$ is compact), the closed-loop system (3.18), (3.20) with $\mathbf{d} = 0$ is asymptotically locally stable for the equilibrium point $\mathbf{x}_*$ according to Proposition 1. This proves the second part of the theorem. □

*Remark* 1. It should be noted that the proof of Theorem 3.3, $\lim_{t\to\infty}\tilde{\mathbf{p}} = 0$, $\lim_{t\to\infty}\tilde{\mathbf{q}} = 0$, $\lim_{t\to\infty}\tilde{\mathbf{v}} = 0$ and $\lim_{t\to\infty}\tilde{\omega} = 0$, meet the conditions of (3.17).

Finally from $\mathbf{u}$, we can have

$$\begin{aligned}
\mathbf{u} &= \begin{bmatrix} g\mathbf{e} - \frac{f}{m}R_q\mathbf{e} \\ G(\tilde{\omega}) - \tau_q \end{bmatrix} \\
&= -\begin{bmatrix} K_p\tilde{\mathbf{p}} + K_v\tilde{\mathbf{v}} \\ K_q\tilde{\mathbf{q}} + K_\omega\tilde{\omega} \end{bmatrix}.
\end{aligned}$$

Then the total force and the torque vector are applied to the quadrotor, $f$ and $\tau_q \in \mathbb{R}^3$,

$$f = (k_{pz}\tilde{z} + k_{vz}\tilde{v}_z + g)\frac{m}{q_0^2 - q_1^2 - q_2^2 + q_3^2} \tag{3.25}$$

$$\tau_q = K_q\tilde{\mathbf{q}} + K_\omega\tilde{\omega} + G(\tilde{\omega}). \tag{3.26}$$

| **Symbol** | **Definition** | **Value** | **Units** |
|---|---|---|---|
| $J_x$ | Roll Inertia | $4.4 \times 10^-3$ | $kg.m^2$ |
| $J_y$ | Pitch Inertia | $4.4 \times 10^-3$ | $kg.m^2$ |
| $J_z$ | Yaw Inertia | $8.8 \times 10^-3$ | $kg.m^2$ |
| $m$ | Mass | 0.5 | $kg$ |
| $g$ | Gravity | 9.81 | $m/s^2$ |
| $l$ | Arm Length | 0.17 | $m$ |
| $J_r$ | Rotor Inertia | $4.4 \times 10^-5$ | $kg.m^2$ |

Table 3.1: Quadrotor Parameters

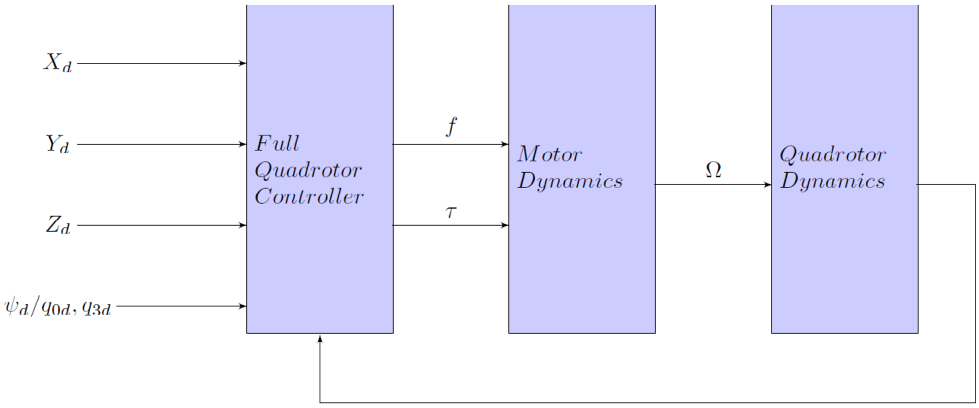

Figure 3.1: One Loop Control Block Diagram

## 3.3 Simulations

In order to determine the efficiency of the proposed controller, a MATLAB quadrotor simulater is used to test it numerically. The one loop control block diagram is used to control the overall quadrotor dynamics, which is illustrated in Figure 3.1. In this case, the $x$ and $y$ position errors are used directly in the torque control law of $q_1$ and $q_2$/ $\theta$ and $\varphi$. The design parameters of the quadrotor used in the simulater are listed in Table 3.1.

### 3.3.1 Attitude Stability

The aim is to control the attitude of the quadrotor to the equilibrium point. The proposed controller was tested for various disturbances, including the model parameter uncertainties (mass and inertia) and torque disturbance $\mathbf{d}$. The constant $\gamma$ was chosen to be $\gamma = 1.05$ and the norm of the two weighting matrices was chosen to be $W_2 = diag(0.0235, 0.0235, 0.0009)$ and $W_4 = diag(0.0043, 0.0043, 0.00156)$. Under these parameters the norm of feedback control matrices can be obtained by solving the conditions in (3.13) and (3.14) to be $K_q = diag(0.5, 0.5, 0.095)$ and $K_\omega = diag(0.07, 0.07, 0.025)$.

Figure 3.2 shows the performance of quaternion components using the $H_\infty$ controller under the action of the designed torque compared with that of the $H_\infty$ controller with the torque disturbances shown in (3.27) and (3.28) and ±30% model parameter uncertainties. It can be seen that the $H_\infty$ controller performance achieved a zero steady-state error in less than two seconds and it could reject the disturbances and cover the change in model parameter uncertainties.

$$\mathbf{d}_1 = 0.01 + 0.01\sin(0.024\pi t) + 0.05\sin(1.32\pi t) \tag{3.27}$$

$$\mathbf{d}_2 = 0.1 + 0.1\cos(0.24\pi t) + 0.5\sin(1.32\pi t). \tag{3.28}$$

The performance of the angular velocities under the use of the $H_\infty$ controller and the $H_\infty$ controller with the effect of disturbances and model parameter uncertainties is shown in Figure 3.3. It illustrates that the angular velocities performance using the $H_\infty$ controller achieved the stability conditions very fast.

Table 3.2 illustrates the Root Mean Square Error (RMSE) of the quaternion components in all the five circumstances. It is clear that the RMSE values of the quaternion parameters were very small when using the $H_\infty$ controller in the normal condition and with the disturbances, while they slightly increased by 0.0003 with the model parameter uncertainties. In general, $H_\infty$ controller yielded a good result in terms of RMSE value, time-consuming, disturbance rejection and model parameter uncertainties change cover.

### 3.3.2 Path Tracking

Two different paths are tested in order to demonstrate the robustness of the proposed controller. The added disturbances include ±30% of the model parameter

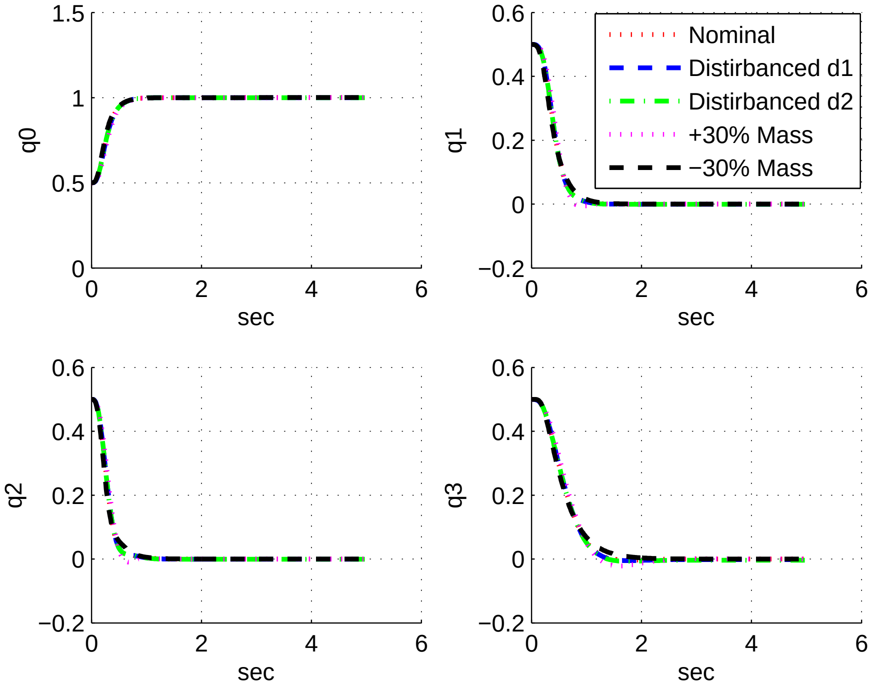

FIGURE 3.2: Quaternion Components under $H_\infty$ Controller

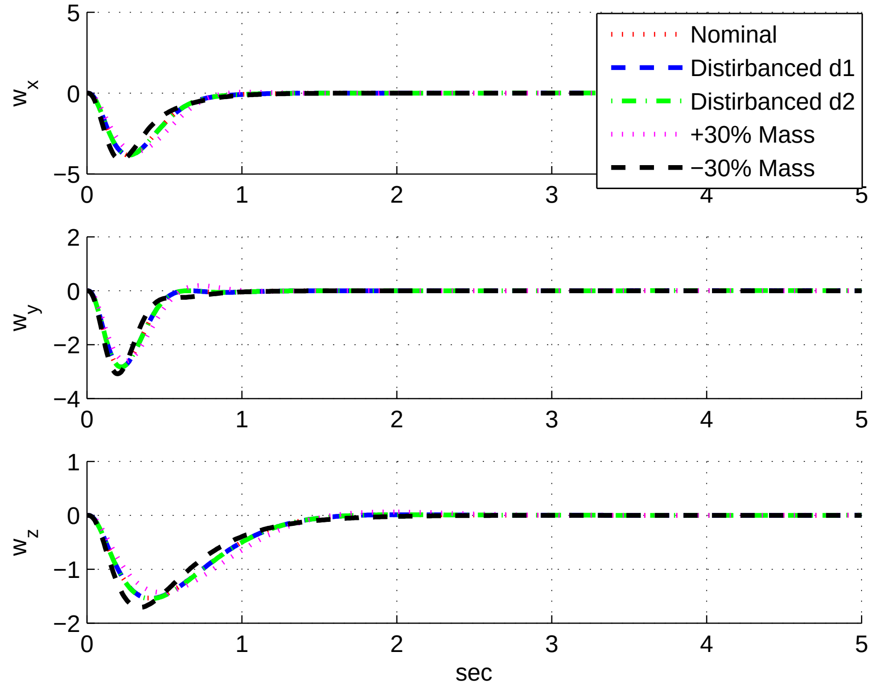

FIGURE 3.3: Angular Velocities under $H_\infty$ Controller

| **RMSE** | $q_0$ | $q_1$ | $q_2$ | $q_3$ |
|---|---|---|---|---|
| $H_\infty$ | 0.0055 | 0.0012 | 0.0034 | 0.0011 |
| $H_\infty + \mathbf{d}_1$ | 0.0056 | 0.0012 | 0.0034 | 0.0012 |
| $H_\infty + \mathbf{d}_2$ | 0.0059 | 0.0012 | 0.0034 | 0.0014 |
| $H_\infty + 30\%$ | 0.0092 | 0.0011 | 0.0037 | 0.0045 |
| $H_\infty - 30\%$ | 0.0110 | 0.0013 | 0.0031 | 0.0068 |

TABLE 3.2: Quaternion Parameter RMSE Values under $H_\infty$ Controller

uncertainties (mass and inertia) and a force disturbance.

In the first path, the initial conditions of the quadrotor are $\mathbf{p}(0) = [0, 0.5, 0]^T$ metres and $Q(0) = [-1, 0, 0, 0]^T$, and the desired path is

$$\begin{cases} x_d = 0.5\sin(t\pi/2) \quad ; \quad y_d = 0.5\cos(t\pi/2) \\ z_d = 1 + 0.1t \quad ; \quad q_{3d} = 0 \end{cases}.$$

The constant $\gamma$ is chosen to be $\gamma = 1.05$ and the weighting matrices are chosen to be $W_{1z} = 1150$, $W_2 = diag(0.0235, 0.0235, 0.0009)$, $W_{3z} = 10$ and $W_4 = diag(0.0043, 0.0043, 0.00156)$. Under these parameters the feedback control matrices can be obtained by solving the conditions in (3.21), (3.22), (3.23) and (3.24) to be $k_{pz} = 112$, $k_{vz} = 50$, $K_q = diag(0.5, 0.5, 0.095)$ and $K_\omega = diag(0.07, 0.07, 0.025)$.

In the second path, the initial conditions of the quadrotor are $\mathbf{p} = [0, 0, 0]^T$ metres and $Q = [-1, 0, 0, 0]^T$, and the desired path is a combination of two parts: the first part is

$$\begin{cases} x_d = 0 \quad ; \quad y_d = 0 \\ z_d = 3 - 2\cos(t\pi/20) \quad ; \quad q_{3d} = 0 \end{cases}$$

when $0 \le t < 10$ seconds, while the second part is

$$\begin{cases} x_d = 2\sin(t\pi/20) \quad ; \quad y_d = 0.1\tan(t\pi/20) \\ z_d = 5 - 2\cos(t\pi/20) \quad ; \quad q_{3d} = 0 \end{cases}$$

when $10 \le t \le 30$ seconds. The constant $\gamma$ is chosen to be $\gamma = 1.05$ and the weighting matrices are chosen to be $W_{1z} = 1025$, $W_2 = diag(0.0235, 0.0235, 0.0009)$, $W_{3z} = 25$ and $W_4 = diag(0.0043, 0.0043, 0.00156)$. Under these parameters the control matrices can be obtained by solving the conditions in (3.21), (3.22), (3.23) and (3.24) to be $k_{pz} = 105$, $k_{vz} = 50$, $K_q = diag(0.5, 0.5, 0.095)$ and $K_\omega = diag(0.07, 0.07, 0.025)$.

The testing results of tracking the first path using the proposed controller are obtained with the conditions (1) no disturbance, (2) force disturbance $d_{vz} = -2$Nm at $10 \leq t \geq 10.25$ seconds, $d_{vx} = 2$Nm at $20 \leq t \geq 20.25$ seconds, $d_{vy} = 2$Nm at $30 \leq t \geq 30.25$ seconds and the attitude part is disturbed using (3.27), (3) +30% model parameter uncertainty, and (4) −30% model parameter uncertainty. The tracking trajectories, positions, quaternions, and angular velocities are shown in Figures 3.4, 3.5, 3.6, and 3.7, respectively.

The testing results of tracking the second path using the proposed controller are obtained with the conditions (1) no disturbance, (2) force disturbance $d_{vz} = -2$Nm at $10 \leq t \geq 10.25$ seconds, $d_{vx} = 2$Nm at $15 \leq t \geq 15.25$ seconds, $d_{vy} = 2$Nm at $25 \leq t \geq 25.25$ seconds and the attitude part is disturbed using (3.27), (3) +30% model parameter uncertainty, and (4) −30% model parameter uncertainty. The tracking trajectories, positions, quaternions, and angular velocities are shown in Figures 3.8, 3.9, 3.10, and 3.11, respectively.

In Figures 3.4, 3.5, 3.8 and 3.9 the desired path (black) is tracked by the proposed controller (:red) and it is caught with less than 3 seconds. In addition, the controller under disturbance (–blue) can track the desired path and recovers from the disturbances within less than one second. The controller under +30% model parameter uncertainty (:magenta) and −30% model parameter uncertainty (–black) can track the desired path with very short time. The same result can be found in Figures 3.6, 3.7, 3.10 and 3.11 with even more obvious disturbance rejection.

It can be seen that the proposed controller is able to track the desired trajectories. The expected robustness is demonstrated by the disturbance rejection and the recovery from changes caused by the parameter uncertainties.

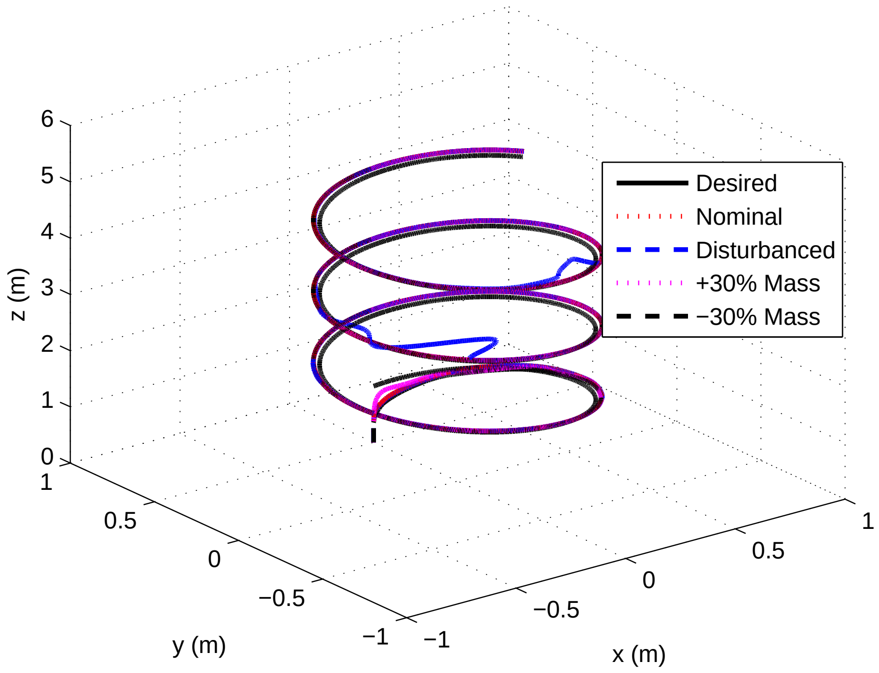

FIGURE 3.4: First Path Tracking under $H_\infty$ Controller Based on Quaternion Representation

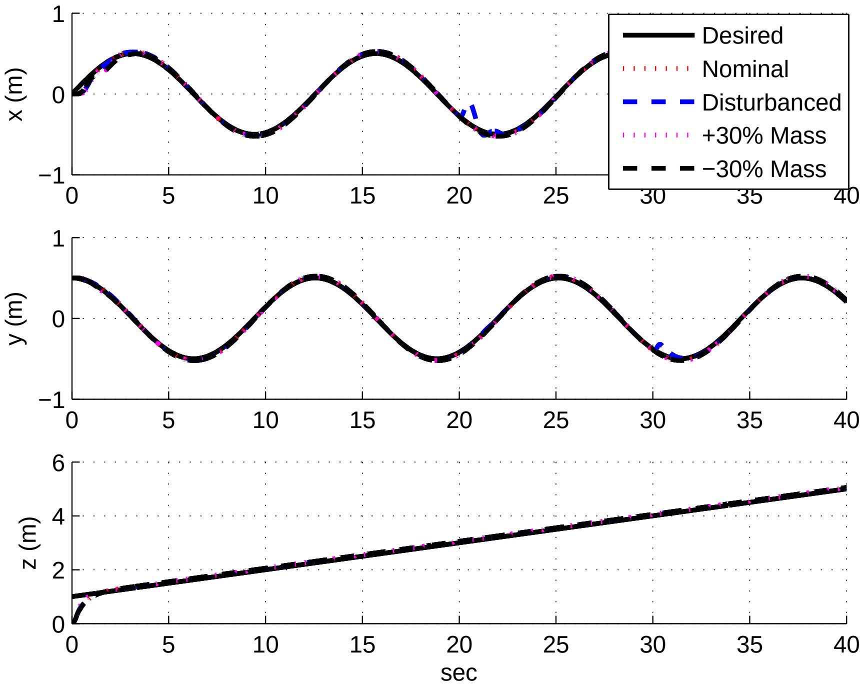

FIGURE 3.5: First Path Positions under $H_\infty$ Controller Based on Quaternion Representation

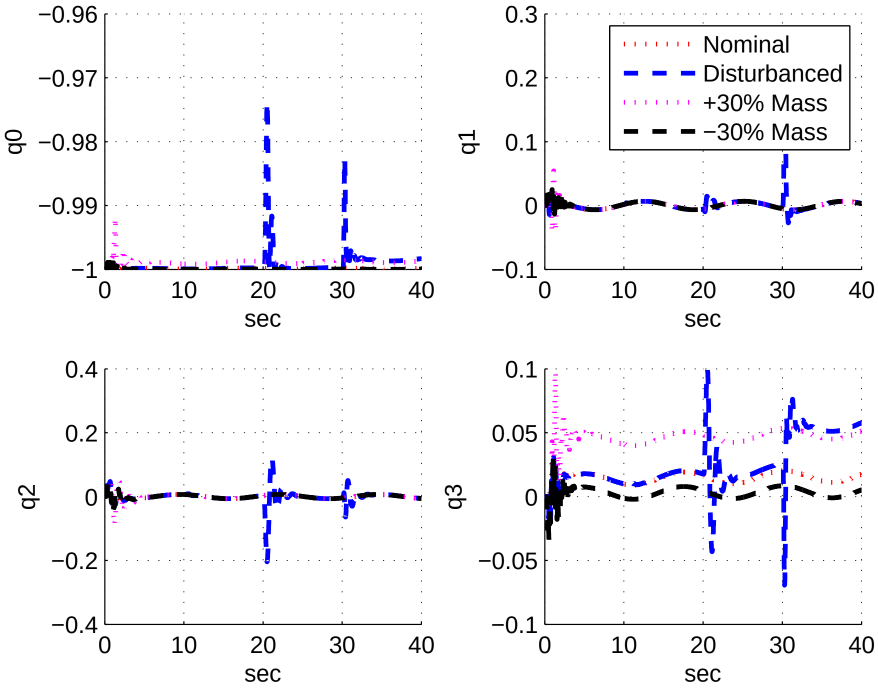

FIGURE 3.6: First Path Quaternion Components under $H_\infty$ Controller Based on Quaternion Representation

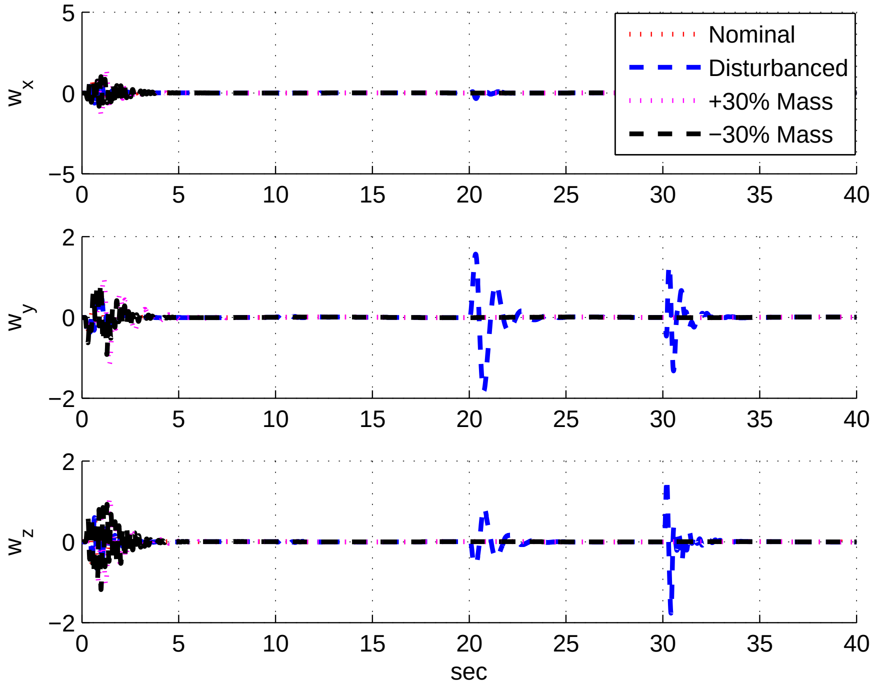

FIGURE 3.7: First Path Angular Velocities under $H_\infty$ Controller Based on Quaternion Representation

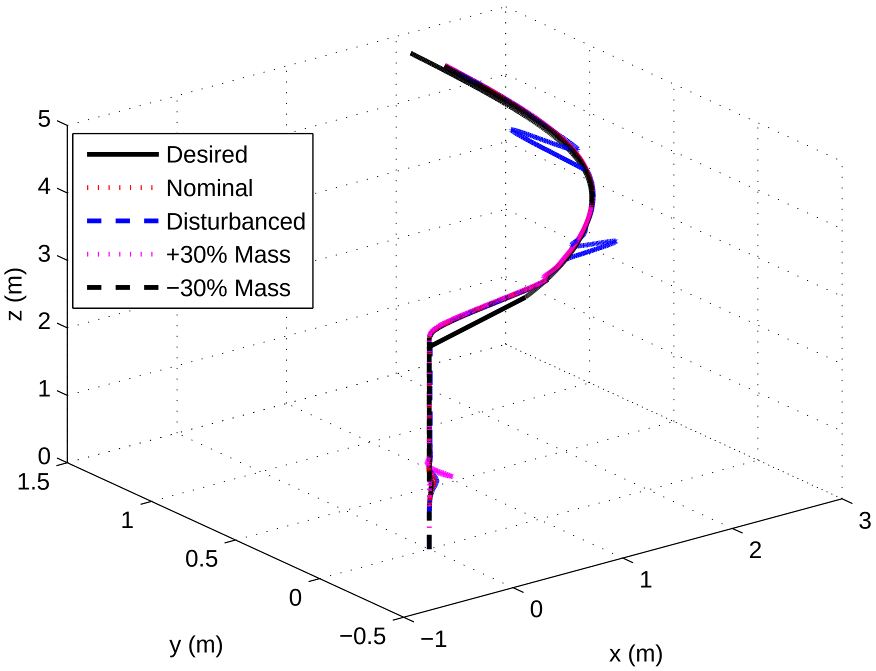

Figure 3.8: Second Path Tracking under $H_\infty$ Controller Based on Quaternion Representation

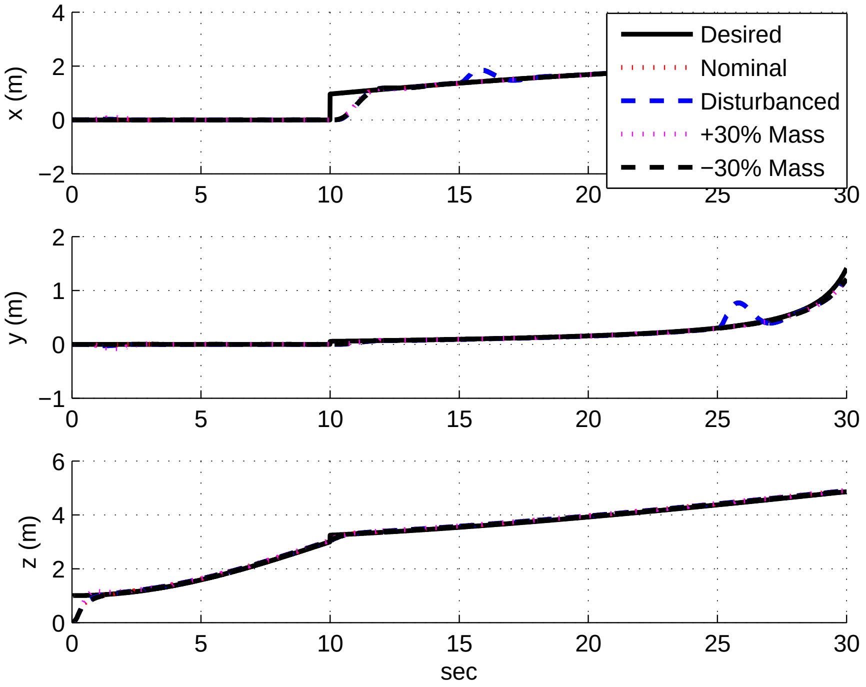

Figure 3.9: Second Path Positions under $H_\infty$ Controller Based on Quaternion Representation

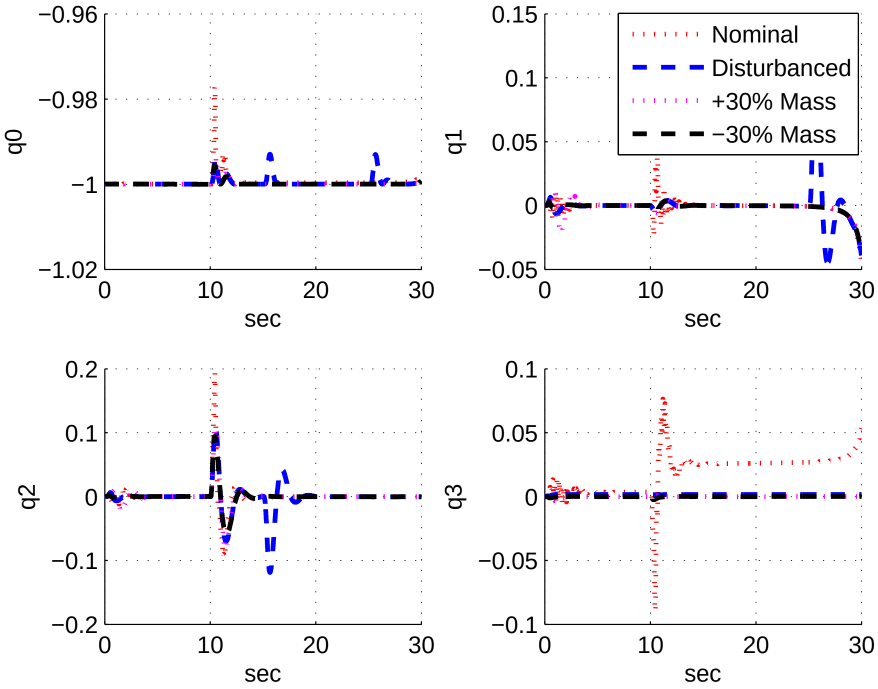

Figure 3.10: Second Path Quaternion Components under $H_\infty$ Controller Based on Quaternion Representation

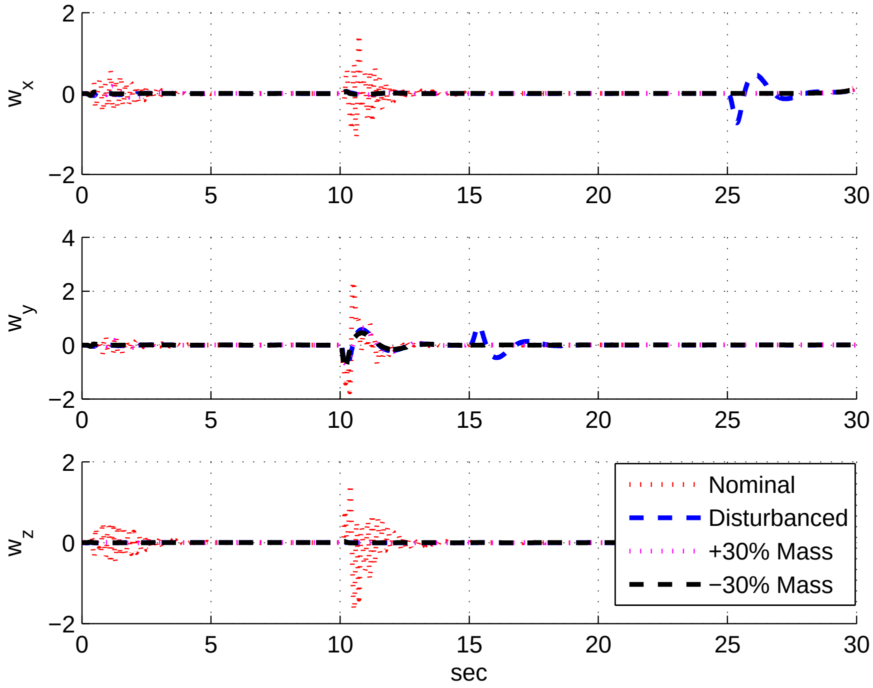

Figure 3.11: Second Path Angular Velocities under $H_\infty$ Controller Based on Quaternion Representation

| | Path 1 | | | | Path 2 | | | |
|---|---|---|---|---|---|---|---|---|
| **RMSE** | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ |
| $H_\infty$ | 0.019 | 0.0062 | 0.0088 | 0.00003 | 0.0266 | 0.0061 | 0.0013 | 0.0005 |
| $H_\infty + \mathbf{d}$ | 0.0221 | 0.0419 | 0.0255 | 0.0018 | 0.0132 | 0.0112 | 0.0384 | 0.0005 |
| $H_\infty$+30% | 0.019 | 0.0062 | 0.0088 | 0.00003 | 0.0324 | 0.0102 | 0.0136 | 0.0017 |
| $H_\infty$−30% | 0.019 | 0.0062 | 0.0088 | 0.00003 | 0.0319 | 0.0088 | 0.0062 | 0.0006 |

Table 3.3: Position and $q_3$ RMSE Values for the Two Paths under $H_\infty$ Controller Based on Quaternion Representation

| | Path 1 | | | | Path 2 | | | |
|---|---|---|---|---|---|---|---|---|
| **Max. Error** | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ |
| $H_\infty$ | 0.0258 | 0.0241 | 0.0530 | 0.0296 | 0.0416 | 0.0022 | 0.0405 | 0.0764 |
| $H_\infty + \mathbf{d}$ | 0.2705 | 0.1136 | 0.0575 | 0.0992 | 0.4354 | 0.4218 | 0.0408 | 0.0017 |
| $H_\infty$+30% | 0.0276 | 0.0240 | 0.0530 | 0.0973 | 0.0848 | 0.0081 | 0.1597 | 0.0007 |
| $H_\infty$−30% | 0.0256 | 0.0242 | 0.0530 | 0.0337 | 0.0447 | 0.0025 | 0.0283 | $2e^{-}5$ |

Table 3.4: Position and $q_3$ Maximum Error Values for the Two Paths under $H_\infty$ Controller Based on Quaternion Representation

## 3.4 Flight Control Based On Euler Angles Representation

In this section, we follow the procedure described earlier in Section 3.2 but with the use of Euler angles to represent the quadrotor dynamical model. External disturbances and model parameter uncertainties change are considered as well. A robust controller is derived and tested in simulation via the $H_\infty$ optimal design approach. The stability analysis is obtained via a selected Lyapunov function.

### 3.4.1 Attitude Stabilisation

The full quadrotor dynamic model is

$$\begin{cases} \dot{\mathbf{p}} = \mathbf{v} \\ \dot{\mathbf{v}} = -g\mathbf{e} + \frac{f}{m} R_\theta \mathbf{e} \\ \dot{R}_\theta = R_\theta S(\omega) \\ J\dot{\omega} = -S(\omega)J\omega - G(\omega) + \tau_E \end{cases} \tag{3.29}$$

and the rotational matrix $R_\theta$ from the inertial frame to the body frame is

$$R_\theta = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\varphi - s\psi c\varphi & c\psi s\theta c\varphi + s\psi s\varphi \\ s\psi c\theta & s\psi s\theta s\varphi + c\psi c\varphi & s\psi s\theta c\varphi - c\psi s\varphi \\ -s\theta & c\theta s\varphi & c\theta c\varphi \end{bmatrix}.$$

More details of the quadrotor model derivation based on Euler angles representation are described in Appendix A.

Assuming that $\varphi$, $\theta$, $\omega_x$, $\omega_y$ and $\omega_z$ are very small, $\zeta = [\varphi, \theta, \psi]^T$, $\eta = \dot{\zeta} = [\dot{\varphi}, \dot{\theta}, \dot{\psi}]^T = [\omega_x, \omega_y, \omega_z]^T$ and $\dot{\eta} = [\ddot{\varphi}, \ddot{\theta}, \ddot{\psi}]^T = [\dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z]^T$, then Equation (3.29) can be written for the attitude control purpose as:

$$\begin{cases} f = mg \\ \ddot{\varphi} = \dot{\theta}\dot{\psi}\frac{J_y - J_z}{J_x} + \frac{J_r}{J_x}\dot{\theta}\Omega + \frac{\tau_\varphi}{J_x} \\ \ddot{\theta} = \dot{\varphi}\dot{\psi}\frac{J_z - J_x}{J_y} - \frac{J_r}{J_y}\dot{\varphi}\Omega + \frac{\tau_\theta}{J_y} \\ \ddot{\psi} = \dot{\varphi}\dot{\theta}\frac{J_x - J_y}{J_z} + \frac{\tau_\psi}{J_z} \end{cases}. \tag{3.30}$$

The main goal is to asymptotically drive the quadrotor angles towards the desired angles $\zeta_d = [0, 0, 0]^T$ from initial angles with $\eta_d = [0, 0, 0]^T$ and the effect of added disturbances $\mathbf{d} = \mathbf{d}_\eta = [d_{\dot{\varphi}}, d_{\dot{\theta}}, d_{\dot{\psi}}]^T$ tending to disappear and changed parameters tending to be recovered. In mathematical terms, the aim is to satisfy the following conditions:

$$\begin{cases} \lim_{t\to\infty} \tilde{\zeta} = \lim_{t\to\infty}(\zeta_d - \zeta) = 0 \\ \lim_{t\to\infty} \tilde{\eta} = \lim_{t\to\infty}(\eta_d - \eta) = 0 \end{cases}. \tag{3.31}$$

Then the rotational part of Equation (3.29) can be written in the error state form as:

$$\begin{cases} \dot{\tilde{\zeta}} = \tilde{\eta} \\ J\dot{\tilde{\eta}} = S(\tilde{\eta})J\tilde{\eta} + G(\tilde{\eta}) - \tau_E \end{cases}. \tag{3.32}$$

Let $\mathbf{x} = [\tilde{\zeta}^T, \tilde{\eta}^T]^T$ and $\mathbf{u} = [G(\tilde{\eta}) - \tau_E]^T$. The nonlinear system (3.32) with the disturbance $\mathbf{d}$ can be written into an affine nonlinear form:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} + k(\mathbf{x})\mathbf{d} \tag{3.33}$$

where

$$f(\mathbf{x}) = \begin{bmatrix} \tilde{\eta} \\ J^{-1}S(\tilde{\eta})J\tilde{\eta} \end{bmatrix}$$

$$g(\mathbf{x}) = k(\mathbf{x}) = \begin{bmatrix} 0_{3\times 3} \\ J^{-1} \end{bmatrix}.$$

The $H_\infty$ suboptimal control approach described earlier in this chapter is used in this section as well. By selecting the following Lyapunov function,

$$V(\mathbf{x}) = \frac{1}{2}\begin{bmatrix} \tilde{\zeta}^T & \tilde{\eta}^T \end{bmatrix}\begin{bmatrix} C_\zeta I & JK_\zeta \\ JK_\zeta & JK_\eta \end{bmatrix}\begin{bmatrix} \tilde{\zeta} \\ \tilde{\eta} \end{bmatrix} \tag{3.34}$$

where diagonal matrices $K_\zeta > 0$ and $K_\eta > 0$ are the proportional and derivative gains for translational and rotational parts. $C_\zeta > 0$ is constant, and its derivative

$$\frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} = [C_\zeta\tilde{\zeta} + JK_\zeta\tilde{\eta} \quad JK_\zeta\tilde{\zeta} + JK_\eta\tilde{\eta}],$$

then the controller is

$$\begin{aligned} \mathbf{u} &= -g^T(\mathbf{x})\frac{\partial^T V(\mathbf{x})}{\partial \mathbf{x}} \\ &= -\begin{bmatrix} K_\zeta\tilde{\zeta} + K_\eta\tilde{\eta} \end{bmatrix}. \end{aligned} \tag{3.35}$$

The following diagonal weighting matrices are chosen $W_2 > 0$ and $W_4 > 0$;

$$h(\mathbf{x}) = \begin{bmatrix} \sqrt{W_2}\tilde{\zeta}^T & \sqrt{W_4}\tilde{\eta}^T \end{bmatrix}^T,$$

which satisfies $h(\mathbf{x}_*) = 0$, where the equilibrium point $\mathbf{x}_* = [0_{1\times3}, 0_{1\times3}]^T$. Then the following conditions are obtained:

$$\begin{aligned}
C_\zeta K_\eta &\geq J K_\zeta^2 \\
C_\zeta &= K_\zeta K_\eta \left(1 - \frac{1}{\gamma^2}\right) \\
\|K_\zeta\|^2 &\geq \frac{\gamma^2 \|W_2\|}{\gamma^2 - 1} \qquad (3.36) \\
\|K_\eta\|^2 &\geq \frac{\gamma^2 (\|W_4\| + 2\|J\|\|K_\zeta\|)}{\gamma^2 - 1} \qquad (3.37) \\
\|W_2\| &> 0; \; \|W_4\| > 0
\end{aligned}$$

using the norms $\|\tilde{\eta}^T J K_\eta S(\tilde{\eta})\tilde{\eta}\| = 0$, $|\tilde{\eta}^T J K_\zeta \tilde{\eta}| \leq \|J\|\|K_\zeta\|\|\tilde{\eta}\|^2$ and $|\tilde{\zeta}^T J K_\eta S(\tilde{\eta})\tilde{\eta}| = 0$.

Then the total torque applied to the quadrotor is obtained as below:

$$\begin{aligned}
\mathbf{u} &= -\left[\; K_\zeta \tilde{\zeta} + K_\eta \tilde{\eta} \;\right] \\
&= G(\tilde{\eta}) - \tau_E.
\end{aligned}$$

Then

$$\tau_E = K_\zeta \tilde{\zeta} + K_\eta \tilde{\eta} + G(\tilde{\eta}). \qquad (3.38)$$

### 3.4.2 Path Tracking Control

The full mathematical model of the quadrotor (3.29) is used to control the quadrotor to track 3D paths. In this section, the procedure described in Section 3.2.3 is followed. The tracking errors can be written as:

$$\begin{aligned}\tilde{\mathbf{p}} &= \mathbf{p}_d - \mathbf{p}\\ \tilde{\mathbf{v}} &= \mathbf{v}_d - \mathbf{v}\\ \tilde{\eta} &= \eta_d - \eta\\ \tilde{\zeta} &= \zeta_d - \zeta\end{aligned}$$

and Equation (3.29) can be rewritten in an error form as:

$$\begin{cases}\dot{\tilde{\mathbf{p}}} = \tilde{\mathbf{v}}\\ \dot{\tilde{\mathbf{v}}} = g\mathbf{e} - \frac{f}{m}R_\theta\mathbf{e}\\ \dot{\tilde{\zeta}} = \tilde{\eta}\\ J\dot{\tilde{\eta}} = S(\tilde{\eta})J\tilde{\eta} + G(\tilde{\eta}) - \tau_E\end{cases}. \tag{3.39}$$

The main goal is to asymptotically drive the quadrotor towards the desired position $\mathbf{p}_d$ from an initial position with the effect of added disturbances tending to disappear and changed parameters tending to be recovered by satisfying the following conditions:

$$\begin{cases}\lim_{t\to\infty}\tilde{\mathbf{p}} = \lim_{t\to\infty}(\mathbf{p}_d - \mathbf{p}) = 0\\ \lim_{t\to\infty}\tilde{\psi} = \lim_{t\to\infty}(\psi_d - \psi) = 0\end{cases}. \tag{3.40}$$

Now we consider the robust control approach to the path tracking problem when considering $\mathbf{d} = [\mathbf{d}_{\mathbf{v}}^T, \mathbf{d}_\eta^T]^T = [d_{vx}, d_{vy}, d_{vz}, d_{\dot{\varphi}}, d_{\dot{\theta}}, d_{\dot{\psi}}]^T$ as the disturbance applied to the nonlinear system (3.39). Those disturbances are used here to model the changes of mass and moment, and the wind disturbances.

Let

$$\mathbf{x} = \begin{bmatrix} \tilde{\mathbf{p}} \\ \tilde{\zeta} \\ \tilde{\mathbf{v}} \\ \tilde{\eta} \end{bmatrix}$$
$$\mathbf{u} = \begin{bmatrix} g\mathbf{e} - \frac{f}{m} R_\theta \mathbf{e} \\ G(\tilde{\eta}) - \tau_E \end{bmatrix}.$$

The dynamic system (3.29) with the disturbance **d** can be written into an affine nonlinear form:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} + k(\mathbf{x})\mathbf{d} \tag{3.41}$$

where

$$f(\mathbf{x}) = \begin{bmatrix} \tilde{\mathbf{v}} \\ \tilde{\eta} \\ 0_{3\times1} \\ J^{-1}S(\tilde{\eta})J\tilde{\eta} \end{bmatrix}$$

$$g(\mathbf{x}) = k(\mathbf{x}) = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \\ I & 0_{3\times3} \\ 0_{3\times3} & J^{-1} \end{bmatrix}.$$

By choosing the following $V$ function

$$V(\mathbf{x}) = \frac{1}{2}\left[\begin{array}{cccc} \tilde{\mathbf{p}}^T & \tilde{\zeta}^T & \tilde{\mathbf{v}}^T & \tilde{\eta}^T \end{array}\right] \begin{bmatrix} C_p I & 0_{3\times3} & K_p & 0_{3\times3} \\ 0_{3\times3} & C_\zeta I & 0_{3\times3} & JK_\zeta \\ K_p & 0_{3\times3} & K_v & 0_{3\times3} \\ 0_{3\times3} & JK_\zeta & 0_{3\times3} & JK_\eta \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}} \\ \tilde{\zeta} \\ \tilde{\mathbf{v}} \\ \tilde{\eta} \end{bmatrix} \tag{3.42}$$

where diagonal matrices $K_p > 0, K_\zeta > 0, K_v > 0, K_\eta > 0$ are the proportional and derivative gains for translational and rotational parts. $C_p > 0, C_\zeta > 0$ are constants. And

$$\frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} = [\ C_p\tilde{\mathbf{p}} + K_p\tilde{\mathbf{v}} \quad C_\zeta\tilde{\zeta} + JK_\zeta\tilde{\eta} \quad K_p\tilde{\mathbf{p}} + K_v\tilde{\mathbf{v}} \quad JK_\zeta\tilde{\zeta} + JK_\eta\tilde{\eta}\ ],$$

the controller is

$$\begin{aligned} \mathbf{u} &= -g^T(\mathbf{x})\frac{\partial^T V(\mathbf{x})}{\partial \mathbf{x}} \\ &= -\begin{bmatrix} K_p\tilde{\mathbf{p}} + K_v\tilde{\mathbf{v}} \\ K_\zeta\tilde{\zeta} + K_\eta\tilde{\eta} \end{bmatrix}. \end{aligned} \tag{3.43}$$

The following diagonal weighting matrices are chosen $W_1 > 0$, $W_2 > 0$, $W_3 > 0$ and $W_4 > 0$;

$$h(\mathbf{x}) = [\sqrt{W_1}\tilde{\mathbf{p}}^T \quad \sqrt{W_2}\tilde{\zeta}^T \quad \sqrt{W_3}\tilde{\mathbf{v}}^T \quad \sqrt{W_4}\tilde{\eta}^T\ ]^T,$$

which satisfies $h(\mathbf{x}_*) = 0$, where the equilibrium point $\mathbf{x}_* = [0_{1\times3}, 0_{1\times3}, 0_{1\times3}, 0_{1\times3}]^T$. And we know

$$V(\mathbf{x}_*) = 0.$$

Now the path tracking problem of the quadrotor under the disturbance $\mathbf{d}$ is defined below.

*Problem* 4. Given the equilibrium point $\mathbf{x}_*$, find the parameters $K_p, K_\zeta, K_v, K_\eta, C_p$, $C_\zeta$ in order to enable the closed-loop system (3.41) with the above controller $\mathbf{u}$ (3.43) to have $L_2$-gain less than or equal to $\gamma$.

Next, our main result is represented in the following theorem.

**Theorem 3.4.** *If the following conditions are satisfied, the closed-loop system* (3.41) *with the above controller* $\mathbf{u}$ (3.43) *has* $L_2$*-gain less than or equal to* $\gamma$*. And the closed loop system* (3.41)*,* (3.43) *with* $\mathbf{d} = 0$ *is asymptotically locally stable for the equilibrium point* $\mathbf{x}_*$*.*

$$\begin{aligned}
C_p C_\zeta &\geq 0 \\
C_p K_v &\geq K_p^2 \\
C_p C_\zeta K_v K_\eta &\geq C_p J K_\zeta^2 K_v - J K_\zeta^2 K_p^2 + C_\zeta K_p^2 K_\eta \\
C_p &= K_p K_v \left(1 - \frac{1}{\gamma^2}\right) \\
C_\zeta &= K_\zeta K_\eta \left(1 - \frac{1}{\gamma^2}\right) \\
\|K_p\|^2 &\geq \frac{\gamma^2 \|W_1\|}{\gamma^2 - 1} \qquad (3.44) \\
\|K_\zeta\|^2 &\geq \frac{\gamma^2 \|W_2\|}{\gamma^2 - 1} \qquad (3.45) \\
\|K_v\|^2 &\geq \frac{\gamma^2 (\|W_3\| + 2\|K_p\|)}{\gamma^2 - 1} \qquad (3.46) \\
\|K_\eta\|^2 &\geq \frac{\gamma^2 (\|W_4\| + 2\|J\| \|K_\zeta\|)}{\gamma^2 - 1} \qquad (3.47) \\
\|W_1\| &> 0; \|W_2\| > 0; \|W_3\| > 0; \|W_4\| > 0.
\end{aligned}$$

*Proof.* With the given conditions, we need to show (1) $V(\mathbf{x}) \geq 0$ and (2) the Hamiltonian $H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq 0$. Then the first part of the theorem can be proved by using Theorem 3.1.

(1) From Equation (3.43) the conditions for $V(\mathbf{x}) \geq 0$ are

$$\begin{aligned} C_p C_\zeta &\geq 0 \\ C_p K_v &\geq K_p^2 \\ C_p C_\zeta K_v K_\eta &\geq C_p J K_\zeta^2 K_v - J K_\zeta^2 K_p^2 + C_\zeta K_p^2 K_\eta \end{aligned}$$

(2)

$$\begin{aligned} H_\gamma(\mathbf{x}, V(\mathbf{x})) &= \tilde{\mathbf{p}}^T C_p \tilde{\mathbf{v}} + \tilde{\mathbf{v}}^T K_p \tilde{\mathbf{v}} + \tilde{\zeta}^T C_\zeta \tilde{\eta} + \tilde{\eta}^T J K_\zeta \tilde{\eta} + \tilde{\zeta}^T J K_\zeta S(\tilde{\eta}) \tilde{\eta} \\ &+ \tilde{\eta}^T J K_\eta S(\tilde{\eta}) \tilde{\eta} + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right) \|K_p \tilde{\mathbf{p}} + K_v \tilde{\mathbf{v}}\|^2 \\ &+ \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right) \|K_\zeta \tilde{\zeta} + K_\eta \tilde{\eta}\|^2 + \frac{1}{2}\|W_1\|\|\tilde{\mathbf{p}}\|^2 + \frac{1}{2}\|W_2\|\|\tilde{\zeta}\|^2 \\ &+ \frac{1}{2}\|W_3\|\|\tilde{\mathbf{v}}\|^2 + \frac{1}{2}\|W_4\|\|\tilde{\eta}\|^2. \end{aligned}$$

By choosing

$$\begin{aligned} C_p &= K_p K_v \left(1 - \frac{1}{\gamma^2}\right) \\ C_\zeta &= K_\zeta K_\eta \left(1 - \frac{1}{\gamma^2}\right), \end{aligned}$$

then

$$\begin{aligned} H_\gamma(\mathbf{x}, V(\mathbf{x})) &= \tilde{\mathbf{v}}^T K_p \tilde{\mathbf{v}} + \tilde{\eta}^T J K_\zeta \tilde{\eta} + \tilde{\zeta}^T J K_\zeta S(\tilde{\eta}) \tilde{\eta} + \tilde{\eta}^T J K_\eta S(\tilde{\eta}) \tilde{\eta} \\ &+ \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right) \left(\|K_p\|^2 \|\tilde{\mathbf{p}}\|^2 + \|K_v\|^2 \|\tilde{\mathbf{v}}\|^2\right) \\ &+ \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right) \left(\|K_\zeta\|^2 \|\tilde{\zeta}\|^2 + \|K_\eta\|^2 \|\tilde{\eta}\|^2\right) + \frac{1}{2}\|W_1\|\|\tilde{\mathbf{p}}\|^2 \\ &+ \frac{1}{2}\|W_2\|\|\tilde{\zeta}\|^2 + \frac{1}{2}\|W_3\|\|\tilde{\mathbf{v}}\|^2 + \frac{1}{2}\|W_4\|\|\tilde{\eta}\|^2. \end{aligned}$$

By using $|\tilde{\mathbf{v}}^T K_p \tilde{\mathbf{v}}| \leq \|K_p\|\|\tilde{\mathbf{v}}\|^2$, $\|S(\tilde{\eta})\| = \|\tilde{\eta}\|$, $\|\tilde{\eta}^T J K_\eta S(\tilde{\eta}) \tilde{\eta}\| = 0$, $|\tilde{\eta}^T J K_\zeta \tilde{\eta}| \leq \|J\|\|K_\zeta\|\|\tilde{\eta}\|^2$ and $|\tilde{\zeta}^T J K_\eta S(\tilde{\eta}) \tilde{\eta}| = 0$, we have

$$
\begin{aligned}
H_\gamma(\mathbf{x}, V(\mathbf{x})) = \|K_p\|\|\tilde{\mathbf{v}}\|^2 + \|J\|\|K_\zeta\|\|\tilde{\eta}\|^2 + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)(\|K_p\|^2\|\tilde{\mathbf{p}}\|^2 \\
+ \|K_v\|^2\|\tilde{\mathbf{v}}\|^2) + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\left(\|K_\zeta\|^2\|\tilde{\zeta}\|^2 + \|K_\eta\|^2\|\tilde{\eta}\|^2\right) \\
+ \frac{1}{2}\|W_1\|\|\tilde{\mathbf{p}}\|^2 + \frac{1}{2}\|W_2\|\|\tilde{\zeta}\|^2 + \frac{1}{2}\|W_3\|\|\tilde{\mathbf{v}}\|^2 + \frac{1}{2}\|W_4\|\|\tilde{\eta}\|^2.
\end{aligned}
$$

Thus, the conditions for $H_\gamma(\mathbf{x}, V(\mathbf{x})) \leq 0$ are

$$
\begin{aligned}
\frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_p\|^2 + \frac{1}{2}\|W_1\| &\leq 0 \\
\frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_\zeta\|^2 + \frac{1}{2}\|W_2\| &\leq 0 \\
\|K_p\| + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_v\|^2 + \frac{1}{2}\|W_3\| &\leq 0 \\
\|J\|\|K_\zeta\| + \frac{1}{2}\left(\frac{1}{\gamma^2} - 1\right)\|K_\eta\|^2 + \frac{1}{2}\|W_4\| &\leq 0;
\end{aligned}
$$

i.e.

$$
\begin{aligned}
\|K_p\|^2 &\geq \frac{\gamma^2\|W_1\|}{\gamma^2 - 1} \\
\|K_\zeta\|^2 &\geq \frac{\gamma^2\|W_2\|}{\gamma^2 - 1} \\
\|K_v\|^2 &\geq \frac{\gamma^2(\|W_3\| + 2\|K_p\|)}{\gamma^2 - 1} \\
\|K_\eta\|^2 &\geq \frac{\gamma^2(\|W_4\| + 2\|J\|\|K_\zeta\|)}{\gamma^2 - 1}.
\end{aligned}
$$

It is trivial to show that the nonlinear system (3.41) is zero-state observable for the equilibrium point $\mathbf{x}_*$. Further, due to the fact that $V(\mathbf{x}) \geq 0$ and it is a proper function (i.e. for each $\beta > 0$ the set $\{x : 0 \leq V(x) \leq \beta\}$ is compact), the closed-loop system (3.41), (3.43) with $\mathbf{d} = 0$ is asymptotically locally stable for the equilibrium point $\mathbf{x}_*$ according to Proposition 1. This proves the second part of the theorem. □

*Remark* 2. It should be noted that the proof of Theorem 3.4, $\lim_{t\to\infty}\tilde{\mathbf{p}} = 0$, $\lim_{t\to\infty}\tilde{\zeta} = 0$, $\lim_{t\to\infty}\tilde{\mathbf{v}} = 0$ and $\lim_{t\to\infty}\tilde{\eta} = 0$ meets the conditions of (3.40).

Finally from $\mathbf{u}$, we can have

$$\begin{aligned}\mathbf{u} &= \begin{bmatrix} g\mathbf{e} - \frac{f}{m}R_\theta\mathbf{e} \\ G(\tilde{\eta}) - \tau_E \end{bmatrix} \\ &= -\begin{bmatrix} K_p\tilde{\mathbf{p}} + K_v\tilde{\mathbf{v}} \\ K_\zeta\tilde{\zeta} + K_\eta\tilde{\eta} \end{bmatrix}.\end{aligned}$$

Then the total force and the torque vector applied to the quadrotor, $f$ and $\tau_E \in \mathbb{R}^3$, are

$$f = (k_{pz}\tilde{z} + k_{vz}\tilde{v}_z + g)\frac{m}{c\varphi c\theta} \tag{3.48}$$

$$\tau_E = K_\zeta\tilde{\zeta} + K_\eta\tilde{\eta} + G(\tilde{\eta}). \tag{3.49}$$

## 3.5 Simulations

The proposed $H_\infty$ controller was tested in a MATLAB quadrotor simulator. To test the robustness of the proposed controller, the model parameter uncertainties (mass and inertia) were increased and decreased by ±30% and a force disturbance of 2N was added in different operation times to the positions for 0.25 seconds duration, while the disturbances added to the attitude were of the form described in (3.27) and (3.28). The quadrotor parameters used in the simulation are described in Table 3.1.

### 3.5.1 Attitude Stability

Attitude control simulation testing included monitoring the response to the external disturbances (3.27) and (3.28) and ±30% model parameter uncertainties

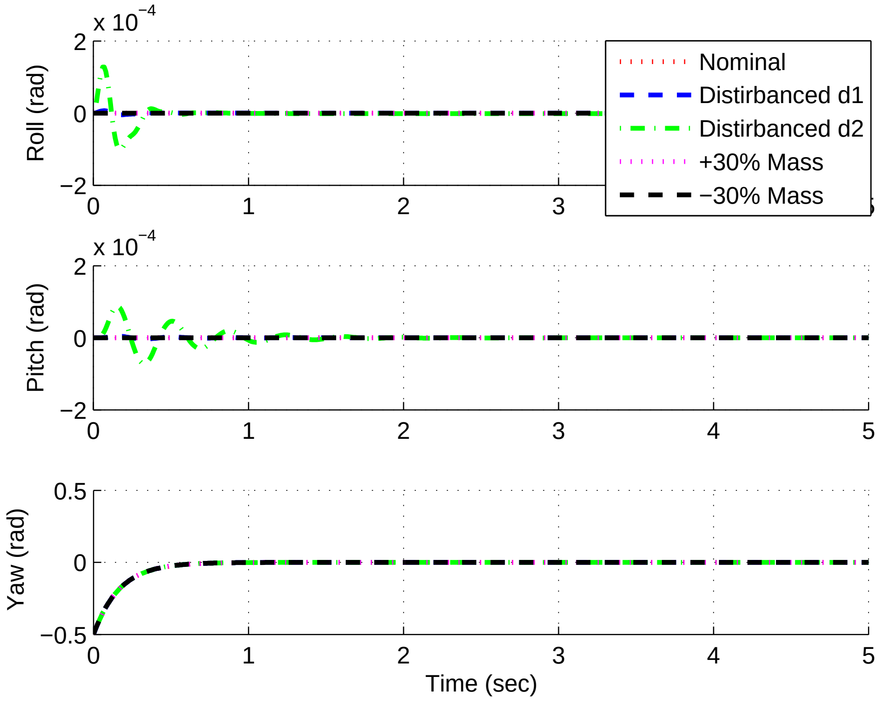

FIGURE 3.12: Euler Angles under $H_\infty$ Controller

with non-zero initial conditions. Running the simulator using the attitude controller (3.38) with no external disturbances and model parameter uncertainties (Nominal), with external disturbance in (3.27) (Disturbanced **d1**), with external disturbance in (3.28) (Disturbanced **d2**), +30% model parameter uncertainties (+30% Mass) and −30% model parameter uncertainties (−30% Mass) shows that the external disturbance disrupted the attitude performance slightly and it was rejected after one second (Figure 3.12). The thrust and torques applied to the system were illustrated in Figure 3.13. Additionally, the controller was able to recover the model parameter uncertainties over the running time interval.

As a result of solving the conditions (3.36) and (3.37) using $\gamma = 1.05$, $W_2 = diag(10, 235, 235)$ and $W_4 = diag(0.01, 5.7, 5.7)$, the controller parameters were $K_\zeta = diag(10, 50, 50)$ and $K_\eta = diag(0.1, 8.09, 8.4)$.

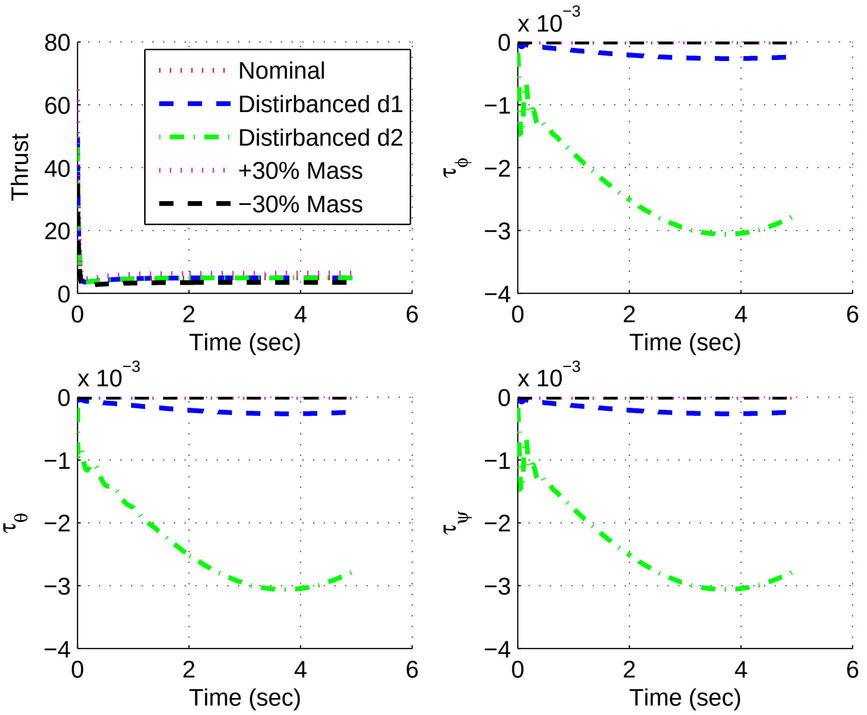

FIGURE 3.13: Thrust and Torques Input under $H_\infty$ Controller

### 3.5.2 Path Tracking

This section presents the path tracking simulation results done with $H_\infty$ control laws (3.48) and (3.49). Two different paths were used to validate the proposed controller in simulation. The first desired path used was defined by

$$\begin{cases} x_d = 2\cos(t\pi/80) \quad ; \quad y_d = 2\sin(t\pi/80) \\ z_d = 1 + 0.1t \quad ; \quad \psi_d = \pi/6 \end{cases}.$$

The quadrotor initial positions were $[x, y, z]^T = [2, 0, 0]^T$ metres and the initial angles were $[\varphi, \theta, \psi]^T = [0, 0, 0]^T$ radian. The second desired path used was defined by

$$\begin{cases} x_d = 0.5\sin(0.5t) \quad ; \quad y_d = 0.5\cos(0.5t) \\ z_d = 1 + 0.1t \quad ; \quad \psi_d = 0 \end{cases}.$$

The quadrotor initial positions were $[x, y, z] = [0, 0.5, 0]$ metres and the initial angles were $[\varphi, \theta, \psi]^T = [0, 0, 0]^T$ radian. The constant $\gamma$ was chosen to be $\gamma = 1.05$ and the norms of the weighting matrices were chosen to be $W_{1z} = 760$,

$W_2 = diag(10, 235, 235)$, $W_{3z} = 50$ and $W_4 = diag(0.01, 5.7, 5.7)$. Under these parameters the feedback control matrices were obtained to be $k_{pz} = 90$, $k_{vz} = 50$, $K_\zeta = diag(10, 50, 50)$ and $K_\eta = diag(0.1, 8.09, 8.4)$.

The performance of tracking the first path is illustrated in Figures 3.14 - 3.16, while Figures 3.17 - 3.19 display the tracking performance of the second path using the proposed $H_\infty$ controller. These figures have the conditions (1) no disturbance, (2) force disturbance $d_{vz} = -2$Nm during $10 \leq t \leq 10.25$ seconds, $d_{vx} = 2$Nm during $20 \leq t \leq 20.25$ seconds, $d_{vy} = 2$Nm during $30 \leq t \leq 30.25$ seconds and the attitude disturbed using (3.27) during the flight interval, (3) $+30\%$ model parameter uncertainty, and (4) $-30\%$ model parameter uncertainty.

As can be seen in Figures 3.14 - 3.19, simulations of the proposed $H_\infty$ controller showed good performance in terms of position tracking errors. The desired path was tracked by the proposed controller and it was caught with less than 3 seconds. In addition, the controller under disturbance was able to track the desired path and recover from the disturbances within less than 2 seconds. The controller under $+30\%$ model parameter uncertainty and $-30\%$ model parameter uncertainty was able to track the desired path as the nominal path.

Table 3.5 demonstrates the positions and yaw angle RMSE values of the two paths. It is clear that the RMSE values of the proposed $H_\infty$ controller were almost the same when using the $H_\infty$ controller in normal conditions and with $\pm30\%$ model parameter uncertainty in both paths, while they slightly increased with the use of disturbances. In general, it can be seen that the proposed controller was able to track the desired trajectories. The expected robustness was demonstrated by the disturbance rejection and the recovery from changes caused by the parameter uncertainties.

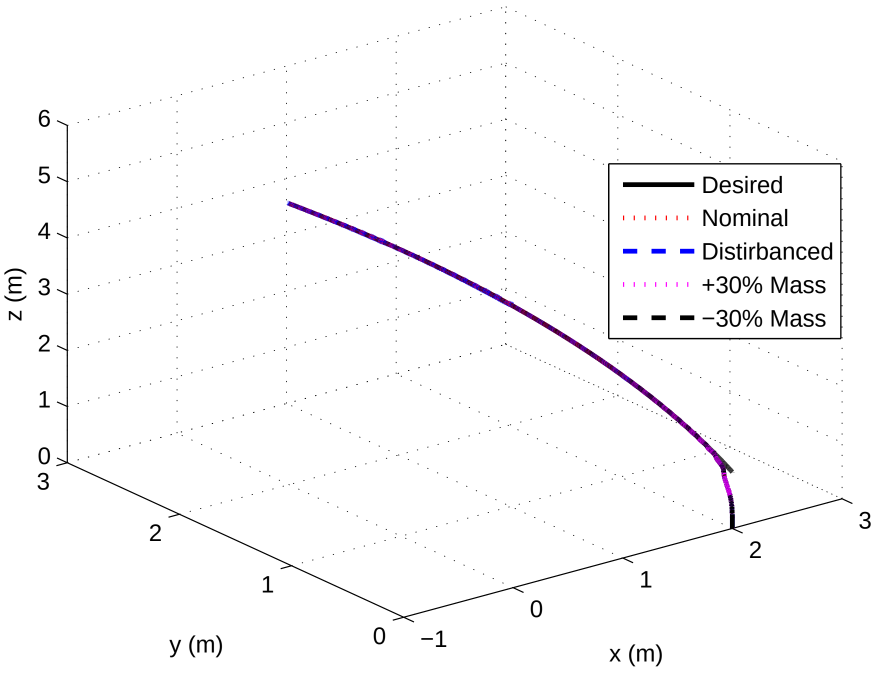

FIGURE 3.14: First Path tracking under $H_\infty$ Controller Based on Euler Angles Representation

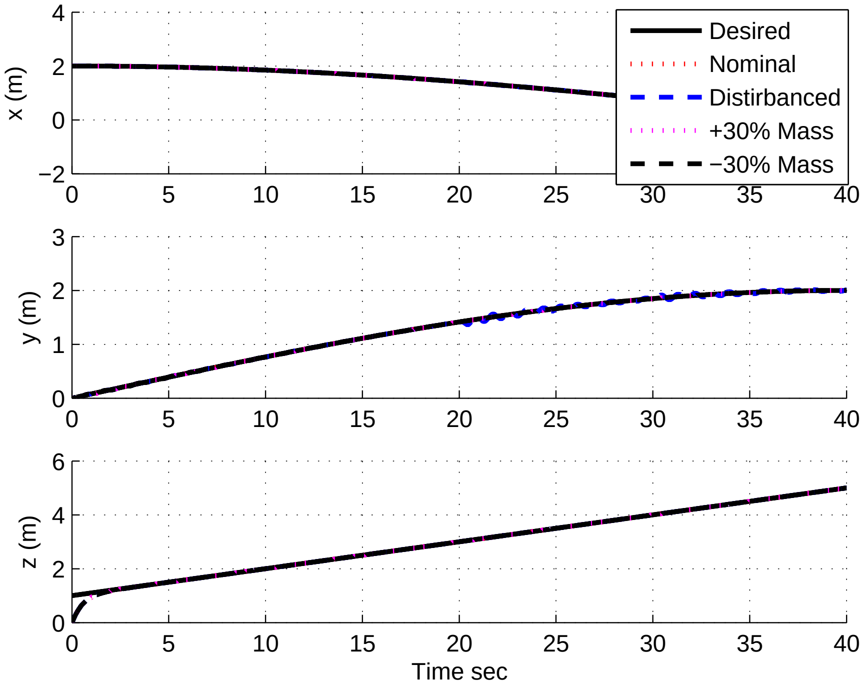

FIGURE 3.15: First Path Position under $H_\infty$ Controller Based on Euler Angles Representation

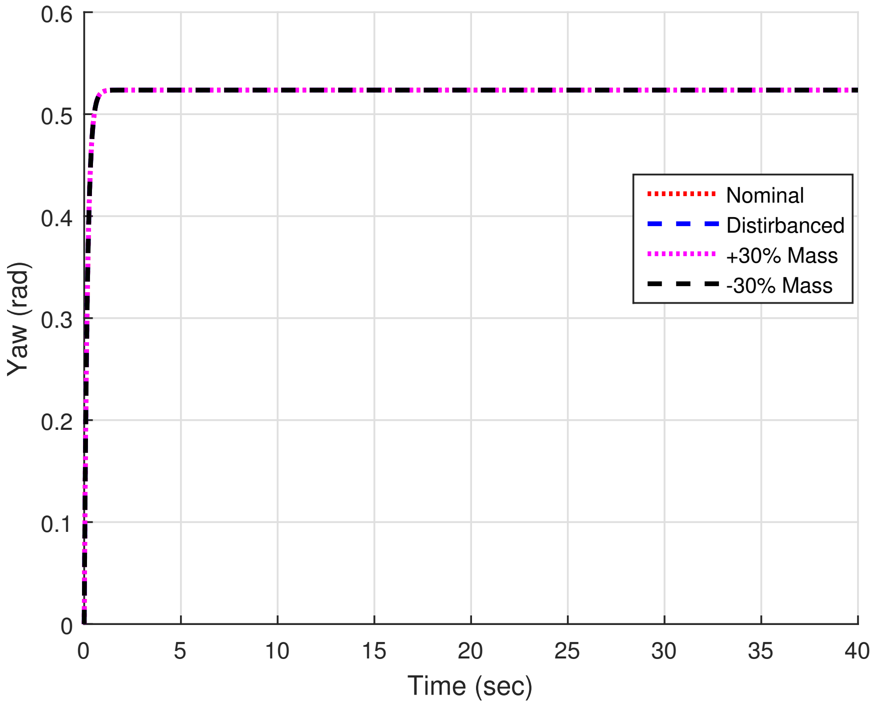

FIGURE 3.16: First Path Yaw Angle under $H_\infty$ Controller Based on Euler Angles Representation

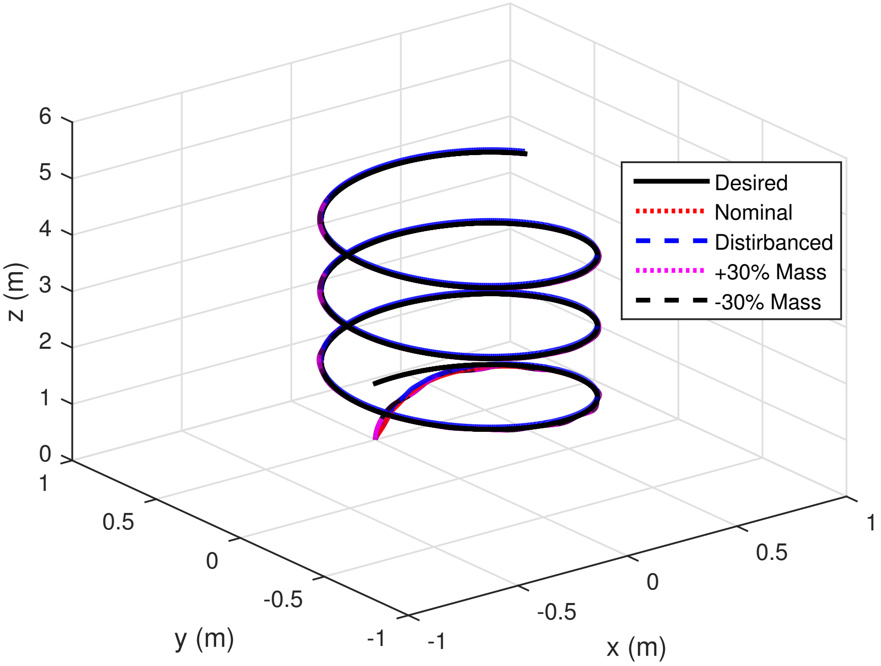

FIGURE 3.17: Second Path tracking under $H_\infty$ Controller Based on Euler Angles Representation

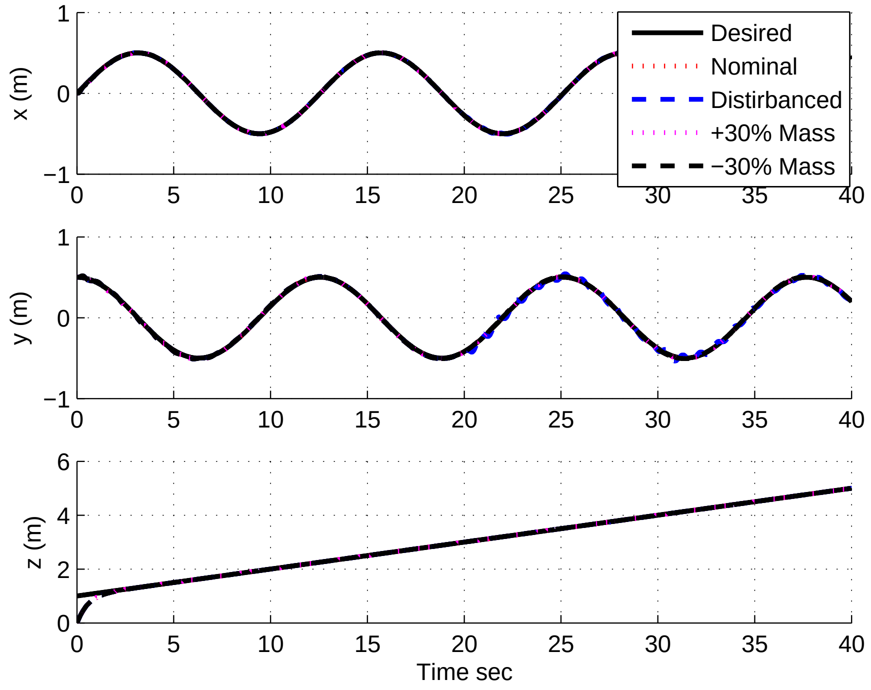

FIGURE 3.18: Second Path Position under $H_\infty$ Controller Based on Euler Angles Representation

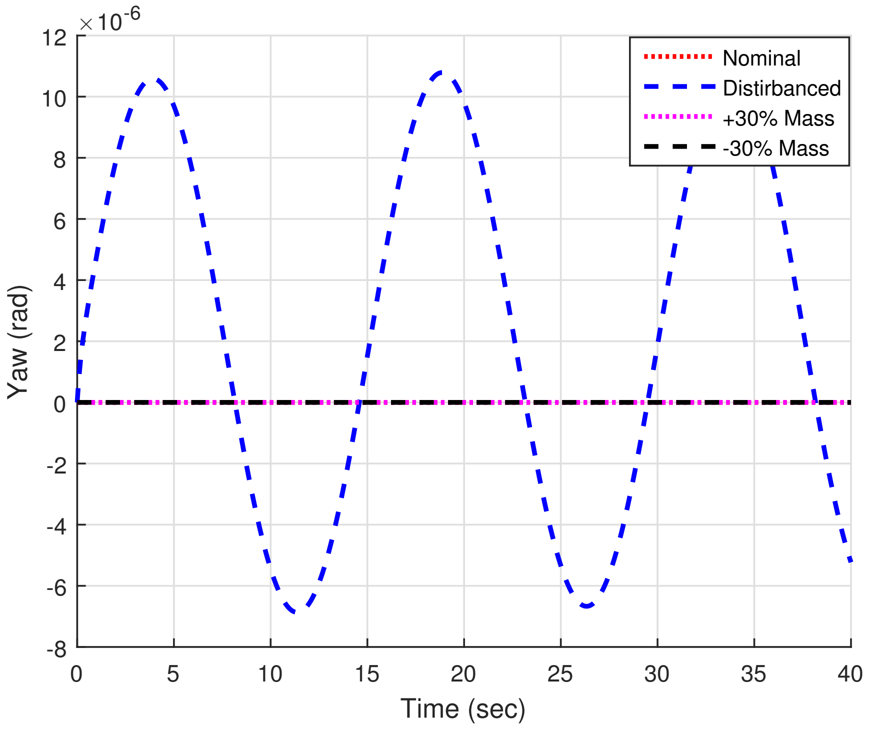

FIGURE 3.19: Second Path Yaw Angle under $H_\infty$ Controller Based on Euler Angles Representation

| | Path 1 | | | | Path 2 | | | |
|---|---|---|---|---|---|---|---|---|
| **RMSE** | $x(m)$ | $y(m)$ | $z(m)$ | $\psi(deg.)$ | $x(m)$ | $y(m)$ | $z(m)$ | $\psi(deg.)$ |
| $H_\infty$ | 0.0004 | 0.0028 | 0.0857 | 0.0004 | 0.0004 | 0.0004 | 0.0857 | 0 |
| $H_\infty + \mathbf{d}$ | 0.0030 | 0.0106 | 0.0857 | 0.0004 | 0.0071 | 0.0108 | 0.0868 | $6e^{-}6$ |
| $H_\infty$+30% | 0.0004 | 0.0028 | 0.0857 | 0.0004 | 0.0020 | 0.0081 | 0.0857 | 0 |
| $H_\infty$−30% | 0.0004 | 0.0028 | 0.0857 | 0.0004 | 0.0020 | 0.0081 | 0.0857 | 0 |

Table 3.5: Position and $\psi$ RMSE Values for the Two Paths under $H_\infty$ Controller Based on Euler Angles Representation

## 3.6 Experiments

### 3.6.1 Quadrotor Platform Setup

The experimental platform available at the robot arena laboratory at the university of Essex and used in this thesis is a Hummingbird quadrotor designed by Ascending Technology shown in Figure 1.1. It is mounted by two micro-controllers with a (60MHz) CPU clock, running at (1KHz) and executing several tasks simultaneously. It also has two standard XBee-PRO wireless serial link modules to send and receive the packet data from the computer to the UAV and vice versa through 12 direct sequence channels and its actual transmission rate is up to 100 Hz sending IMU and gyro data. Moreover, the vehicle has useful indoor and outdoor application sensors with small neglected errors, such as IMU, whereas the others have significant errors in indoor applications, and are useful only outdoors, such as GPS and Barometer [120][121].

The AscTec. Hummingbird quadrotor UAV has four motors. Each one creates a vertical force $f_i$ which depends on its angular speed according to Equation (3.50),

$$f_i = K_f \Omega_i^2 \tag{3.50}$$

where $K_f \approx 6.11 \times 10^{-}10 \frac{N}{rpm^2}$. The moment is also produced according to Equation (3.51).

$$M_i = K_M \Omega_i^2 \tag{3.51}$$

where $K_M \approx 1.5 \times 10^-9 \frac{N_m}{rpm^2}$. The motor mathematical model is a first order differential equation that illustrates the rotor speed depending on the desired speed.

$$\dot{\Omega}_i = K_m(\Omega^{des_i} - \Omega_i) \tag{3.52}$$

where $K_m = 20s^-1$ [122].

The information used to control the quadrotor was received from an IMU, mounted on the quadrotor itself, and from a Vicon Motion Capture System which was fixed in the robot arena laboratory. The speed and the angular velocity of the quadrotor were received from the IMU sensor while the estimated position of the quadrotor was received from the Vicon system. Figure 3.20 shows the control diagram used in this work; it can be noticed that the communication between the computer and the quadrotor was achieved via two Xbees: the first one was mounted on the quadrotor while the second one was connected to the computer. The position of the quadrotor was obtained and calculated by the Vicon system at 50 Hz and received by the computer, which also ran at 50 Hz, to be used in the controller calculations. Then the controller's action was sent via the Xbee to the high level on-board microcontroller in the vehicle, which directly controls the motors' speed. Two of this type of quadrotor will be used in Chapter 4 as leader and follower in real tests.

### 3.6.2 Quadrotor Flight Performance

In order to validate the proposed $H_\infty$ controller and assess its robustness to the external disturbances and weight changes, the controller was implemented on the vehicle and tested to follow three different paths in indoor flights with a consideration of the external disturbances and weight changes. The quadrotor in path tracking flight is shown in Figure 3.28. The three trajectories that were followed by the vehicle were as follows.

(1) It took off at the origin point and hovered at 1.5 metres, then it moved 1 metre towards the x-direction. After that, it drew a square of 2 metres side length and then landed.

(2) It took off and hovered at 1.5 metres above the origin point. Then it moved 1 meter towards the x-direction, drew a circle of a 2 metres diameter and landed.

(3) It took off and hovered at 0.5 metres, flew around a vertical spiral of 1.5 metres height and a 2 metres circle diameter, moved 1 meter towards the origin point and then landed.

The results are illustrated in Figures 3.21 - 3.27 with (:red) is the normal performance, (:magenta) is the performance of +20% mass, and (–blue) is the performance with disturbances (the external disturbance input is introduced by a wooden stick).

Figure 3.21 shows the take-off for 1.5 metres, hovering for 60 seconds and landing with disturbances in different times and directions. Figure 3.22 illustrates the performance of the vehicle tracking the first trajectory in three dimensions, and Figure 3.23 shows its top view. Figures 3.24 and 3.25 demonstrate the performance of the quadrotor following the second path in a three dimensions and a two dimensions $X - Y$ horizontal plane, respectively. The results of tracking the third path in three dimension and its positions are illustrated in Figures 3.26 and 3.27 respectively. The disturbances of the trajectories in these figures were caused by the external force exerted on the vehicle. As depicted in these figures, the controller achieved good tracking performance with a very small overshoot and a steady state error. The results also show the very good robustness in the presence of weight changes and external disturbances. The controller reveals the ability to recover the effect of external disturbance and the weight changes quickly and smoothly. It should be noted that the proposed controllers were able to track different trajectories, reject the external disturbances and cover the model parameter uncertainties practically with very small errors.

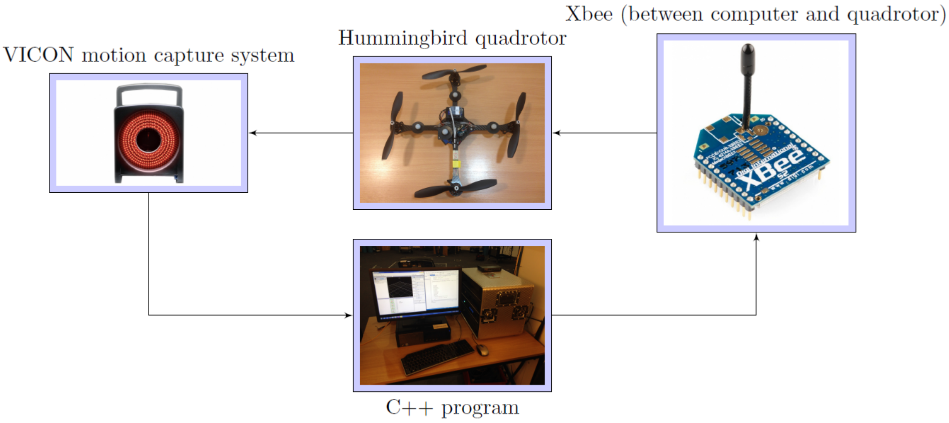
Xbee (between computer and quadrotor)
Hummingbird quadrotor
VICON motion capture system
C++ program

Figure 3.20: Experimental Control Block Diagram

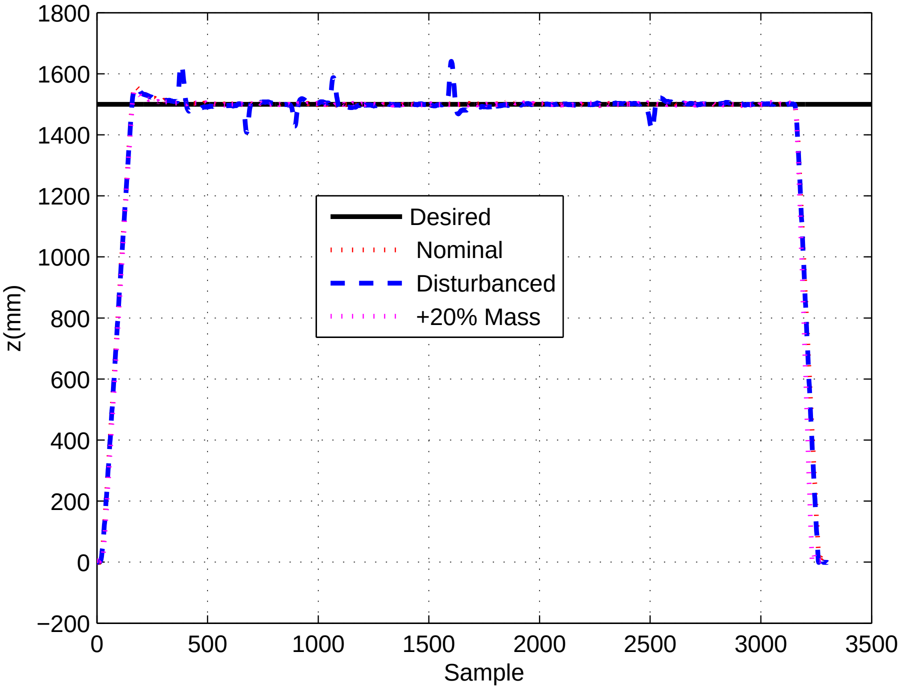

Figure 3.21: Takeoff, Hovering and Landing

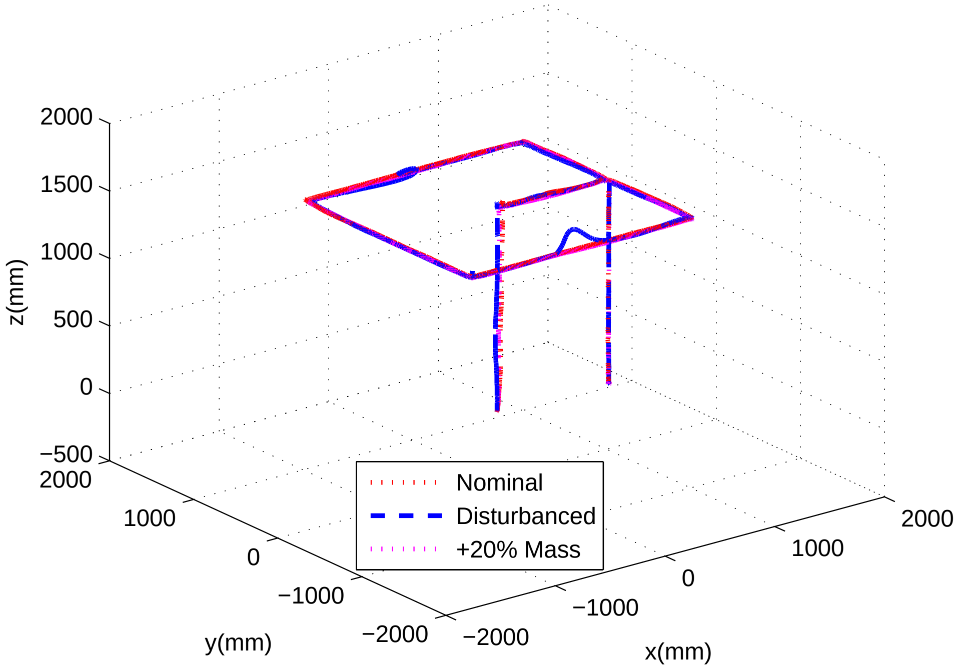

Figure 3.22: Takeoff, Square and Landing in 3D

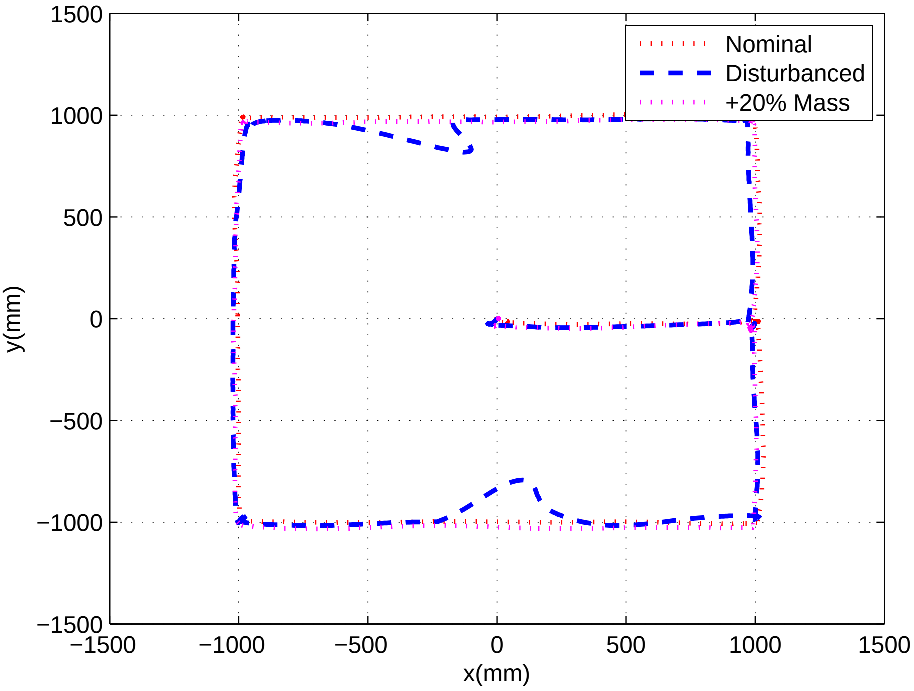

Figure 3.23: Takeoff, Square and Landing, Top View

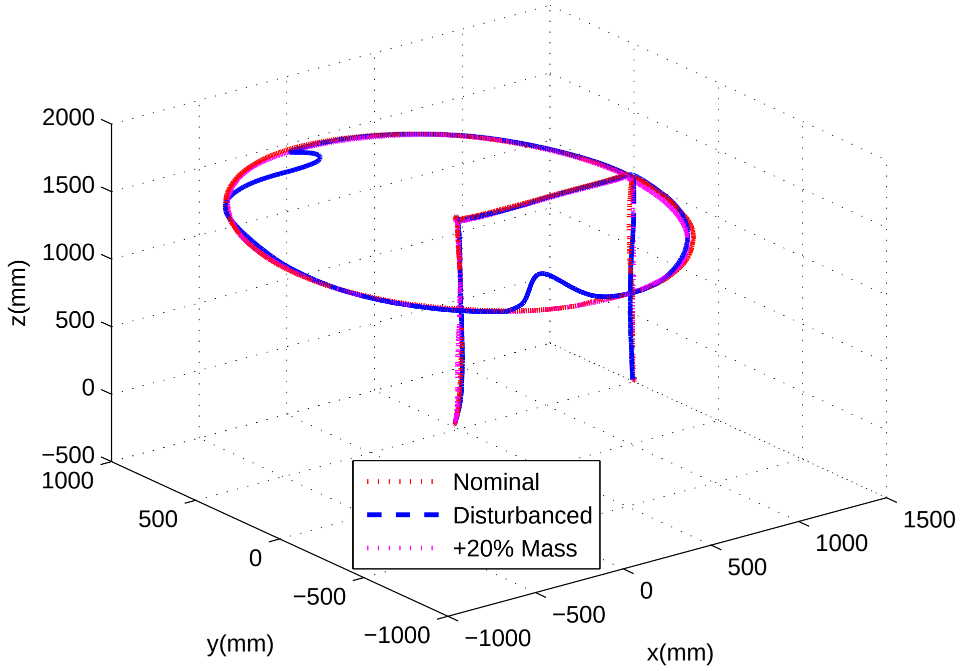

Figure 3.24: Takeoff, Circle and Landing in 3D

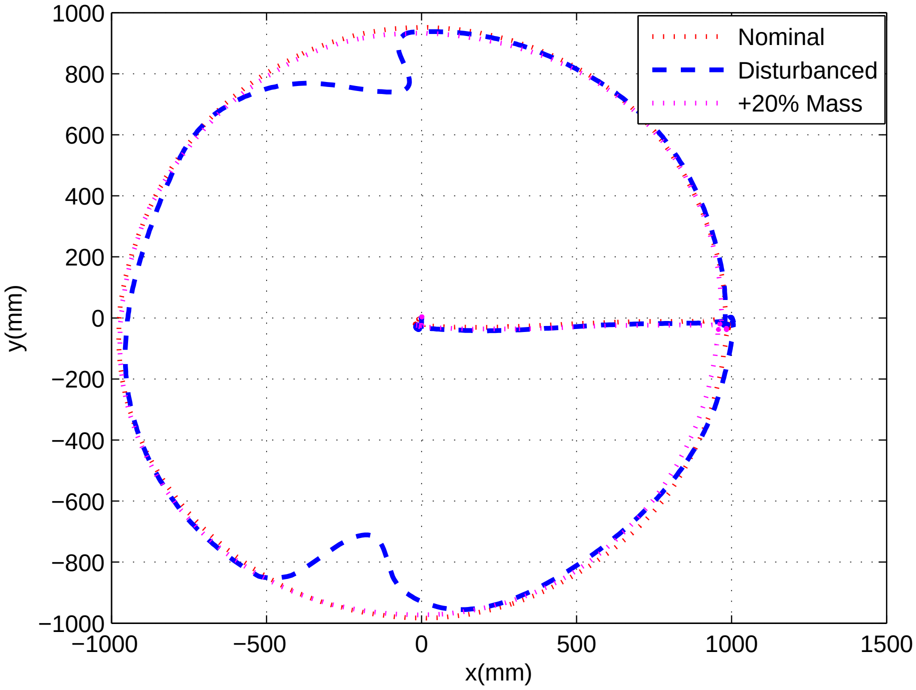

Figure 3.25: Takeoff, Circle and Landing, Top View

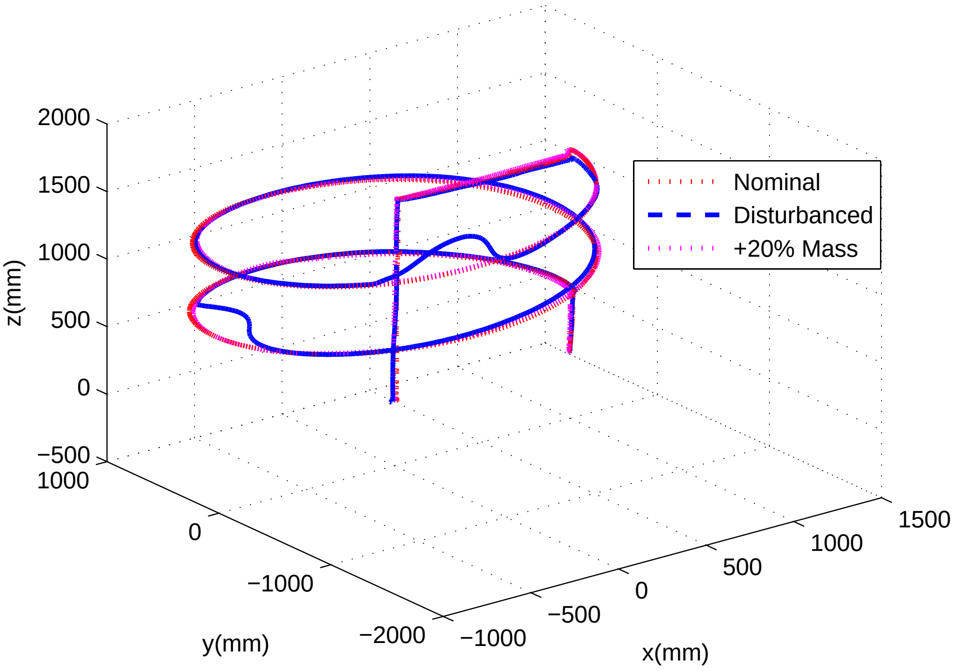

Figure 3.26: Takeoff, Spiral and Landing in 3D

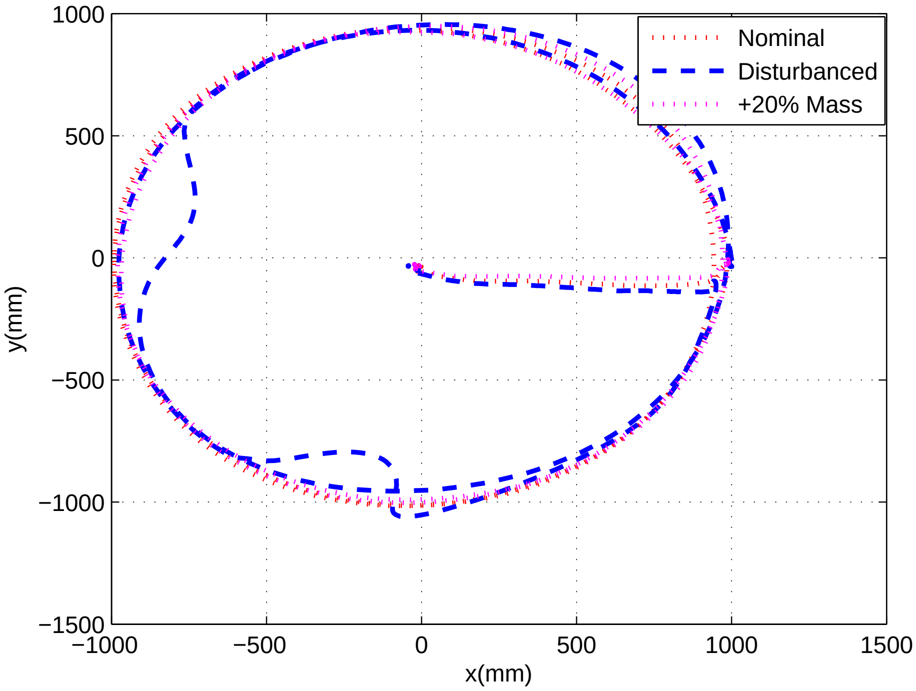

Figure 3.27: Takeoff, Spiral and Landing, Top View

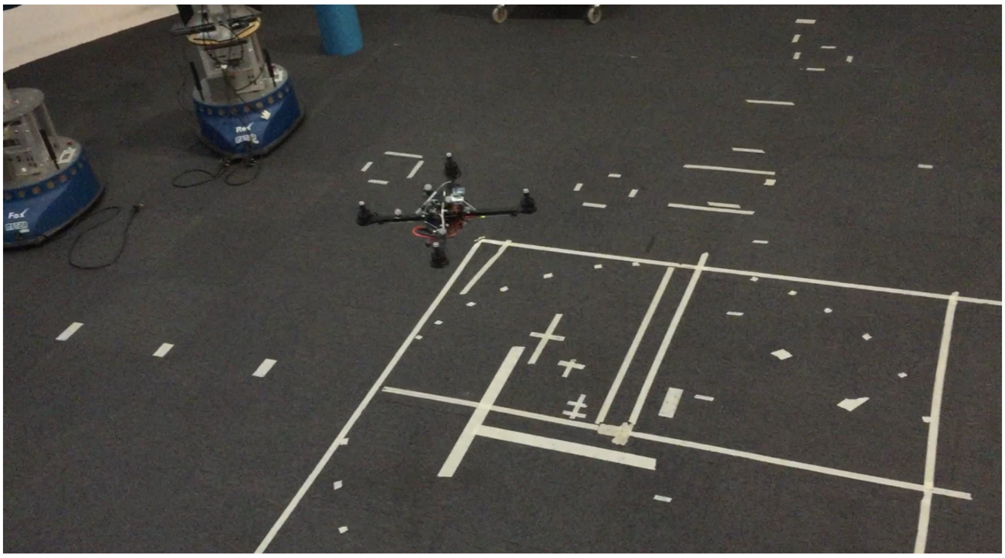

FIGURE 3.28: Individual Quadrotor in Path Tracking Flight

## 3.7 Discussions

This chapter has illustrated the theory and derivation procedure of the $H_\infty$ to find a suboptimal controller for the rotational and translational parts of a quadrotor to perform autonomous flight with the presence of external disturbances and model parameter uncertainties effects. The control strategy had a feedback loop to control the quadrotor. Our controllers were derived for the quadrotor dynamic system represented based on quaternions and Euler angles with a nonlinear dynamic effect. The stability issue of the proposed $H_\infty$ suboptimal controller was examined and its conditions were obtained via choosing suitable Lyapunov functions. The overall controllers were successfully tested in simulation through several paths. Experientially the position controller was efficiently tested on an AscTec. Hummingbird quadrotor to track various paths. In simulation and experiment flight, the proposed controllers showed the ability to cover the model parameter uncertainties change and quick external disturbances rejection. The quadrotor controlled in this chapter is reused as a leader in leader-follower team formation control in next chapter. The $H_\infty$ theory is also reused in the next chapter for team formation control design to find the follower controllers.

# Chapter 4

# Team Formation Control

## 4.1 Introduction

The multi-robot formation control problem has received a lot of attention in the last decades. In this chapter, the $H_\infty$ control technique described earlier in Chapter 3 was extended to the leader-follower formation control problem of quadrotor UAVs, with no change to the process of designing a state feedback controller. The leader-follower formation control problem to be solved in this chapter was a distributed control scheme; i.e. the leader and the follower had their own individual controllers without the need for a centralised unit. Therefore, the controllers derived in Chapter 3 for individual quadrotor path tracking were used as a leader in formation control. The follower controllers were designed to track the leader with the presence of external disturbances, once when the quadrotors were represented based on quaternion and the other when they were represented based on Euler angles. The stability and robustness conditions of the follower controllers were presented by selecting a suitable Lyapunov function.

Two identical quadrotors were used as leader-follower in simulation and practical tests. Two different predefined paths were presented to prove the robustness of the $H_\infty$ control technique in simulation and several paths were also presented in the practical phase.

## 4.2 The Leader-Follower Formation Problem Based on Quaternion Representation

To describe the orientation of a quadrotor, the quaternion representation is used. The full dynamic model (3.7) of a quadrotor can be written as:

$$\begin{cases} \dot{\mathbf{p}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = -g\mathbf{e} + \frac{f_i}{m_i} R_{iq}\mathbf{e} \\ \begin{bmatrix} \dot{q}_{i0} \\ \dot{\mathbf{q}}_i \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\mathbf{q}_i^T \omega_i \\ (q_{i0} I + S(\mathbf{q}_i))\, \omega_i \end{bmatrix} \\ J_i \dot{\omega}_i = -S(\omega_i) J_i \omega_i - G(\omega_i) + \tau_{iq} \end{cases} . \tag{4.1}$$

### 4.2.1 The Leader-Follower Formation Control Problem

One leader and one follower are considered in the leader-follower formation control problem to be solved in this chapter. The leader control problem is formulated as a trajectory tracking, and the follower control problem is also formulated as a tracking problem, but with a different tracking target.

The follower keeps its yaw angle $(q_{F0}, q_{F3})$ the same as the leader when it maintains the formation pattern. It moves to a desired position $\mathbf{p}_{Fd}$, which is determined by a desired distance $d$, a desired incidence angle $\rho$, and a desired bearing angle $\sigma$. A new frame $F'$ is defined by the translation of the leader frame $L$ to the frame with the desired follower position $\mathbf{p}_{Fd}$ as the origin. As shown in Figure 4.1, the desired incidence angle is measured between the desired distance $d$ and the $x - y$ plane in the new frame $F'$, and the desired bearing angle is measured between the $x$ axis and the projection of the $d$ in $x - y$ plane in the new frame $F'$. The desired position $\mathbf{p}_{Fd}$ is

$$\mathbf{p}_{Fd} = \mathbf{p}_L - R_{Lq}^T d \begin{bmatrix} \cos\rho\cos\sigma \\ \cos\rho\sin\sigma \\ \sin\rho \end{bmatrix} .$$

Now, the formation control problem for the follower is to satisfy the following conditions:

$$\begin{cases} \lim_{t\to\infty}(\mathbf{p}_{Fd}-\mathbf{p}_F)=0 \\ \lim_{t\to\infty}(q_{L0}-q_{F0})=0 \\ \lim_{t\to\infty}(q_{L3}-q_{F3})=0 \end{cases}. \tag{4.2}$$

The leader just tracks a desired trajectory represented by $(\mathbf{p}_{Ld}, q_{L0d}, q_{L3d})$. So, the formation control problem for the leader is to satisfy the following conditions:

$$\begin{cases} \lim_{t\to\infty}(\mathbf{p}_{Ld}-\mathbf{p}_L)=0 \\ \lim_{t\to\infty}(q_{L0d}-q_{L0})=0 \\ \lim_{t\to\infty}(q_{L3d}-q_{L3})=0 \end{cases}. \tag{4.3}$$

In summary, the leader-follower formation control problem to be solved in this chapter is a distributed control scheme. Assume both the leader and the follower are able to obtain their own pose information and the follower is able to obtain the leader's pose information via wireless communication. The design goal of the controllers is to find the state feedback control law for the thrust and torque inputs for both the leader and the follower. The leader-follower formation control problem is solved if both conditions (4.2) and (4.3) are satisfied.

The communication among the robots is assumed to be available. The position $\mathbf{p}_L$, quaternion components $q_{L0}$ and $q_{L3}$ of the leader $L$ and their first and second derivatives $\dot{q}_{L0}$, $\ddot{q}_{L0}$, $\dot{q}_{L3}$ and $\ddot{q}_{L3}$ are assumed available and measurable. The linear velocity of the leader $L$ and its derivative $\mathbf{v}_L$ and $\dot{\mathbf{v}}_L$ are assumed bounded and available for the follower.

## 4.3 Formation $H_\infty$ Controllers

The controller design for the leader and the follower is based on $H_\infty$ suboptimal control. The follower $H_\infty$ controller is designed by following the introduction of

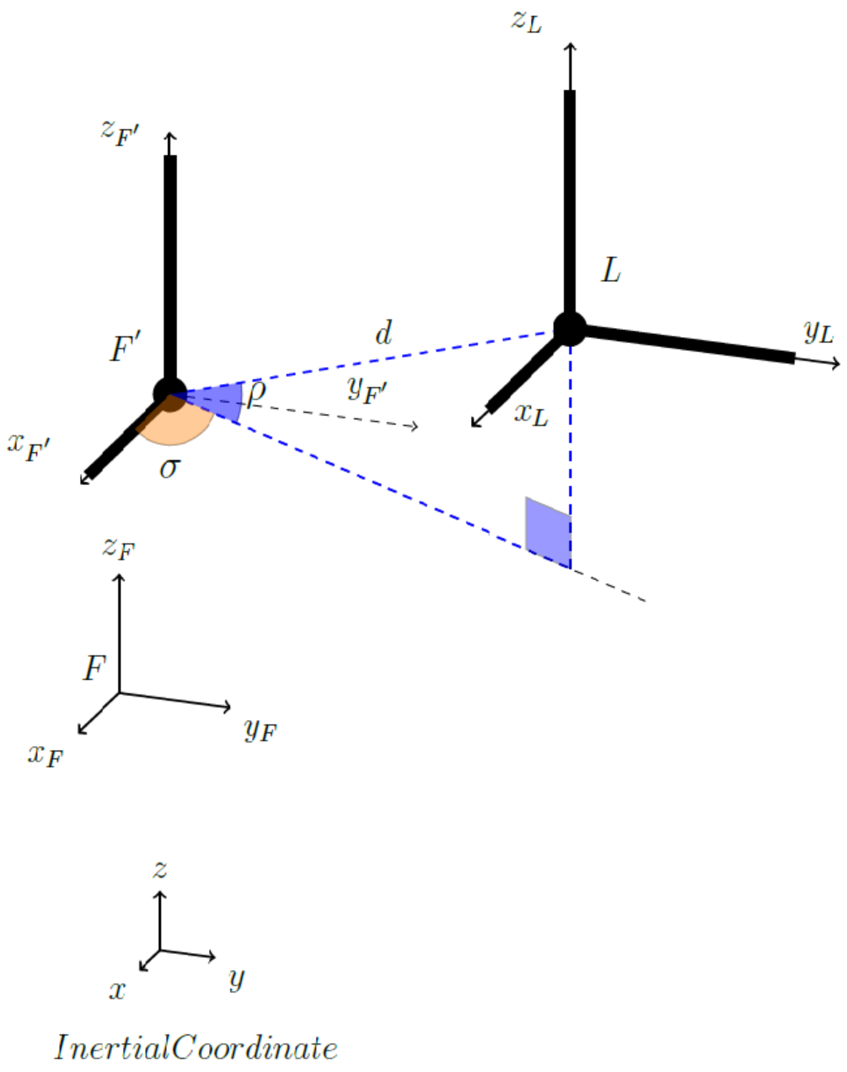

Figure 4.1: Body Frames in Formation

an error state model, and the introduction of a $H_\infty$ control theorem for general affine systems. The leader $H_\infty$ controller was presented in the previous chapter.

### 4.3.1 Follower State Error Model

The control strategy for the follower is to track the desired position $\mathbf{p}_{Fd}$. The tracking errors for the follower according to the nonlinear dynamic system (4.1) can be written as:

$$\begin{aligned}
\tilde{\mathbf{p}}_F &= \mathbf{p}_{Fd} - \mathbf{p}_F \\
\tilde{\mathbf{v}}_F &= \mathbf{v}_{Fd} - \mathbf{v}_F \\
\begin{bmatrix} \tilde{q}_{F0} \\ \tilde{\mathbf{q}}_F \end{bmatrix} &= \begin{bmatrix} q_{F0d} - q_{F0} \\ \mathbf{q}_{Fd} - \mathbf{q}_F \end{bmatrix} \\
\tilde{\omega}_F &= \omega_{Fd} - \omega_F
\end{aligned}$$

where $\mathbf{v}_{Fd} = \dot{\mathbf{p}}_{Fd}$ is the desired linear velocity, $[q_{F0d}, \mathbf{q}_{Fd}]^T = [q_{L0}, 0, 0, q_{L3}]^T$ is the desired quaternion, and $[\omega_{Fd}] = [0, 0, 0]^T$ is the desired angular velocity.

### 4.3.2 Follower $H_\infty$ Controller

The control strategy for the follower is to track the leader trajectory. The tracking errors for the follower according to the nonlinear dynamic system (4.1) can be written as:

$$\begin{cases}
\dot{\tilde{\mathbf{p}}}_F = \tilde{\mathbf{v}}_F \\
\dot{\tilde{\mathbf{v}}}_F = \dot{\mathbf{v}}_{Fd} + g\mathbf{e} - \frac{f_F}{m_F} R_{Fq}\mathbf{e} \\
\begin{bmatrix} \dot{\tilde{q}}_{F0} \\ \dot{\tilde{\mathbf{q}}}_F \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \tilde{\mathbf{q}}_F^T \tilde{\omega}_F \\ -\left(\tilde{q}_{F0} I + S(\tilde{\mathbf{q}}_F)\right) \tilde{\omega}_F \end{bmatrix} \\
J_F \dot{\tilde{\omega}}_F = S(\tilde{\omega}_F) J_F \tilde{\omega}_F + G(\tilde{\omega}_F) - \tau_{Fq}
\end{cases} . \tag{4.4}$$

Consider the external disturbances $\mathbf{d}_F = [\mathbf{d}_{\mathbf{v}F}^T, \mathbf{d}_{\omega F}^T]^T$ applied to the nonlinear system (4.4).

Let

$$\mathbf{x}_F = \begin{bmatrix} \tilde{\mathbf{p}}_F \\ \tilde{q}_{F0} \\ \tilde{\mathbf{q}}_F \\ \tilde{\mathbf{v}}_F \\ \tilde{\omega}_F \end{bmatrix}$$

$$\mathbf{u}_F = \begin{bmatrix} \dot{\mathbf{v}}_{Fd} + g\mathbf{e} - \frac{f_F}{m_F} R_{Fq}\mathbf{e} \\ G(\tilde{\omega}_F) - \tau_{Fq} \end{bmatrix}.$$

The nonlinear dynamic system (4.4) with the disturbance vector $\mathbf{d}_F$ can be written into an affine nonlinear form:

$$\dot{\mathbf{x}}_F = f(\mathbf{x}_F) + g(\mathbf{x}_F)\mathbf{u}_F + k(\mathbf{x}_F)\mathbf{d}_F \tag{4.5}$$

where

$$f(\mathbf{x}_F) = \begin{bmatrix} \tilde{\mathbf{v}}_F \\ \frac{1}{2}\tilde{\mathbf{q}}_F^T\tilde{\omega}_F \\ -\frac{1}{2}\left(\tilde{q}_{F0}I + S(\tilde{\mathbf{q}}_F)\right)\tilde{\omega}_F \\ 0_{3\times 1} \\ J_F^{-1}S(\tilde{\omega}_F)J_F\tilde{\omega}_F \end{bmatrix}$$

$$g(\mathbf{x}_F) = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} \\ 0_{1\times 3} & 0_{1\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \\ I & 0_{3\times 3} \\ 0_{3\times 3} & J_F^{-1} \end{bmatrix}$$

$$k(\mathbf{x}_F) = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} \\ 0_{1\times 3} & 0_{1\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \\ I & 0_{3\times 3} \\ 0_{3\times 3} & J_F^{-1} \end{bmatrix}.$$

The $H_\infty$ suboptimal control approach is used to design the follower controller. By defining an energy function, the follower controller is obtained as below by following a similar procedure to that described in Section 3.2.3 for stability analysis.

The total force and the torque vector are applied to the follower, $f_F$ and $\tau_{Fq} \in \mathbb{R}^3$;

$$\begin{aligned} f_F =& (k_{Fz}\tilde{z}_F + k_{Fv_z}\tilde{v}_{Fz} + \dot{v}_{Lz} - d(R_{q31}\cos\rho\cos\sigma + R_{q32}\cos\rho\sin\sigma + R_{q33}\sin\rho) + g) \\ & \frac{m_F}{q_{F0}^2 - q_{F1}^2 - q_{F2}^2 + q_{F3}^2} \\ \tau_{Fq} =& K_{Fq}\tilde{\mathbf{q}}_F + K_{F\omega}\tilde{\omega}_F + G(\tilde{\omega}_F) \end{aligned}$$

where

$$\ddot{R}_{Lq}^T = \begin{bmatrix} R_{q11} & R_{q12} & R_{q13} \\ R_{q21} & R_{q22} & R_{q23} \\ R_{q31} & R_{q32} & R_{q33} \end{bmatrix}.$$

## 4.4 Simulations

A scenario of two identical quadrotors using a MATLAB simulator was considered to track a desired path for the leader and maintain the desired distance, desired incidence angle and desired bearing angle between them for the follower. The quadrotor parameters used in the simulation are described in Table 3.1.

Two predefined or desired paths were tested. The first path for the leader to track was

$$\begin{cases} x_{Ld} = 2\cos(t\pi/80) & ; \quad y_{Ld} = 2\sin(t\pi/80) \\ z_{Ld} = 1 + 0.1t & ; \quad q_{L3d} = 0 \end{cases}$$

with the initial conditions $\mathbf{p}_L = [2,0,0]^T$ metres and $[q_{L0}, \mathbf{q}_L^T]^T = [-1,0,0,0]^T$. The follower tried to maintain the desired distance with the leader $d = 2$ metres, the desired incidence angle $\rho = 0$ and the desired bearing angle $\sigma = -\pi/12$. The initial condition of the follower was $\mathbf{p}_F = [0.1, 0.5, 0]^T$ metres and $[q_{F0}, \mathbf{q}_F^T]^T = [-1,0,0,0]^T$.

The second path was

$$\begin{cases} x_{Ld} = \sin(t\pi/20) \quad ; \quad y_{Ld} = \cos(t\pi/20) \\ z_d = 1 + 0.1t \quad ; \quad q_{L3d} = 0 \end{cases}$$

with the initial conditions $\mathbf{p}_L = [0, 1, 0]^T$ metres and $[q_{L0}, \mathbf{q}_L^T]^T = [-1, 0, 0, 0]^T$. The follower tried to maintain the desired distance with the leader $d = 2$ metres, the desired incidence angle $\rho = \pi/6$ and the desired bearing angle $\sigma = 0$. The initial condition of the follower was $\mathbf{p}_F = [-1.7, 1, 0]^T$ metres and $[q_{F0}, \mathbf{q}_F^T]^T = [-1, 0, 0, 0]^T$.

For the first path, the constant $\gamma$ was chosen to be $\gamma_L = \gamma_F = 1.05$ and the weighting matrices were chosen to be $W_{L1z} = 1150$, $W_{F1z} = 1575$, $W_{L2} = W_{F2} = diag(0.0235, 0.0235, 0.0009)$, $W_{L3z} = 10$, $W_{F3z} = 675$, and $W_{L4} = W_{F4} = diag(0.0043, 0.0043, 0.00156)$. Under these parameters, the feedback control matrices were obtained to be $k_{Lz} = 111$, $k_{Lv_z} = 50$, $k_{Fz} = 130$, $k_{Fv_z} = 100$, $K_{Lq} = K_{Fq} = diag(0.5, 0.5, 0.095)$ and $K_{L\omega} = K_{F\omega} = diag(0.07, 0.07, 0.025)$.

For the second path, the constant $\gamma$ was chosen to be $\gamma_L = \gamma_F = 1.05$, and the weighting matrices were chosen to be $W_{L1z} = W_{F1z} = 1150$, $W_{L2} = W_{F2} = diag(0.0235, 0.0235, 0.0009)$, $W_{L3z} = W_{F3z} = 10$ and $W_{L4} = W_{F4} = diag(0.0043, 0.0043, 0.00156)$. Under these parameters, the feedback control matrices were obtained as, $k_{Lz} = k_{Fz} = 111$, $k_{Lv_z} = k_{Fv_z} = 50$, $K_{Lq} = K_{Fq} = diag(0.5, 0.5, 0.095)$ and $K_{L\omega} = K_{F\omega} = diag(0.07, 0.07, 0.025)$.

The obtained results are shown in Figures 4.2 - 4.8 with the conditions (1) no disturbance, (2) force disturbance $d_{v_{iz}} = -2$Nm at $10 \leq t \geq 10.25$ seconds, $d_{v_{ix}} = 2$Nm at $20 \leq t \geq 20.25$ seconds, $d_{v_{iy}} = 2$Nm at $30 \leq t \geq 30.25$ seconds and the attitude part for the leader and the follower is disturbed using (3.27), (3) +20% model parameter uncertainty, and (4) −20% model parameter uncertainty. The above conditions were applied for the leader and the follower at the same time.

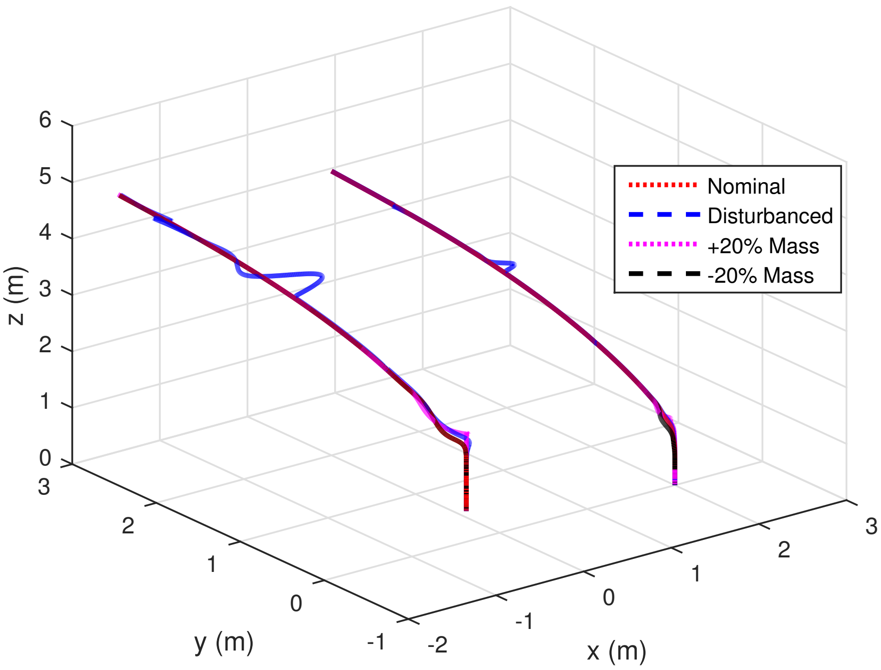

Figure 4.2: Leader-Follower Formation in First Path under $H_\infty$ Controller Based on Quaternion Representation

Figures 4.2 and 4.5 show the formation trajectories of two quadrotors obtained using the $H_\infty$ controllers when they tracked path 1 and path 2, respectively. From these figures we can see that the $H_\infty$ controllers produced good formation performances with small acceptable errors, fast rejection of the external disturbances, and quick recovery of the model parameter uncertainties. The quaternions of the leader and the follower for path 1 and path 2 are shown in Figures 4.3, 4.4, 4.6, and 4.7 with small oscillations. The distances between the leader and the follower for the two paths are shown in Figure 4.8. Again, less oscillation in disturbance rejection was observed from the result of both paths.

Figures 4.9 and 4.10 show the performance of both paths when only the leader was affected by force disturbance $d_{v_{Lz}} = -4$Nm during $10 \leq t \geq 10.25$ seconds, $d_{v_{Lx}} = 4$Nm during $20 \leq t \geq 20.25$ seconds, $d_{v_{Ly}} = 4$Nm during $30 \leq t \geq 30.25$ seconds, and the leader attitude part is disturbed using (3.27).

It is clear that the follower tracked the leader and maintained the distance with very small errors in all circumstances.

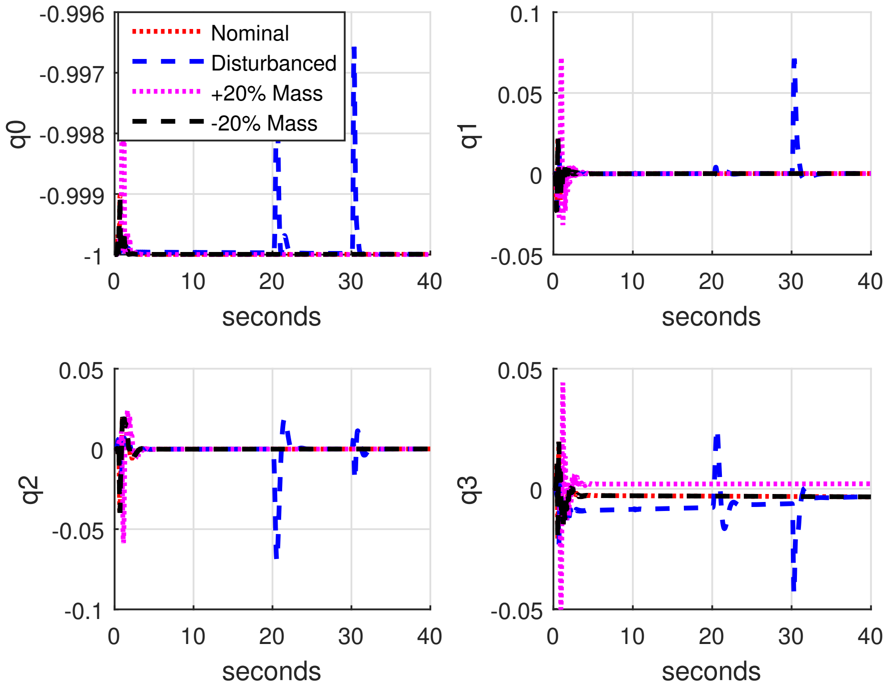

Figure 4.3: Leader Quaternions in First Path under $H_\infty$ Controller Based on Quaternion Representation

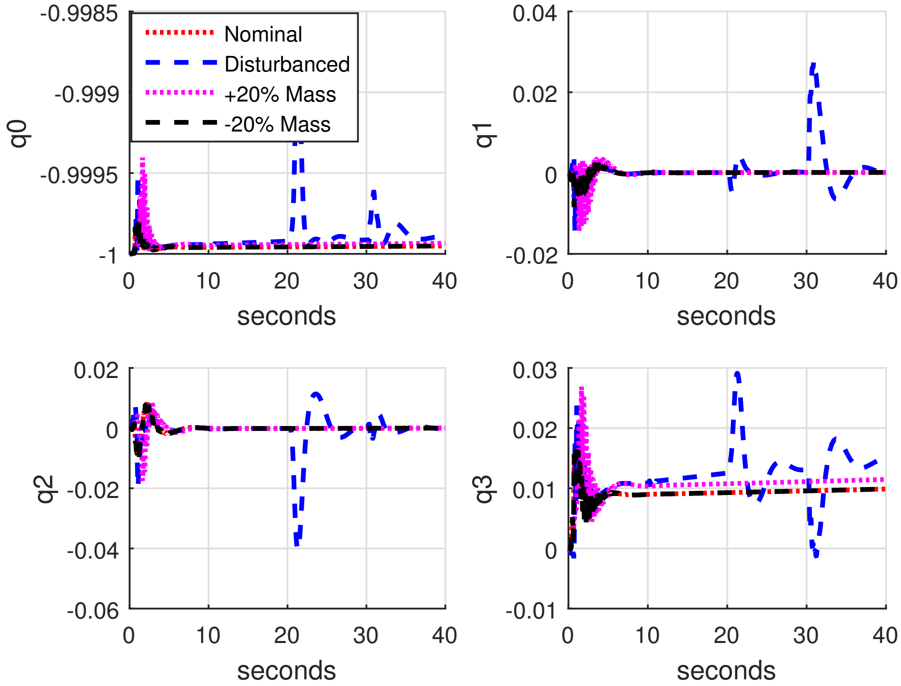

Figure 4.4: Follower Quaternions in First Path under $H_\infty$ Controller Based on Quaternion Representation

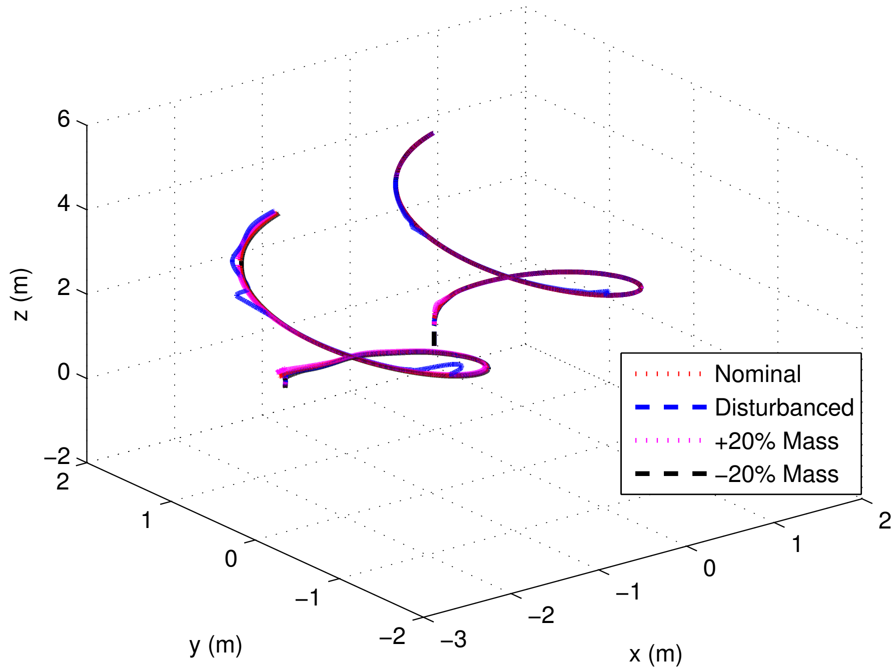

FIGURE 4.5: Leader-Follower Formation in Second Path under $H_\infty$ Controller Based on Quaternion Representation

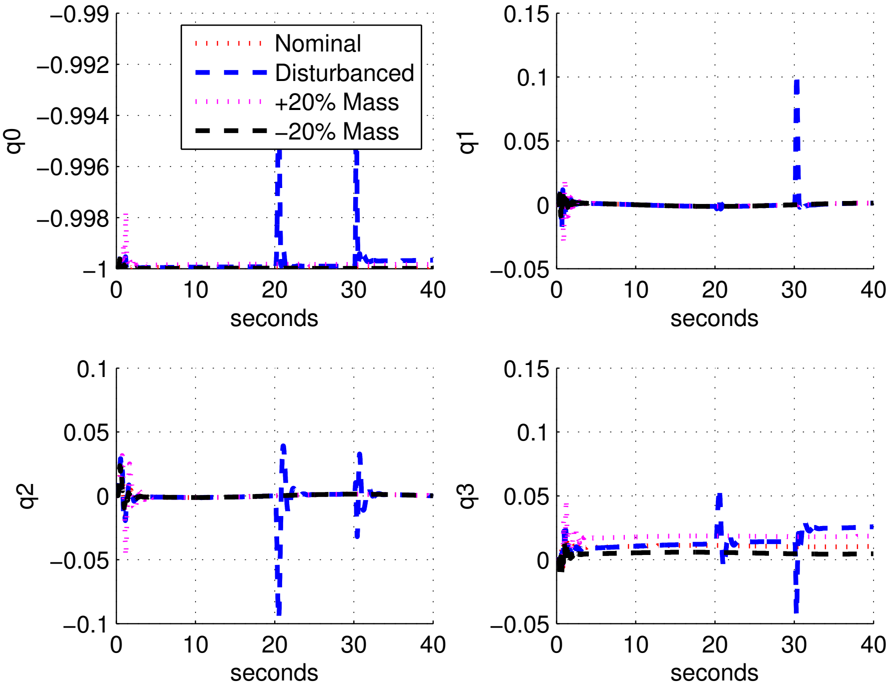

FIGURE 4.6: Leader Quaternions in Second Path under $H_\infty$ Controller Based on Quaternion Representation

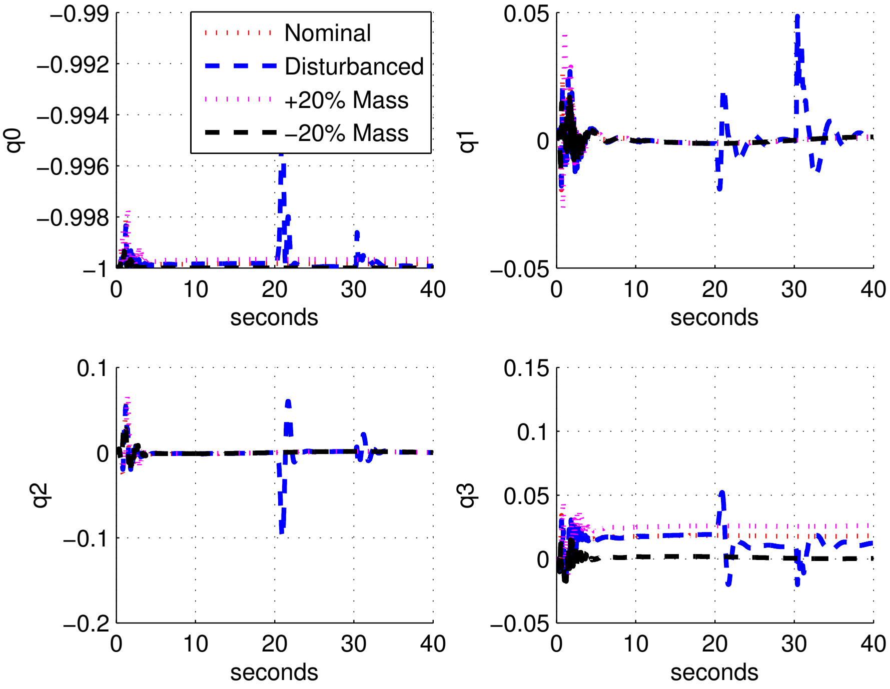

FIGURE 4.7: Follower Quaternions in Second Path under $H_\infty$ Controller Based on Quaternion Representation

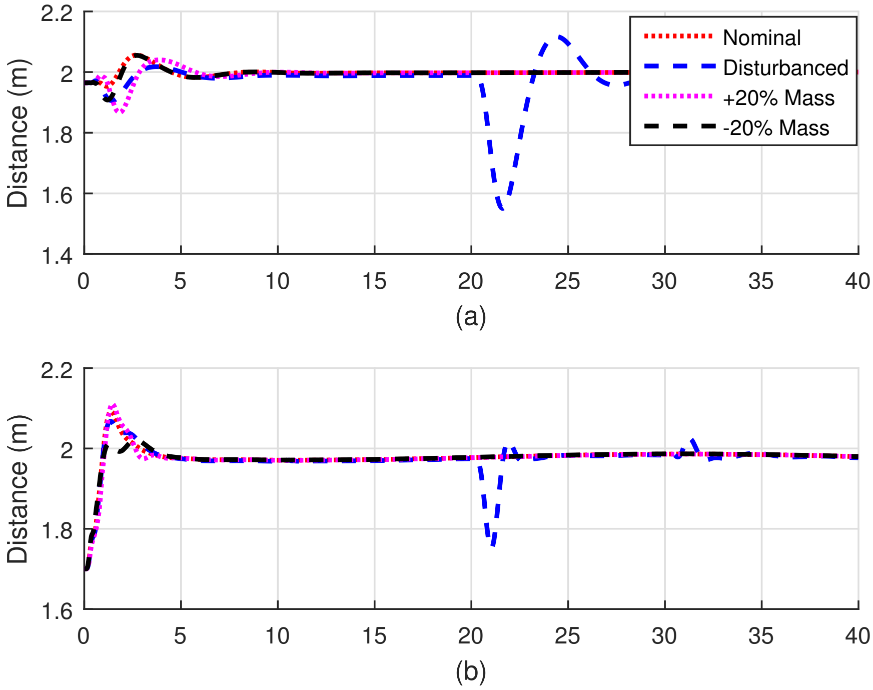

FIGURE 4.8: The Distance Between the Leader and the Follower under $H_\infty$ Controller Base on Quaternion Representation in (a) The First Path, (b) The Second Path

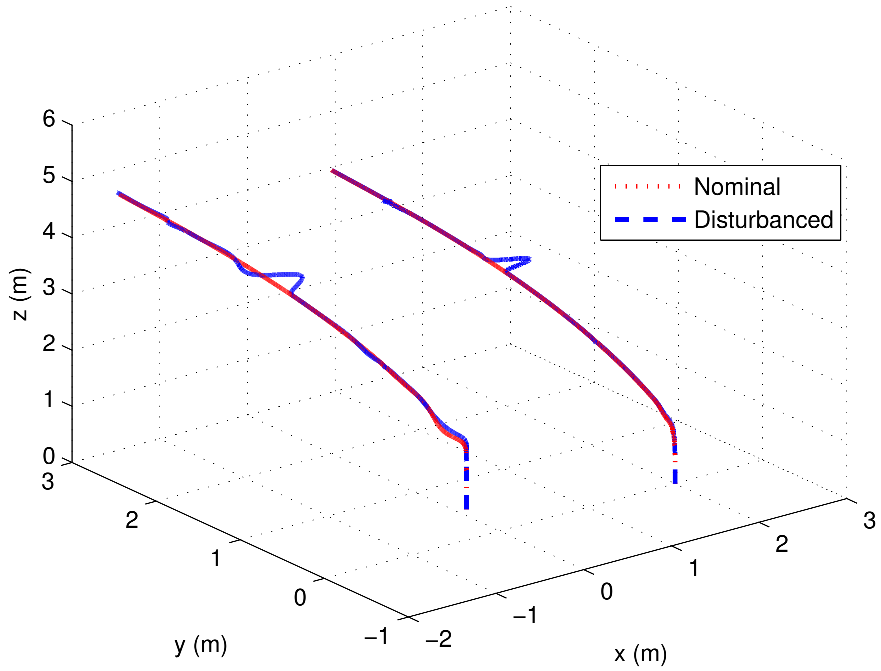

FIGURE 4.9: Leader-Follower Formation in First Path under $H_\infty$ Controller Based on Quaternion Representation with Leader Disturbance Only

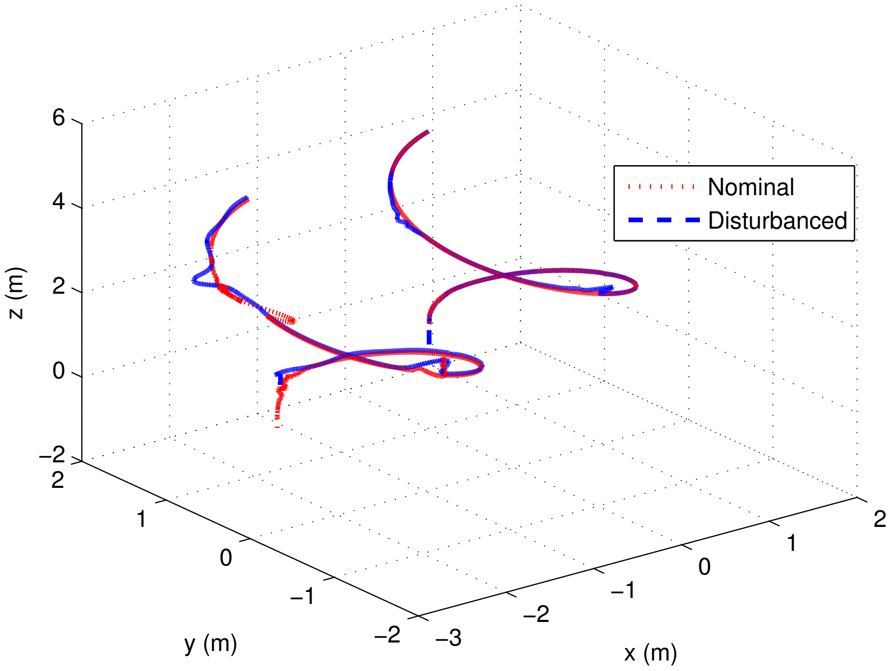

FIGURE 4.10: Leader-Follower Formation in Second Path under $H_\infty$ Controller Based on Quaternion Representation with Leader Disturbance Only

## 4.5 The Leader-Follower Formation Problem Based on Euler Angles Representation

The full dynamic model based on Euler angles (3.29) of a quadrotor can be written as:

$$\begin{cases} \dot{\mathbf{p}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = -g\mathbf{e} + \frac{f_i}{m_i} R_{i\theta}\mathbf{e} \\ \dot{\zeta}_i = \eta_i \\ J_i\dot{\eta}_i = S(\eta_i)J_i\eta_i + G(\eta_i) - \tau_{iE} \end{cases} \tag{4.6}$$

and the formation problem is to satisfy the following conditions:

$$\begin{cases} \lim_{t\to\infty}(\mathbf{p}_{Fd} - \mathbf{p}_F) = 0 \\ \lim_{t\to\infty}(\psi_L - \psi_F) = 0 \end{cases} \tag{4.7}$$

and

$$\begin{cases} \lim_{t\to\infty}(\mathbf{p}_{Ld} - \mathbf{p}_L) = 0 \\ \lim_{t\to\infty}(\psi_{Ld} - \psi_L) = 0 \end{cases} \tag{4.8}$$

where

$$\mathbf{p}_{Fd} = \mathbf{p}_L - R_{L\theta}^T d \begin{bmatrix} \cos\rho\cos\sigma \\ \cos\rho\sin\sigma \\ \sin\rho \end{bmatrix}.$$

The communication among the robots is assumed to be available. The position $\mathbf{p}_L$, yaw angle $\psi_L$ of the leader $L$ and its first and second derivatives $\dot{\psi}_L$ and $\ddot{\psi}_L$ are assumed to be available and measurable. The linear velocity of the leader $L$ and its derivatives $\mathbf{v}_L$ and $\dot{\mathbf{v}}_L$ are assumed bounded and available for the follower.

## 4.6 Formation $H_\infty$ Controllers

The controller design for the leader and the follower is based on $H_\infty$ suboptimal control as well. The follower $H_\infty$ controller is designed by following the introduction of an error state model, and the introduction of a $H_\infty$ control theorem for general affine systems. The leader $H_\infty$ controller was presented in Section 3.4.2.

### 4.6.1 Follower State Error Model

The control strategy for the follower is to track the desired position $\mathbf{p}_{Fd}$. The tracking errors for the follower according to the nonlinear dynamic system (4.6) can be written as:

$$\begin{aligned} \tilde{\mathbf{p}}_F &= \mathbf{p}_{Fd} - \mathbf{p}_F \\ \tilde{\mathbf{v}}_F &= \mathbf{v}_{Fd} - \mathbf{v}_F \\ \tilde{\zeta}_F &= \zeta_{Fd} - \zeta_F \\ \tilde{\eta}_F &= \eta_{Fd} - \eta_F \end{aligned}$$

where $\mathbf{v}_{Fd} = \dot{\mathbf{p}}_{Fd}$ is the desired linear velocity, $[\zeta_{Fd}] = [0, 0, \psi_L]^T$ is the desired angles, and $[\eta_{Fd}] = [0, 0, 0]^T$ is the desired angles derivative.

### 4.6.2 Follower $H_\infty$ Controller

Equation (4.6) can be rewritten in an error form as:

$$\begin{cases} \dot{\tilde{\mathbf{p}}}_F = \tilde{\mathbf{v}}_F \\ \dot{\tilde{\mathbf{v}}}_F = \dot{\mathbf{v}}_{Fd} + g\mathbf{e} - \frac{f_F}{m_F} R_{F\theta}\mathbf{e} \\ \dot{\tilde{\zeta}}_F = \tilde{\eta}_F \\ J_F \dot{\tilde{\eta}}_F = S(\tilde{\eta}_F) J_F \tilde{\eta}_F + G(\tilde{\eta}_F) - \tau_{FE} \end{cases} . \tag{4.9}$$

Consider the external disturbances $\mathbf{d}_F = [\mathbf{d}_{\mathbf{v}F}^T, \mathbf{d}_{\eta F}^T]^T$ applied to the nonlinear system (4.9).

Let

$$\mathbf{x}_F = \begin{bmatrix} \tilde{\mathbf{p}}_F \\ \tilde{\zeta}_F \\ \tilde{\mathbf{v}}_F \\ \tilde{\eta}_F \end{bmatrix}$$

$$\mathbf{u}_F = \begin{bmatrix} \dot{\mathbf{v}}_{Fd} + g\mathbf{e} - \frac{f_F}{m_F} R_{F\theta}\mathbf{e} \\ G(\tilde{\eta}_F) - \tau_{FE} \end{bmatrix}.$$

The nonlinear dynamic system (4.9) with the disturbance vector $\mathbf{d}_F$ can be written into an affine nonlinear form:

$$\dot{\mathbf{x}}_F = f(\mathbf{x}_F) + g(\mathbf{x}_F)\mathbf{u}_F + k(\mathbf{x}_F)\mathbf{d}_F \tag{4.10}$$

where

$$f(\mathbf{x}_F) = \begin{bmatrix} \tilde{\mathbf{v}}_F \\ \tilde{\eta}_F \\ 0_{3\times1} \\ J_F^{-1} S(\tilde{\eta}_F) J_F \tilde{\eta}_F \end{bmatrix}$$

$$g(\mathbf{x}_F) = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \\ I & 0_{3\times3} \\ 0_{3\times3} & J_F^{-1} \end{bmatrix}$$

$$k(\mathbf{x}_F) = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \\ I & 0_{3\times3} \\ 0_{3\times3} & J_F^{-1} \end{bmatrix}.$$

The $H_\infty$ suboptimal control approach is used to design the follower controller. By defining an energy function, the follower controller is obtained as below by following a similar procedure to that described in Section 3.4.2 for stability analysis.

Then from $\mathbf{u}_F$, we can have

$$\begin{aligned}\mathbf{u}_F &= \begin{bmatrix} \dot{\mathbf{v}}_{Fd} + g\mathbf{e} - \frac{f_F}{m_F} R_{F\theta}\mathbf{e} \\ G(\tilde{\eta}_F) - \tau_{FE} \end{bmatrix} \\ &= - \begin{bmatrix} K_{Fp}\tilde{\mathbf{p}}_F + K_{Fv}\tilde{\mathbf{v}}_F \\ K_{F\zeta}\tilde{\zeta}_F + K_{F\eta}\tilde{\eta}_F \end{bmatrix}.\end{aligned}$$

Then the total force and the torque vector are applied to the follower, $f_F$ and $\tau_{FE} \in \mathbb{R}^3$;

$$\begin{aligned}f_F =&(k_{Fz}\tilde{z}_F + k_{Fv_z}\tilde{v}_{Fz} + \dot{v}_{Lz} - d(R_{\theta 31}\cos\rho\cos\sigma + R_{\theta 32}\cos\rho\sin\sigma + R_{\theta 33}\sin\rho) + g) \\ &\frac{m_F}{c\varphi_L c\theta_L} \\ \tau_{FE} =& K_{F\zeta}\tilde{\zeta}_F + K_{F\eta}\tilde{\eta}_F + G(\tilde{\eta}_F)\end{aligned}$$

where

$$\ddot{R}_{L\theta}^T = \begin{bmatrix} R_{\theta 11} & R_{\theta 12} & R_{\theta 13} \\ R_{\theta 21} & R_{\theta 22} & R_{\theta 23} \\ R_{\theta 31} & R_{\theta 32} & R_{\theta 33} \end{bmatrix}.$$

## 4.7 Simulations

Two paths were presented in the simulation to show the performance of using the proposed controller with four different circumstances. The first desired path to be tracked by the leader was

$$\begin{cases} x_{Ld} = 2\cos(t\pi/80) \quad ; \quad y_{Ld} = 2\sin(t\pi/80) \\ z_{Ld} = 1 + 0.1t \quad ; \quad \psi_{Ld} = \pi/6 \end{cases}.$$

The leader initial positions were $[x_L, y_L, z_L]^T = [2, 0, 0]^T$ metres and the initial angles were $[\varphi_L, \theta_L, \psi_L]^T = [0, 0, 0]^T$ radian. Then the follower followed the leader and maintained the desired distance between them $d = 2$ metres, the desired incidence and bearing angles $\rho = -\pi/6$, $\sigma = \pi/6$ radian, respectively. The follower initial positions were $[x_F, y_F, z_F]^T = [0.5, 0, 0]^T$ metres and the initial angles were $[\varphi_F, \theta_F, \psi_F]^T = [0, 0, 0]^T$ radian. The second desired path to be tracked by the leader was

$$\begin{cases} x_{Ld} = 4\cos(t\pi/40) \quad ; \quad y_{Ld} = 4\sin(t\pi/40) \\ z_{Ld} = 1 + 0.1t \quad ; \quad \psi_{Ld} = \pi/6 \end{cases}.$$

The leader initial positions were $[x_L, y_L, z_L]^T = [4, 0, 0]^T$ metres and the initial angles were $[\varphi_L, \theta_L, \psi_L]^T = [0, 0, 0]^T$ radian. Then the follower followed the leader and maintained the desired distance between them $d = 3$ metres, the desired incidence and bearing angles $\rho = 0$, $\sigma = \pi/6$ radian, respectively. The follower initial positions were $[x_F, y_F, z_F]^T = [1.4, -1.5, 0]^T$ metres and the initial angles were $[\varphi_F, \theta_F, \psi_F]^T = [0, 0, 0]^T$ radian.

Figures 4.11 and 4.14 indicate the response of the proposed $H_\infty$ when the leader tracked the first and second desired paths, respectively. Figure 4.17 shows the distances between the leader and the follower in the two paths, and Figures 4.12, 4.13, 4.15 and 4.16 illustrate the yaw angle behaviour for the leader and the follower via the two paths, respectively. Its four circumstances included: (1) no disturbance, (2) force disturbance $d_{v_{ix}} = -2$Nm during $10 \leq t \geq 10.25$ seconds, $d_{v_{iz}} = 2$Nm during $20 \leq t \geq 20.25$ seconds, $d_{v_{iy}} = 2$Nm during $30 \leq t \geq 30.25$ seconds in the first path, $d_{v_{ix}} = -2$Nm during $20 \leq t \geq 20.25$ seconds, $d_{v_{iz}} = 2$Nm during $60 \leq t \geq 60.25$ seconds, $d_{v_{iy}} = 2$Nm during $100 \leq t \geq 100.25$ seconds in the second path, and the attitude part for the leader and the follower is disturbed using (3.27), applied at the same time for both the leader and the follower, (3) +30% model parameter uncertainty, and (4) −30% model parameter uncertainty.

Figures 4.18 and 4.19 show the performance of both paths when only the leader was affected by force disturbance $d_{v_{Lx}} = -4$Nm during $20 \leq t \geq 20.25$ seconds, $d_{v_{Lz}} = 4$Nm during $60 \leq t \geq 60.25$ seconds, $d_{v_{Ly}} = 4$Nm during $100 \leq t \geq 100.25$ seconds, and the leader attitude part is disturbed using (3.27).

From Figures 4.11 - 4.19 the overshoots of using the $H_\infty$ controller were very small and the RMSE values of the desired distances between the leader and the follower

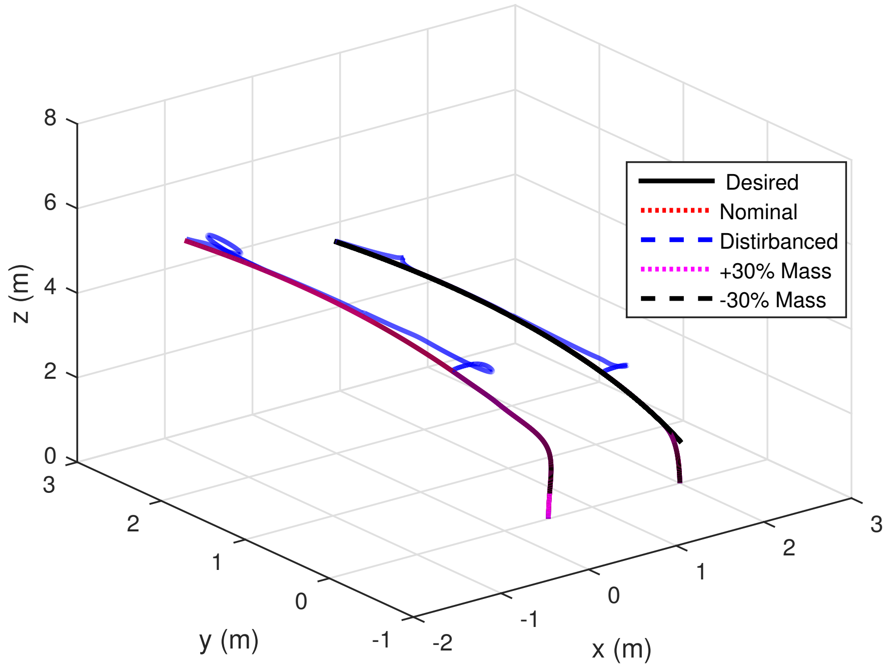

Figure 4.11: Leader-Follower Formation in First Path under $H_\infty$ Controller Based on Euler Angles Representation

were also very small and the controller's performance was fast in rejecting the disturbances as well. As a result, the proposed $H_\infty$ controller indeed produced excellent control performance.

## 4.8 Experimental Results

Experimental results are presented in this section with one leader and one follower tested by using three different paths in an indoor flight environment with consideration of external disturbances and weight changes. The leader tracked a predefined path, then the follower used the leader's actual position to calculate its path to follow. The information used to control the leader and the follower was received from the IMU and the Vicon Motion Capture System. Figure 3.20 shows the control diagram used in this work; here it can be noticed that the communication between the computer and the quadrotors was linked via two Xbees: the first one was mounted on the quadrotor while the second one was connected

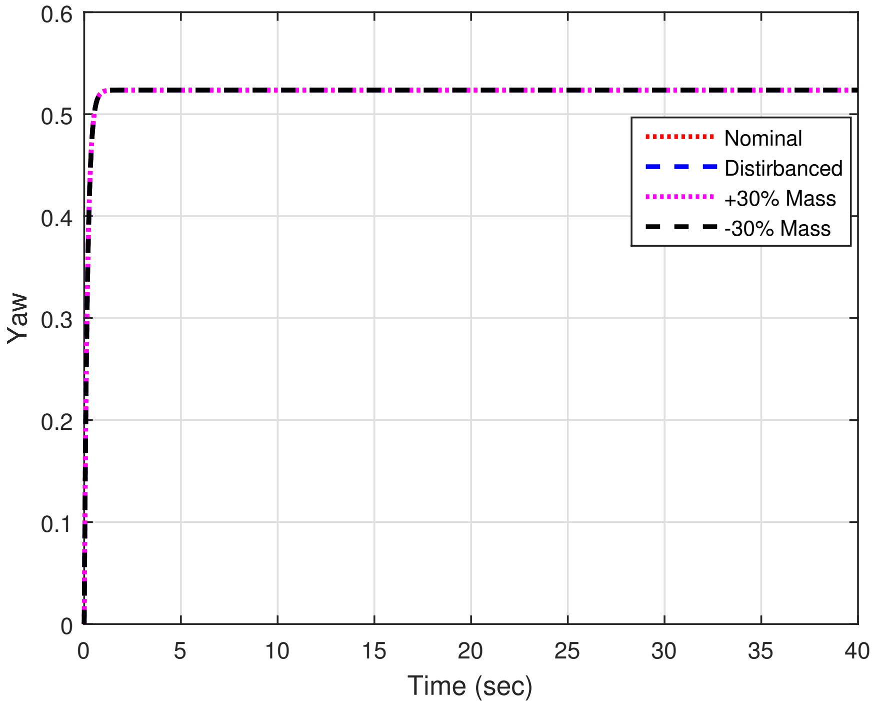

FIGURE 4.12: Leader Yaw Angle in First Path under $H_\infty$ Controller Based on Euler Angles Representation

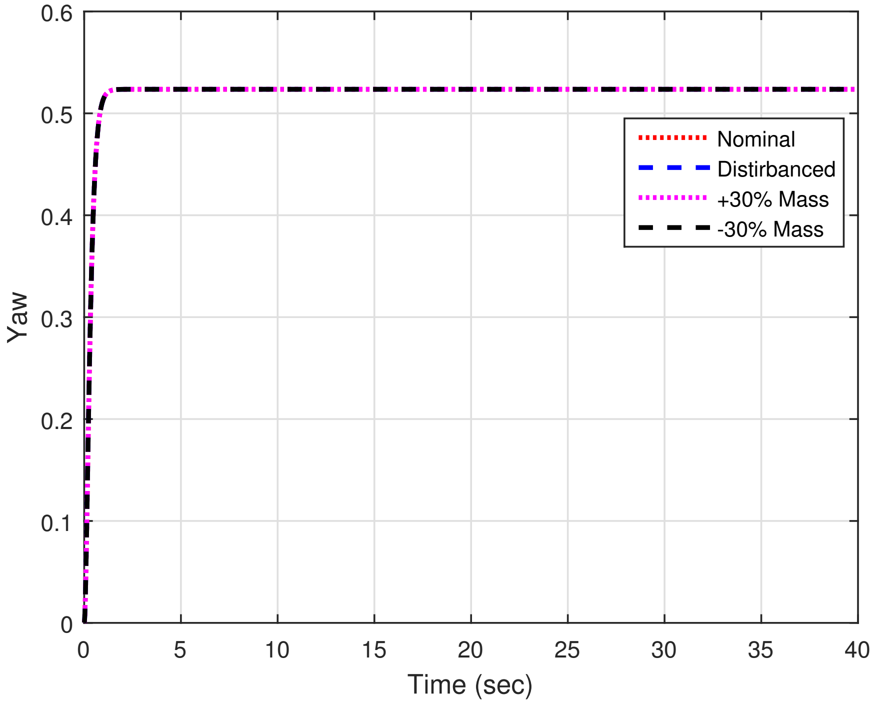

FIGURE 4.13: Follower Yaw Angle in First Path under $H_\infty$ Controller Based on Euler Angles Representation

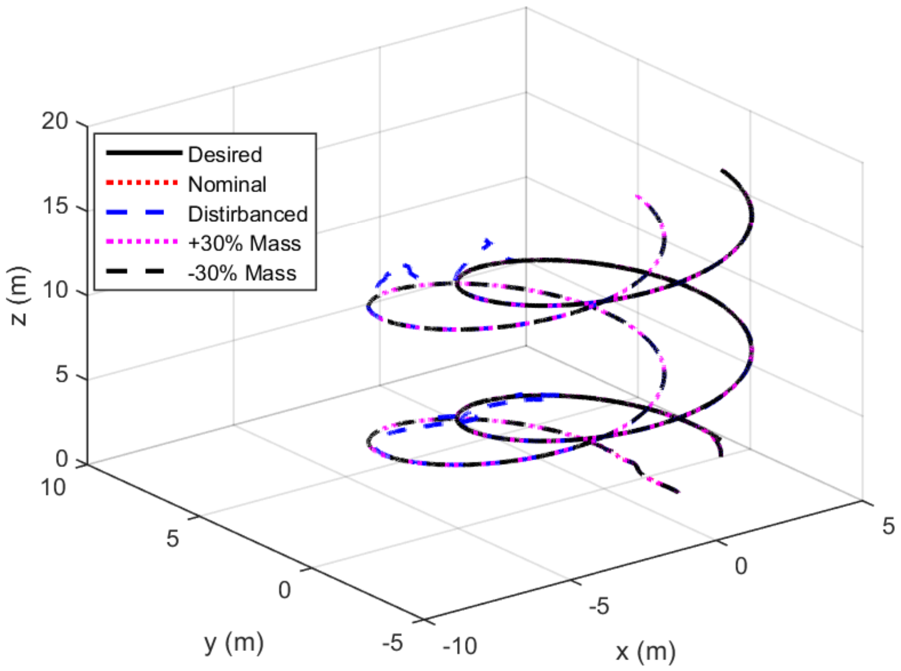

FIGURE 4.14: Leader-Follower Formation in Second Path under $H_\infty$ Controller Based on Euler Angles Representation

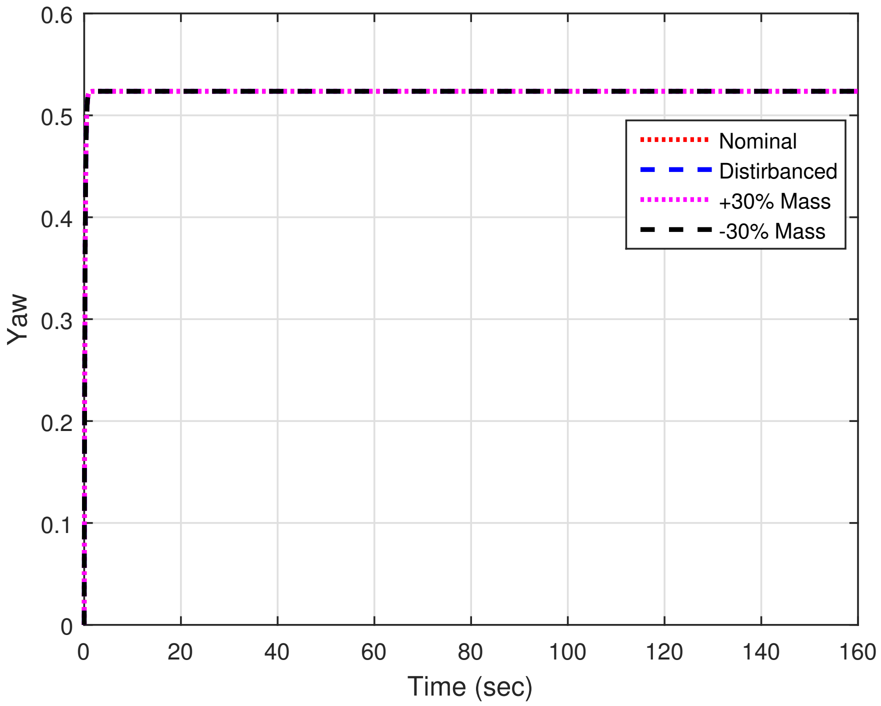

FIGURE 4.15: Leader Yaw Angle in Second Path under $H_\infty$ Controller Based on Euler Angles Representation

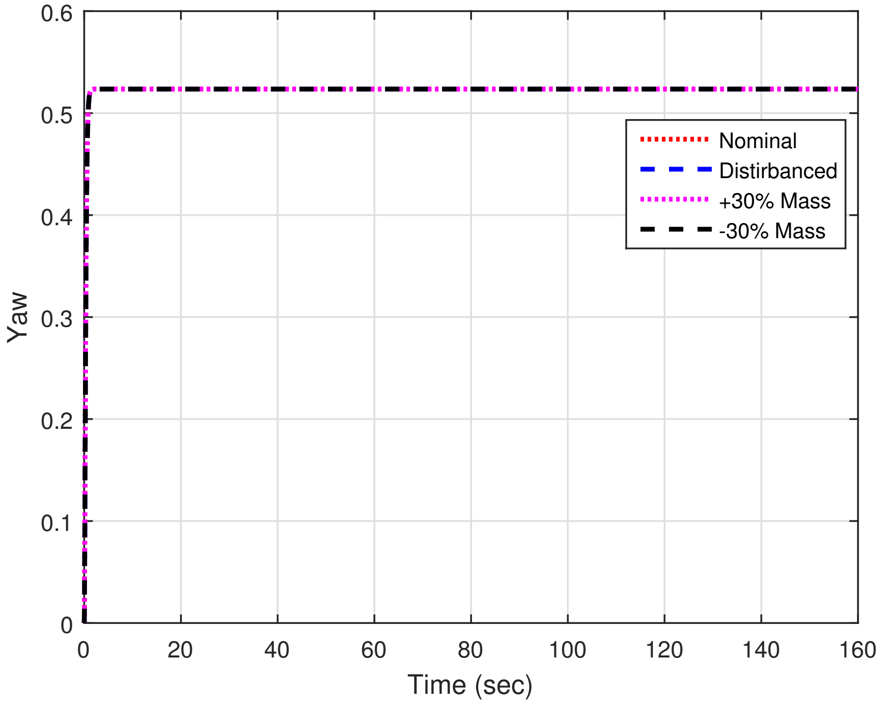

FIGURE 4.16: Follower Yaw Angle in Second Path under $H_\infty$ Controller Based on Euler Angles Representation

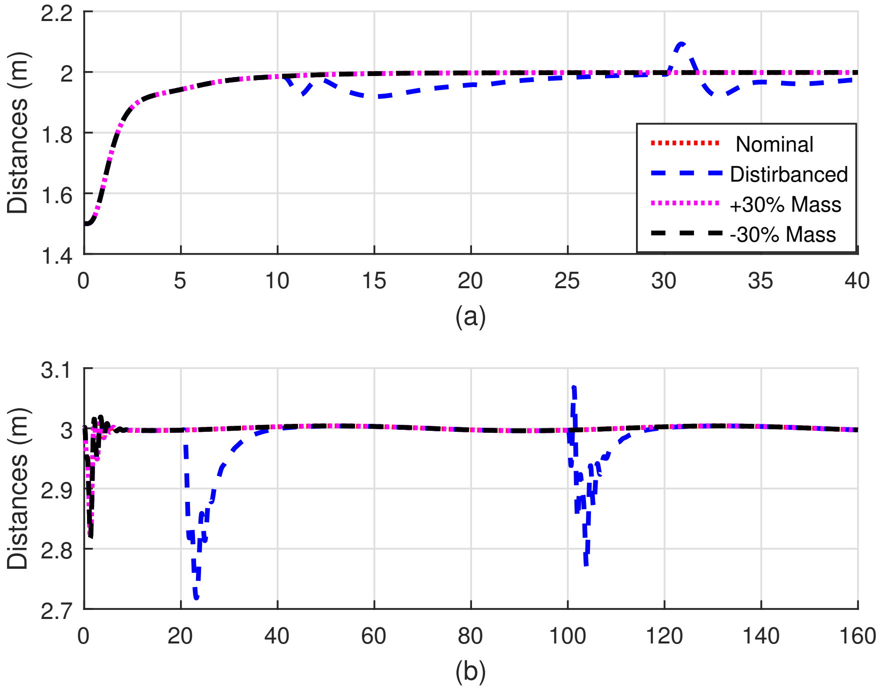

FIGURE 4.17: The Distance Between the Leader and the Follower under $H_\infty$ Controller Based on Euler Angles Representation in (a) The First Path, (b) The Second Path

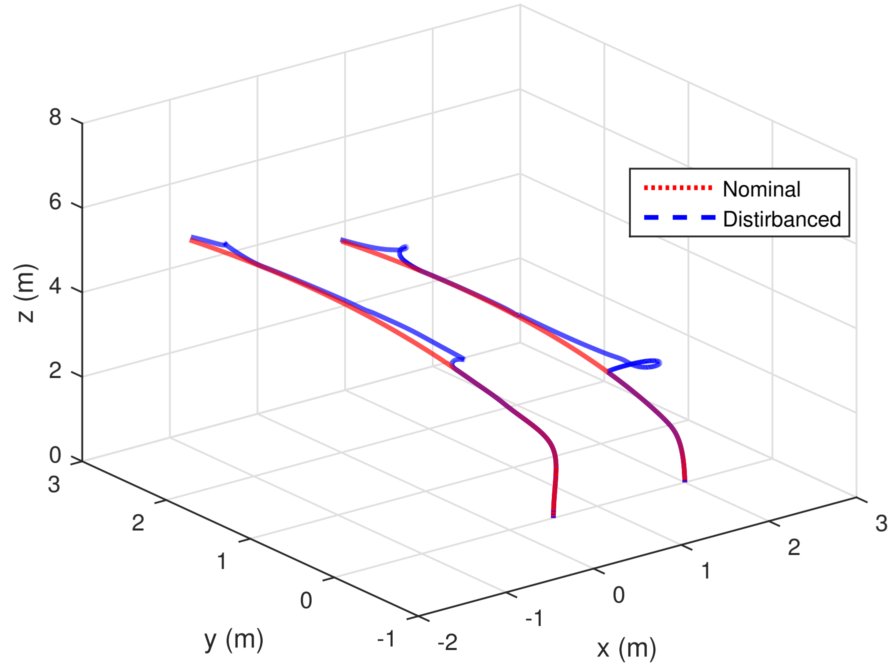

FIGURE 4.18: Leader-Follower Formation in First Path under $H_\infty$ Controller Based on Euler Angles with Leader Disturbance Only

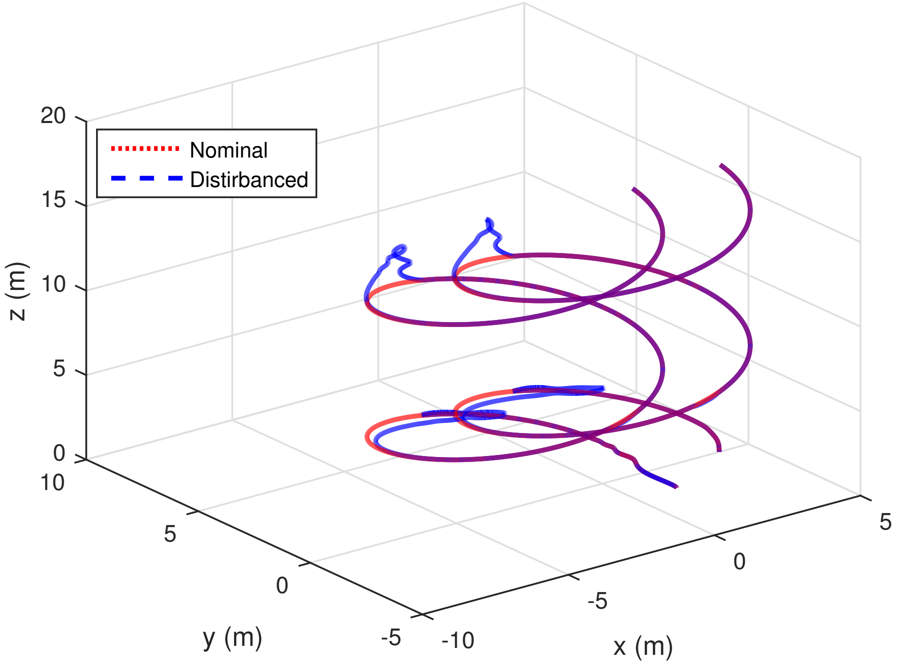

FIGURE 4.19: Leader-Follower Formation in Second Path under $H_\infty$ Controller Based on Euler Angles with Leader Disturbance Only

to the computer. The position of the leader was sent to the computer, and then it was sent from the computer to the follower to be used for the desired follower position. Then the controller outputs were sent to the vehicles via the Xbees to the high level on-board microcontrollers which directly control the motors speed. The two quadrotors in leader-follower formation flight is shown in Figure 4.24.

In the first test, both the leader and the follower took off to 0.5 metres and the formation controllers started from this point. Then the leader continued to climb to 1.5 metres, moved forward for 1 metre toward the x-direction, drew a square of 2 metres side length and then landed. The follower followed the leader and maintained the desired distance between them $d = 2$ metres, the desired incidence angle $\rho = \pi/12$ and the desired bearing angle $\sigma = -\pi/2$. In the second test, both the leader and the follower took off to 0.5 metres. Then the leader tracked a helical path of 2 metres circle diameter and 1.5 metres height, moved 1 metre towards the origin point and then landed. The follower tracked the leader and maintained the desired distance between them $d = 2$ metres, the desired incidence angle $\rho = -\pi/12$ and the desired bearing angle $\sigma = -\pi/2$. In the third test, both the leader and the follower took off to 0.5 metres. Then the leader tracked an eight-shaped path. The follower tracked the leader and maintained the desired distance between them $d = 2$ metres, the desired incidence angle $\rho = -\pi/12$ and the desired bearing angle $\sigma = -\pi/3$.

The experimental trajectories of the first, second and third tests are shown in Figures 4.20 - 4.22, respectively, with the conditions (1) no disturbance, (2) force disturbance, and (3) +20% mass. These conditions were applied to both the leader and the follower. Figure 4.23 illustrates the actual distance between the leader and the follower during the tests. The disturbances of the trajectories in these figures were caused by the external force exerted on the vehicles. These experimental results show that the follower successfully tracked the leader and maintained the distance, incidence and bearing angles between them with an acceptable error, less than 5 centimetres. The proposed controllers also show a good stability and robustness when considering the external disturbances and the mass change.

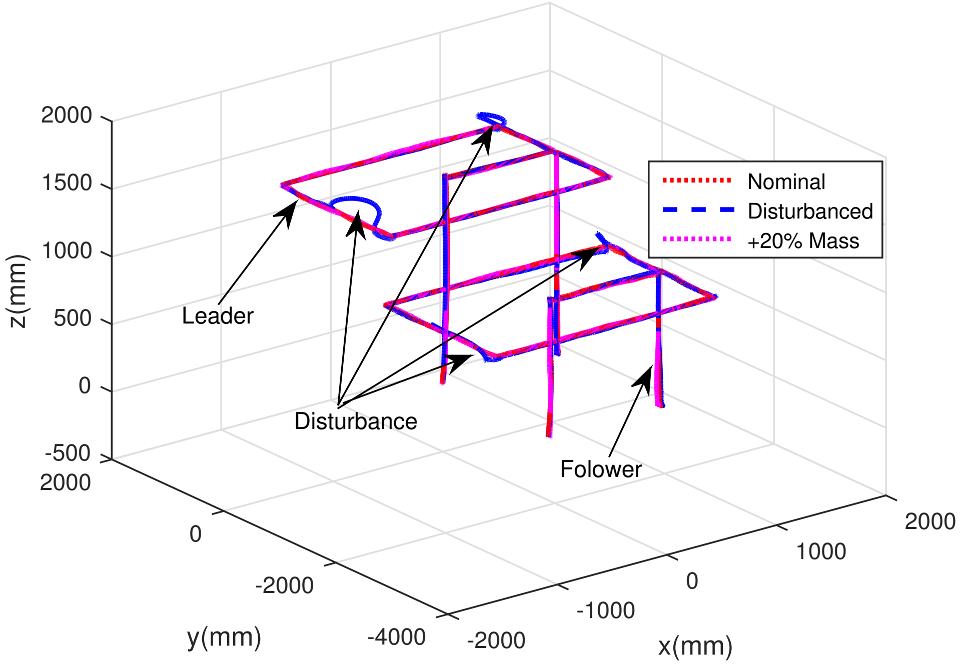

Figure 4.20: Leader-Follower Formation in the Square Path Test

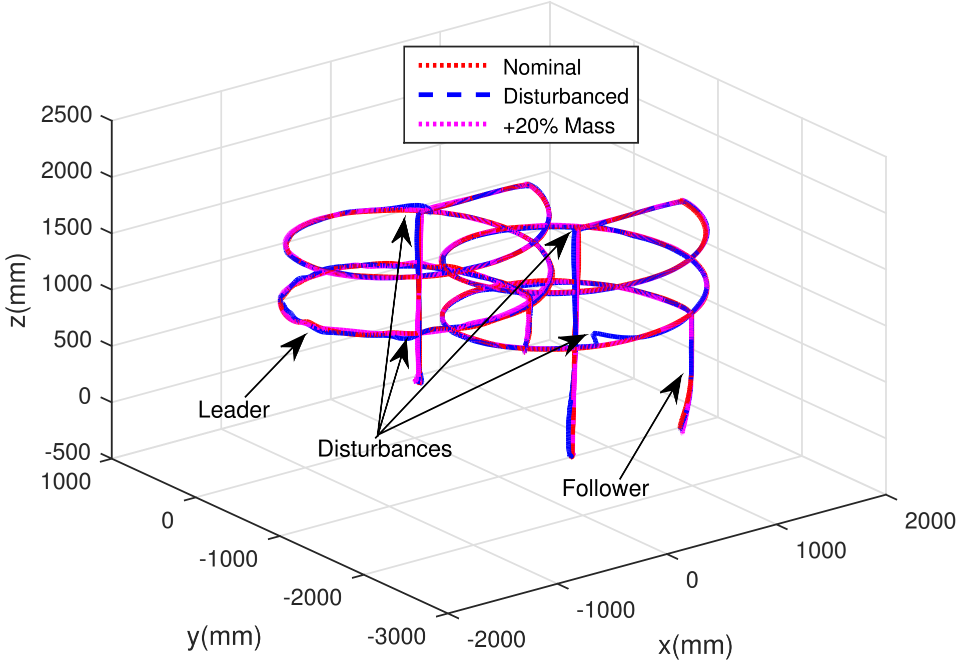

Figure 4.21: Leader-Follower Formation in the Helical Path Test

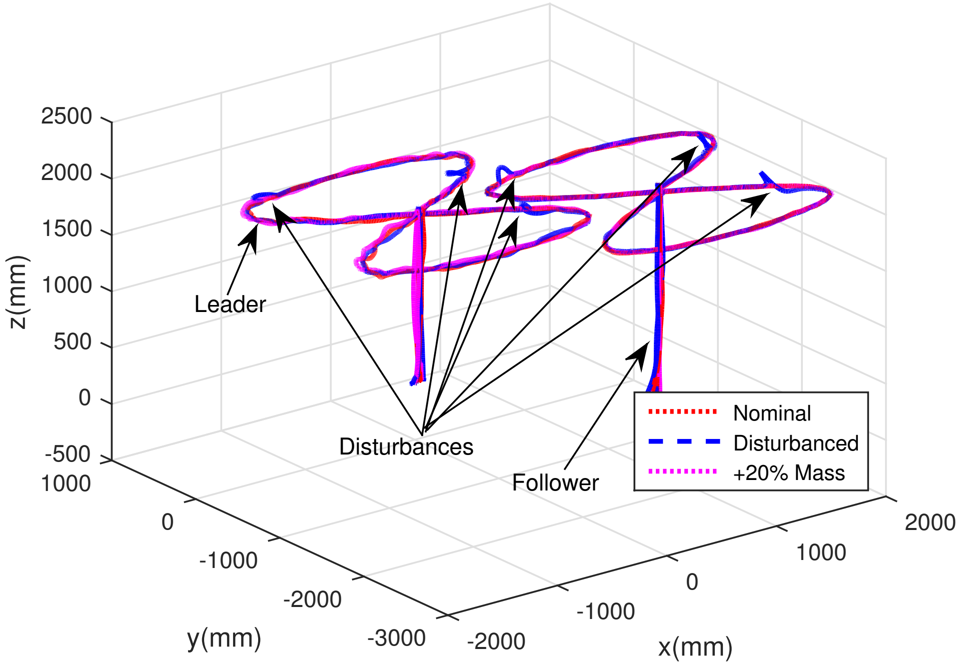

FIGURE 4.22: Leader-Follower formation in the Eight Shape Test

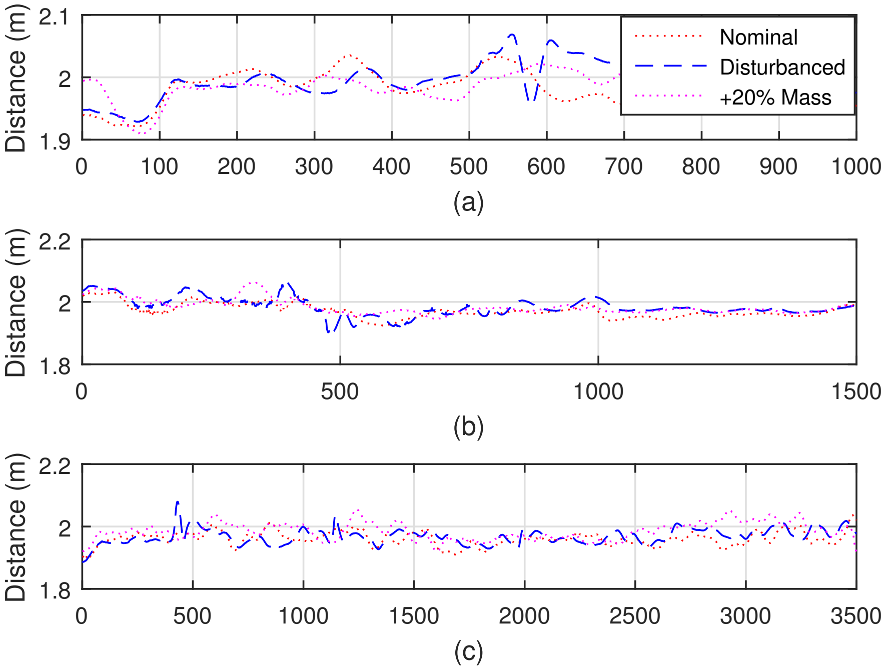

FIGURE 4.23: The Distance Between the Leader and the Follower in (a) The Square Test, (b) The Helical Test, (c) The Eight Shape Test

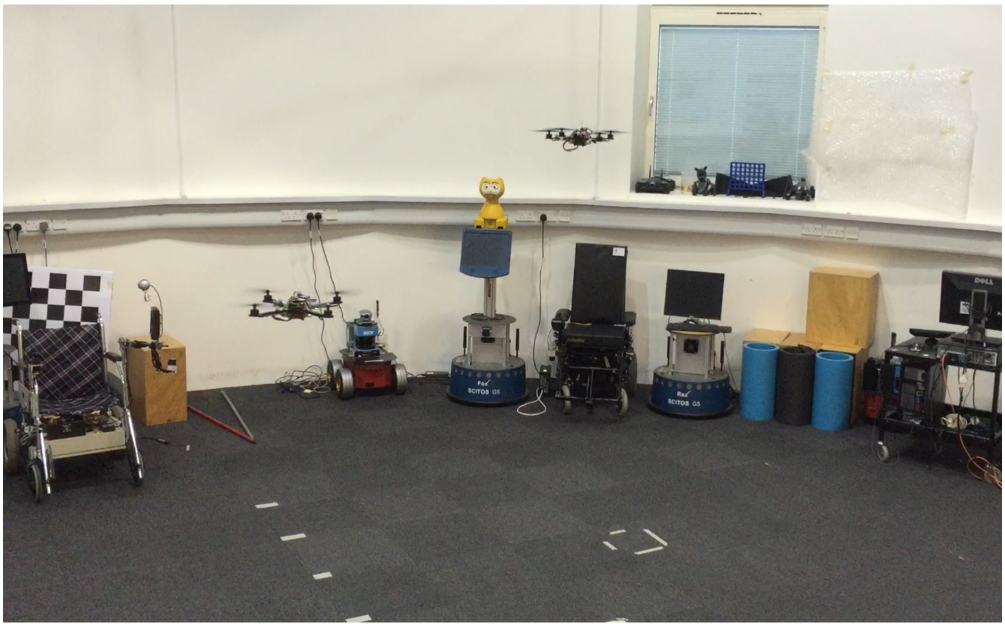

Figure 4.24: Two Quadrotors in Real Leader-Follower Formation

## 4.9 Discussions

This chapter has presented the performance of applying the $H_\infty$ controller to the leader-follower formation control problem of quadrotors. The effect of the external disturbance and the model parameter uncertainties were considered. The controller stability and robustness were analysed and a set of corresponding conditions were given. The controller was tested in the MATLAB simulater. The simulation results show that the proposed $H_\infty$ controller achieved excellent performance. The proposed $H_\infty$ controller was tested on the vehicles via several flight scenarios with external disturbance and mass change consideration. The experimental results verified the robustness and the stability of the proposed controller.

It is important to state two issues in the use of two quadrotors in leader-follower formation control in the real tests. The first issue is that they require a high speed Vicon System software. At the start, the Vicon System software was very slow, and controlling the follower failed several times as it calculated the position of the leader and sent it to the computer, then from the computer to the follower. A good solution to solve this problem was the tracking software, which was fast enough. The second issue is that using one programming software was better than combining two of them. The Hummingbird quadrotor could be controlled using MATLAB and C Language. The MATLAB code was slow in calculations compared with the C code and it was difficult to combine these two codes. Therefore, in this work the MATLAB code was tried to control the vehicle but it was difficult to use, then the C code was used.

# Chapter 5

# Integral Backstepping Controller

## 5.1 Introduction

In this chapter, two controllers are investigated for stabilisation, path tracking and leader-follower team formation and are compared with the $H_\infty$ controller. The first controller is a $PD^2$ obtained in [31] and implemented for attitude stability. The second controller is an IBS control algorithm presented for the path tracking and leader-follower team formation problems of quadrotors. This nonlinear control technique divide the control into two loops, the inner loop is for the attitude stabilisation and the outer loop is for the position control as shown in Figure 5.1. In this case, the leader and the follower desired quaternions are assumed to be $q_{L1d} = q_{L2d} = 0$ and $q_{F1d} = q_{L1}$ and $q_{F2d} = q_{L2}$.

The dynamic model of a quadrotor is represented based once on unit quaternion and another on Euler angles representation and includes some modelled aerodynamical effects as a nonlinear part. The IBS controller is designed for the translational part to track the desired trajectory. Stability analysis is achieved via a suitable Lyapunov function. The external disturbance and model parameters uncertainty are considered in the simulation tests to be compared with the $H_\infty$ controller results in all circumstances.

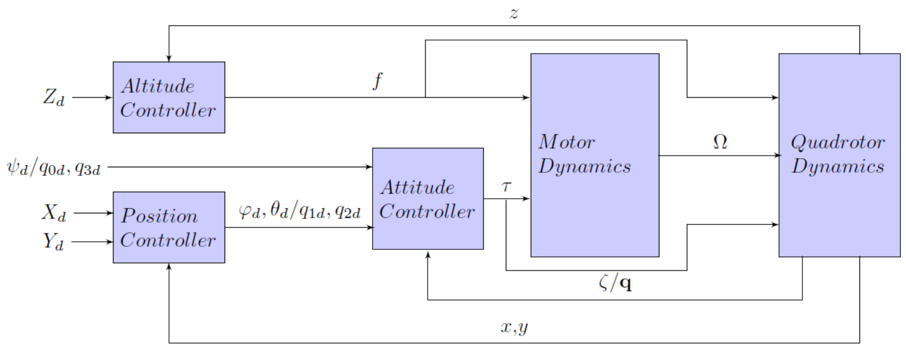

FIGURE 5.1: Two-Loop Control Block Diagram

## 5.2 Integral Backstepping Technique

Integral backstepping control is one of the popular control approaches for both individual and multiple quadrotors. In this section, the integral backstepping control is applied for the individual quadrotor path tracking and leader-follower formation problems. Its main result is also used later in simulation for evaluating the robustness of $H_\infty$ controllers.

### 5.2.1 Backstepping Control Concept

Backstepping is a recursive design mechanism to asymptotically stabilise a controller for the following system [123]:

$$\begin{cases} \dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathit{\Gamma} \\ \dot{\mathit{\Gamma}} = \mathbf{u} \end{cases} . \tag{5.1}$$

This system is described as an initial system in Figure 5.2, where $\mathbf{x} \in \mathbb{R}^n$ and $\mathit{\Gamma} \in \mathbb{R}$ are the system state and $\mathbf{u} \in \mathbb{R}$ is the control input. $f, g : \mathcal{D} \to \mathbb{R}^n$ are assumed to be smooth and $f(0) = 0$. A stabilising state feedback control law $\mathit{\Gamma} = \Phi(\mathbf{x})$, assuming $\Phi(0) = 0$, exists, in addition to a Lyapunov function

$V_1 : \mathcal{D} \to \mathbb{R}^+$ such that

$$\dot{V}_1(\mathbf{x}) = \frac{\partial V_1}{\partial \mathbf{x}} \left[ f(\mathbf{x}) + g(\mathbf{x})\Phi(\mathbf{x}) \right] \leq -V_\varepsilon(\mathbf{x}), \forall \mathbf{x} \in \mathcal{D}$$

where $V_\varepsilon(\mathbf{x}) : \mathcal{D} \to \mathbb{R}^+$ is a positive semidefinite function. Now, the following algebraic manipulation is required: by adding and subtracting the term $g(\mathbf{x})\Phi(\mathbf{x})$ to/from the subsystem (5.1) we can have the following system:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\Phi(\mathbf{x}) + g(\mathbf{x})s \tag{5.2}$$

where $s = \mathit{\Gamma} - \Phi(\mathbf{x})$, by this construction, when $s \to 0$, $\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\Phi(\mathbf{x})$ which is asymptotically stable. The derivative of $s$ is

$$\dot{s} = \dot{\mathit{\Gamma}} - \dot{\Phi}(\mathbf{x}) = \mathbf{u} - \dot{\Phi}(\mathbf{x}) = \upsilon \tag{5.3}$$

which is the backstepping, since $\Phi(\mathbf{x})$ is stepped back by differentiation as described in Figure 5.3. So we have

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x})\Phi(\mathbf{x}) + g(\mathbf{x})s \\ \dot{s} &= \upsilon \end{aligned} \tag{5.4}$$

This system is equivalent to the initial system (5.1), where $\dot{\Phi} = \frac{\partial \Phi}{\partial \mathbf{x}}\dot{\mathbf{x}} = \frac{\partial \Phi}{\partial \mathbf{x}} \left[ f(\mathbf{x}) + g(\mathbf{x})\mathit{\Gamma} \right]$. The next step is to stabilise the system (5.4),and the following Lyapunov function is considered:

$$V(\mathbf{x}, s) = V_1(\mathbf{x}) + \frac{1}{2}s^2.$$

Then

$$\begin{aligned} \dot{V} &= \frac{\partial V_1}{\partial \mathbf{x}} \left[ f(\mathbf{x}) + g(\mathbf{x})\Phi(\mathbf{x}) + g(\mathbf{x})s \right] + s\upsilon \\ &\leq -V_\epsilon(\mathbf{x}) + \left[ \frac{\partial V_1}{\partial \mathbf{x}} g(\mathbf{x}) + \upsilon \right] s. \end{aligned}$$

Let

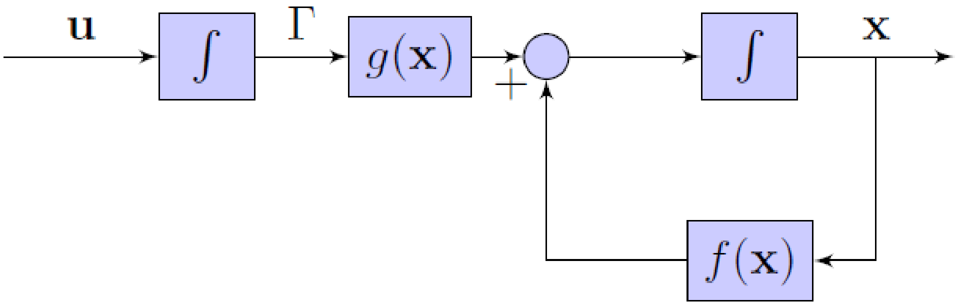

FIGURE 5.2: Initial System

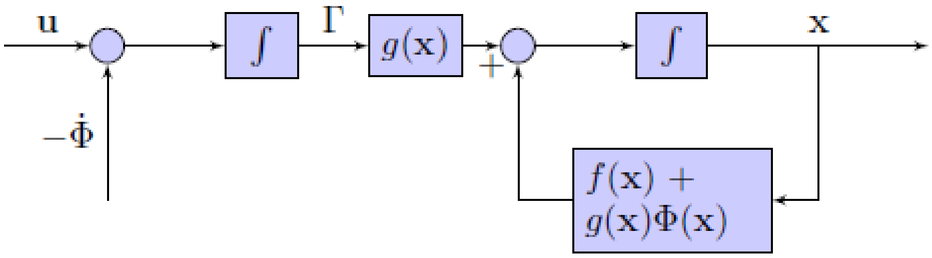

FIGURE 5.3: Backstepping System

$$\begin{cases} \upsilon = -\frac{\partial V_1}{\partial \mathbf{x}} g(\mathbf{x}) - \varepsilon s \\ \varepsilon > 0 \end{cases} .$$

Then

$$\dot{V} \leq -V_\epsilon(\mathbf{x}) - \varepsilon s^2 < 0. \tag{5.5}$$

This signifies that the origin $(\mathbf{x} = 0, s = 0)$ is asymptotically stable. Since $\Phi(0) = 0$, then the origin $\mathbf{x} = 0$ and $\mathit{\Gamma} = 0$ is also asymptotically stable. In the next step an integral part is added to the BS controller to eliminate the steady state error which occurred in the simulation results and is called IBS.

### 5.2.2 Follower integral backstepping controller

The IBS controller for the follower is to track the leader and maintain a desired distance between them with desired incidence and bearing angles.

#### 5.2.2.1 Based on Quaternion Representation

We start with the follower's translational part, which can be rewritten from the dynamic model (4.1) as:

$$\ddot{\mathbf{p}}_F = f(\mathbf{p}_F) + g(\mathbf{p}_F) f_F \tag{5.6}$$

where

$$f(\mathbf{p}_F) = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^T$$

$$g(\mathbf{p}_F) = \begin{bmatrix} u_{Fx}/m_F \\ u_{Fy}/m_F \\ ({q_{F0}}^2 - {q_{F1}}^2 - {q_{F2}}^2 + {q_{F3}}^2)/m_F \end{bmatrix}$$

with

$$\begin{cases} u_{Fx} = 2\left(q_{F1}q_{F3} + q_{F0}q_{F2}\right) \\ u_{Fy} = 2\left(q_{F2}q_{F3} - q_{F0}q_{F1}\right) \end{cases} .$$

Then the position tracking error between the leader and the follower can be calculated as:

$$\tilde{\mathbf{p}}_F = \mathbf{p}_{Fd} - \mathbf{p}_F = \mathbf{p}_L - R_L^T d \begin{bmatrix} \cos\rho\cos\sigma \\ \cos\rho\sin\sigma \\ \sin\rho \end{bmatrix} - \mathbf{p}_F \tag{5.7}$$

and its derivative is

$$\dot{\tilde{\mathbf{p}}}_F = \dot{\mathbf{p}}_{Fd} - \dot{\mathbf{p}}_F = \dot{\mathbf{p}}_{Fd} - \mathbf{v}_F \tag{5.8}$$

where $\mathbf{v}_F$ is a virtual control, and its desirable value can be described as:

$$\mathbf{v}_F^d = \dot{\mathbf{p}}_{Fd} + b_F \tilde{\mathbf{p}}_F + k_F \bar{\mathbf{p}}_F \tag{5.9}$$

where the integration of the follower position error is added to minimise the steady-state error.

Now, consider the linear velocity error between the leader and the follower as:

$$\tilde{\mathbf{v}}_F = \mathbf{v}_F^d - \dot{\mathbf{p}}_F. \tag{5.10}$$

By substituting (5.9) into (5.10) we obtain

$$\tilde{\mathbf{v}}_F = \dot{\mathbf{p}}_{Fd} + b_F\tilde{\mathbf{p}}_F + k_F\bar{\mathbf{p}}_F - \dot{\mathbf{p}}_F \tag{5.11}$$

and its time derivative becomes

$$\dot{\tilde{\mathbf{v}}}_F = \ddot{\mathbf{p}}_{Fd} + b_F\dot{\tilde{\mathbf{p}}}_F + k_F\tilde{\mathbf{p}}_F - \ddot{\mathbf{p}}_F. \tag{5.12}$$

Then from (5.9) and (5.10) we can rewrite (5.8) in terms of the linear velocity error as:

$$\dot{\tilde{\mathbf{p}}}_F = \tilde{\mathbf{v}}_F - b_F\tilde{\mathbf{p}}_F - k_F\bar{\mathbf{p}}_F. \tag{5.13}$$

By substituting (5.6) and (5.13) into (5.12), the time derivative of the linear velocity error can be rewritten as:

$$\dot{\tilde{\mathbf{v}}}_F = \ddot{\mathbf{p}}_{Fd} + b_F\tilde{\mathbf{v}}_F - b_F^2\dot{\tilde{\mathbf{p}}}_F - b_Fk_F\bar{\mathbf{p}}_F + k_F\tilde{\mathbf{p}}_F - f(\mathbf{p}_F) - g(\mathbf{p}_F)f_F. \tag{5.14}$$

The desirable time derivative of the linear velocity error is supposed to be

$$\dot{\tilde{\mathbf{v}}}_F = -c_F\tilde{\mathbf{v}}_F - \tilde{\mathbf{p}}_F. \tag{5.15}$$

Now, the total thrust $f_F$, the longitudinal $u_{Fx}$ and the lateral $u_{Fy}$ motion control can be found by subtracting (5.14) from (5.15) as follows:

$$\begin{aligned} f_F =& (g + \dot{v}_{Lz} + (1 - b_{Fz}^2 + k_{Fz})\tilde{z}_F + (b_{Fz} + c_{Fz})\tilde{v}_{Fz} - b_{Fz}k_{Fz}\bar{z}_F - d(R_{q31}\cos\rho\cos\sigma \\ &+ R_{q32}\cos\rho\sin\sigma + R_{q33}\sin\rho))\frac{m_F}{({q_{F0}}^2 - {q_{F1}}^2 - {q_{F2}}^2 + {q_{F3}}^2)} \end{aligned} \tag{5.16}$$

$$\begin{aligned} u_{Fx} =& (\dot{v}_{Lx} + (1 - b_{Fx}^2 + k_{Fx})\tilde{x}_F + (b_{Fx} + c_{Fx})\tilde{v}_{Fx} - b_{Fx}k_{Fx}\bar{x}_F - d(R_{q11}\cos\rho\cos\sigma \\ &+ R_{q12}\cos\rho\sin\sigma + R_{q13}\sin\rho))\frac{m_F}{f_F} \end{aligned} \tag{5.17}$$

$$\begin{aligned} u_{Fy} =& (\dot{v}_{Ly} + (1 - b_{Fy}^2 + k_{Fy})\tilde{y}_F + (b_{Fy} + c_{Fy})\tilde{v}_{Fy} - b_{Fy}k_{Fy}\bar{y}_F - d(R_{q21}\cos\rho\cos\sigma \\ &+ R_{q22}\cos\rho\sin\sigma + R_{q23}\sin\rho))\frac{m_F}{f_F}. \end{aligned} \tag{5.18}$$

For the attitude stability, the following $H_\infty$ nonlinear controller is used:

$$\tau_{Fq} = K_{Fq}\tilde{\mathbf{q}}_F + K_{F\omega}\tilde{\omega}_F + G(\tilde{\omega}_F).$$

#### 5.2.2.2 Simulations Based on Euler Angles Representation

In this subsection, we derive the IBS controller for the follower when the dynamic model is represented based on Euler angles. Let us recall the follower translational part (5.6):

$$\ddot{\mathbf{p}}_F = f(\mathbf{p}_F) + g(\mathbf{p}_F)f_F \tag{5.19}$$

where

$$f(\mathbf{p}_F) = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^T$$

$$g(\mathbf{p}_F) = \begin{bmatrix} u_{Fx}/m_F & u_{Fy}/m_F & c\theta_F c\varphi_F/m_F \end{bmatrix}^T$$

where

$$u_{Fx} = (c\psi_F s\theta_F c\varphi_F + s\psi_F s\varphi_F)$$

$$u_{Fy} = (s\psi_F s\theta_F c\varphi_F - c\psi_F s\varphi_F).$$

By following the procedure in the above subsection, the following controllers are obtained:

$$\begin{aligned} f_F =&(g + \dot{\upsilon}_{Lz} + (1 - b_{Fz}^2 + k_{Fz})\tilde{z}_F + (b_{Fz} + c_{Fz})\tilde{\upsilon}_{Fz} - b_{Fz}k_{Fz}\bar{z}_F - d(R_{\theta 31}\cos\rho\cos\sigma \\ &+ R_{\theta 32}\cos\rho\sin\sigma + R_{\theta 33}\sin\rho))\frac{m_F}{c\theta_F c\varphi_F} \end{aligned} \tag{5.20}$$

$$\begin{aligned} u_{Fx} =&(\dot{\upsilon}_{Lx} + (1 - b_{Fx}^2 + k_{Fx})\tilde{x}_F + (b_{Fx} + c_{Fx})\tilde{\upsilon}_{Fx} - b_{Fx}k_{Fx}\bar{x}_F - d(R_{\theta 11}\cos\rho\cos\sigma \\ &+ R_{\theta 12}\cos\rho\sin\sigma + R_{\theta 13}\sin\rho))\frac{m_F}{f_F} \end{aligned} \tag{5.21}$$

$$\begin{aligned} u_{Fy} =&(\dot{\upsilon}_{Ly} + (1 - b_{Fy}^2 + k_{Fy})\tilde{y}_F + (b_{Fy} + c_{Fy})\tilde{\upsilon}_{Fy} - b_{Fy}k_{Fy}\bar{y}_F - d(R_{\theta 21}\cos\rho\cos\sigma \\ &+ R_{\theta 22}\cos\rho\sin\sigma + R_{\theta 23}\sin\rho))\frac{m_F}{f_F}. \end{aligned} \tag{5.22}$$

For the attitude stability, the following $H_\infty$ nonlinear controller is used:

$$\tau_{FE} = K_{F\zeta}\tilde{\zeta}_F + K_{F\eta}\tilde{\eta}_F + G(\tilde{\eta}_F).$$

Next, we show the stability of the follower's translational part.

### 5.2.3 Follower Controller Stability Analysis

The following candidate Lyapunov function is chosen for the stability analysis for the follower's translational part with the IBS controller:

$$V = \frac{1}{2}(\tilde{\mathbf{p}}_F^T\tilde{\mathbf{p}}_F + \tilde{\mathbf{v}}_F^T\tilde{\mathbf{v}}_F + k_F\bar{\mathbf{p}}_F^T\bar{\mathbf{p}}_F) \tag{5.23}$$

and its time derivative is

$$\dot{V} = \tilde{\mathbf{p}}_F^T\dot{\tilde{\mathbf{p}}}_F + \tilde{\mathbf{v}}_F^T\dot{\tilde{\mathbf{v}}}_F + k_F\bar{\mathbf{p}}_F^T\dot{\bar{\mathbf{p}}}_F. \tag{5.24}$$

By substituting $\dot{\tilde{\mathbf{p}}}_F = \tilde{\mathbf{p}}_F$ and Equations (5.13) and (5.15) into (5.24), Equation (5.24) becomes

$$\dot{V} = -b_F \tilde{\mathbf{p}}_F^T \tilde{\mathbf{p}}_F - c_F \tilde{\mathbf{v}}_F^T \tilde{\mathbf{v}}_F \leq 0. \tag{5.25}$$

Finally, (5.25) is less than zero provided $b_F$ and $c_F$ are positive diagonal matrices, i.e. $\dot{V} < 0$, $\forall(\tilde{\mathbf{p}}_F, \tilde{\mathbf{v}}_F) \neq 0$ and $\dot{V}(0) = 0$. It can be concluded from the positive definition of $V$ and applying LaSalle theorem that a global asymptotic stability is guaranteed. This leads us to conclude that $\lim_{t\to\infty} \tilde{\mathbf{p}}_F = 0$ and $\lim_{t\to\infty} \tilde{\mathbf{v}}_F = 0$, which meets the position condition of (4.7).

### 5.2.4 Leader integral backstepping controller

The leader is to track a desired trajectory $\mathbf{p}_{Ld}$. Its IBS controller is developed by following the procedure described for the follower quadrotor.

#### 5.2.4.1 Based on Quaternion Representation

The result is that the total force and horizontal position control laws $f_L$, $u_{Lx}$ and $u_{Ly}$ can be written using quaternion dynamic model representation as:

$$\begin{aligned} f_L = (&\ddot{z}_{Ld} + g + (1 - {b_{Lz}}^2 + k_{Lz})\tilde{z}_L + (b_{Lz} + c_{Lz})\tilde{v}_{Lz} - b_{Lz}k_{Lz}\bar{z}_L) \\ &\frac{m_L}{q_{L0}^2 - q_{L1}^2 - q_{L2}^2 + q_{L3}^2} \end{aligned} \tag{5.26}$$

$$u_{Lx} = (\ddot{x}_{Ld} + (1 - {b_{Lx}}^2 + k_{Lx})\tilde{x}_L + (b_{Lx} + c_{Lx})\tilde{v}_{Lx} - b_{Lx}k_{Lx}\bar{x}_L)\frac{m_L}{f_L} \tag{5.27}$$

$$u_{Ly} = (\ddot{y}_{Ld} + (1 - {b_{Ly}}^2 + k_{Ly})\tilde{y}_L + (b_{Ly} + c_{Ly})\tilde{v}_{Ly} - b_{Ly}k_{Ly}\bar{y}_L)\frac{m_L}{f_L} \tag{5.28}$$

and the linear velocity tracking errors are defined as:

$$\begin{cases} \tilde{v}_{Lx} = b_{Lx}\tilde{x}_L + \dot{x}_{Ld} + k_{Lx}\bar{x}_L - \dot{x}_L \\ \tilde{v}_{Ly} = b_{Ly}\tilde{y}_L + \dot{y}_{Ld} + k_{Ly}\bar{y}_L - \dot{y}_L \\ \tilde{v}_{Lz} = b_{Lz}\tilde{z}_L + \dot{z}_{Ld} + k_{Lz}\bar{z}_L - \dot{z}_L \end{cases} .$$

The torque vector applied to the leader quadrotor $\tau_{Lq} \in \mathbb{R}^3$ is designed as:

$$\tau_{Lq} = K_{Lq}\tilde{\mathbf{q}}_L + K_{L\omega}\tilde{\omega}_L + G(\tilde{\omega}_L).$$

#### 5.2.4.2 Based on Euler Angles Representation

The result is that the total force and horizontal position control laws $f_L$, $u_{Lx}$ and $u_{Ly}$ can be written using Euler angles dynamic model representation as:

$$f_L = (\ddot{z}_{Ld} + g + (1 - {b_{Lz}}^2 + k_{Lz})\tilde{z}_L + (b_{Lz} + c_{Lz})\tilde{v}_{Lz} - b_{Lz}k_{Lz}\bar{z}_L)\frac{m_L}{c_{\theta L}c_{\varphi L}} \tag{5.29}$$

$$u_{Lx} = (\ddot{x}_{Ld} + (1 - {b_{Lx}}^2 + k_{Lx})\tilde{x}_L + (b_{Lx} + c_{Lx})\tilde{v}_{Lx} - b_{Lx}k_{Lx}\bar{x}_L)\frac{m_L}{f_L} \tag{5.30}$$

$$u_{Ly} = (\ddot{y}_{Ld} + (1 - {b_{Ly}}^2 + k_{Ly})\tilde{y}_L + (b_{Ly} + c_{Ly})\tilde{v}_{Ly} - b_{Ly}k_{Ly}\bar{y}_L)\frac{m_L}{f_L}. \tag{5.31}$$

The torque vector applied to the leader quadrotor $\tau_{LE} \in \mathbb{R}^3$ is designed as:

$$\tau_{LE} = K_{L\zeta}\tilde{\zeta}_L + K_{L\eta}\tilde{\eta}_L + G(\tilde{\eta}_L).$$

These leader controllers are used for path tracking tests.

# 5.3 Simulations

## 5.3.1 Based on Quaternion Representation

### 5.3.1.1 $PD^2$ Controller for Attitude Stabilisation

To compare the attitude stability results obtained by the $H_\infty$ controller described in Subsection 3.3.1, a $PD^2$ controller (5.32) proposed in [31] was implemented and tested in simulation:

$$\tau = \omega \times J\omega + G(\tilde{\omega}) - (\mu_3 + \mu_2\mu_1)\tilde{\mathbf{q}} - \mu_1 J\dot{\tilde{\mathbf{q}}} - \mu_2\tilde{\omega}. \tag{5.32}$$

Figures 5.4 and 5.5 show the quaternion components and the angular velocities performance using the $PD^2$ controller, respectively. The plot with disturbance effect does not appear in these figures because the system is unstable in this circumstance. It can be seen that the $H_\infty$ controller performance in Figures 3.2 and 3.3 achieved zero steady-state error in less than two seconds and it was able to reject the disturbances and cover the change in model parameter uncertainties; while the $PD^2$ controller performance achieved approximately zero steady-state error after 100 seconds and it was not able to reject the disturbances.

Table 5.1 illustrates the RMSE of the quaternion components using the $PD^2$ controller in the three circumstances. It is clear that the RMSE values of the quaternion parameters obtained by using the $PD^2$ controller in normal conditions and with model parameter uncertainties were high compared with those obtained by using the $H_\infty$ controller. In general, the $H_\infty$ controller yielded a good result compared with that of the $PD^2$ controller in terms of RMSE values, time-consumption, disturbance rejection and model parameter uncertainties change coverage.

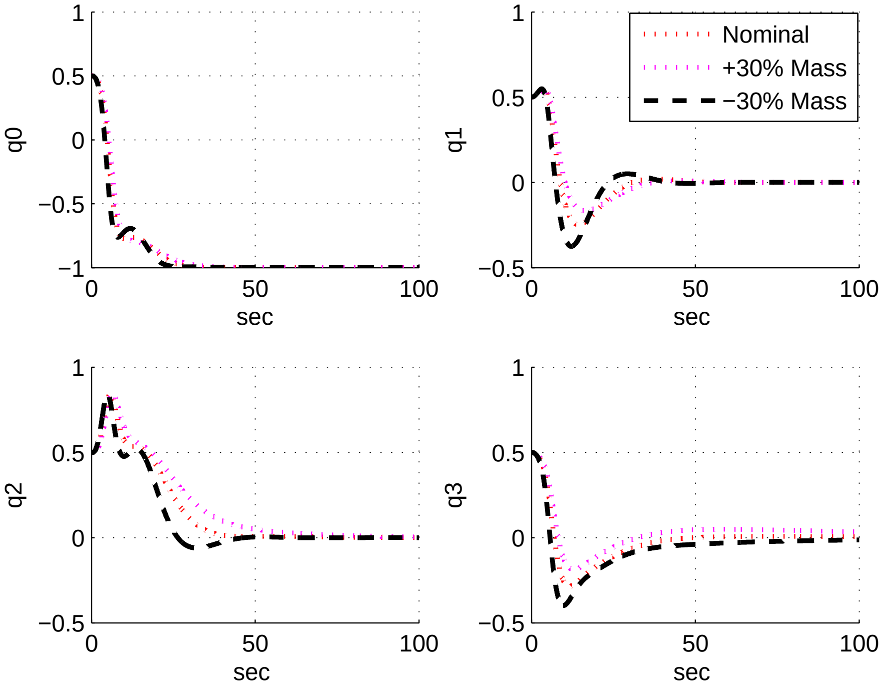

FIGURE 5.4: Quaternion Components under $PD^2$ Controller

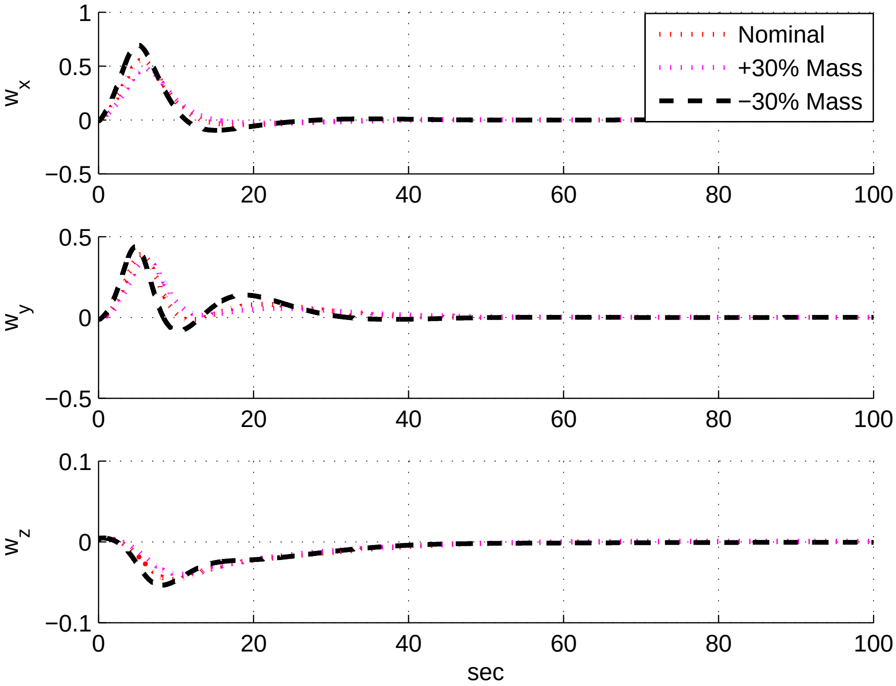

FIGURE 5.5: Angular Velocities under $PD^2$ Controller

| **RMSE** | $q_0$ | $q_1$ | $q_2$ | $q_3$ |
|---|---|---|---|---|
| $PD^2$ | 0.1207 | 0.0226 | 0.0794 | 0.0188 |
| $PD^2 + 30\%$ | 0.1315 | 0.0219 | 0.0934 | 0.0163 |
| $PD^2 - 30\%$ | 0.1128 | 0.0246 | 0.0643 | 0.0240 |

TABLE 5.1: Quaternion Parameter RMSE Values under $PD^2$ Controller Based on Quaternion Representation

#### 5.3.1.2 IBS Controller for Path Tracking

In order to compare the results obtained by using IBS with those of the $H_\infty$ controller, the same two paths tracked by the quadrotor using the $H_\infty$ controller described in Subsection 3.3.2 are tracked by using the IBS controller. For the first path, the parameters of controllers were obtained to be $b = diag(0.7, 0.3, 3)$, $c = diag(0.03, 0.017, 4)$ while $k = diag(0.01, 0.04, 0.03)$, and $b = diag(1, 1, 180)$, $c = diag(0.6, 0.6, 1)$ and $k = diag(0.001, 0.001, 0.001)$ for the second path.

The simulation results of tracking the desired trajectories using the IBS controller are shown in Figures 5.6 - 5.13, which describe the tracking trajectories, positions, quaternions and angular velocities, respectively.

Figures 5.6 and 5.10 illustrate the tracking performance, and the positions of the two paths are shown in Figures 5.7 and 5.11. It can be seen from the figures that the integral backstepping controller was able to lead the quadrotor to track the two desired paths, reject the external disturbance with high errors and oscillations, and recover no more than ±15% model parameter uncertainties.

From Figures 5.8 and 5.12, the quaternions successfully converged to their desired values. And Figures 5.9 and 5.13 show the angular velocities of the quadrotor while tracking the two desired paths. Table 5.2 illustrates the RMSE values of the positions and $q_3$ for the two paths, in which the RMSE of the positions and $q_3$ was almost the same and it was significantly higher than the others with the effect of disturbances. As expected, the controller drove the quadrotor to track the desired trajectories with a steady-state error. These results show the stability of the controller although a bounded external disturbance and model parameter uncertainty were considered.

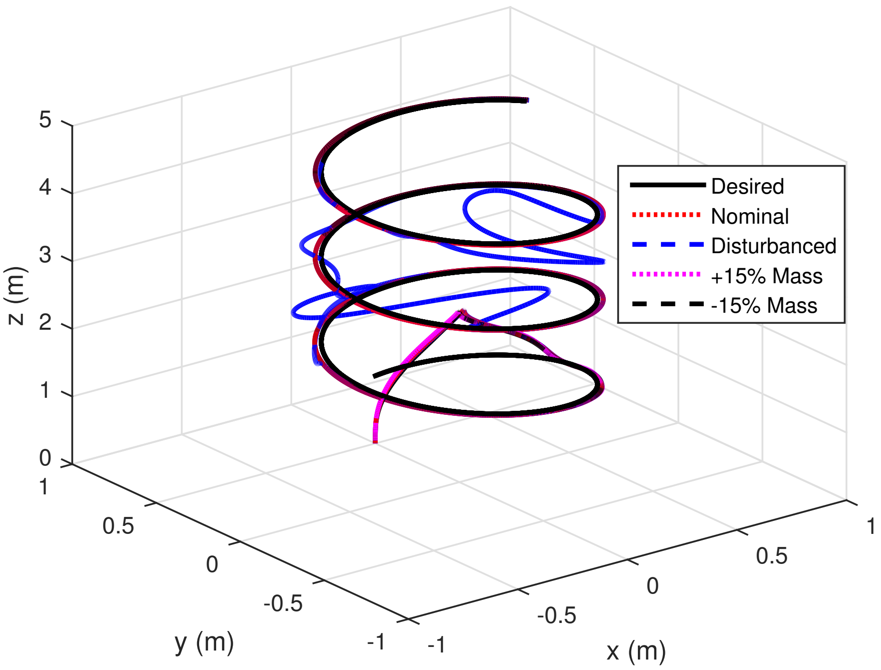

Figure 5.6: First Path Tracking under IBS Controller Based on Quaternion Representation

| | Path 1 | | | | Path 2 | | | |
|---|---|---|---|---|---|---|---|---|
| **RMSE** | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ |
| $IBS$ | 0.0199 | 0.0129 | 0.1276 | 0.0213 | 0.1581 | 0.0219 | 0.1307 | 0.0839 |
| $IBS + \mathbf{d}$ | 0.0672 | 0.0790 | 0.1294 | 0.0673 | 0.2142 | 0.0888 | 0.1311 | 0.0457 |
| $IBS + 15\%$ | 0.0187 | 0.0128 | 0.1274 | 0.0199 | 0.1652 | 0.0180 | 0.1288 | 0.0463 |
| $IBS - 15\%$ | 0.0195 | 0.0126 | 0.1293 | 0.0291 | 0.1602 | 0.0185 | 0.1325 | 0.0517 |

Table 5.2: Position and $q_3$ RMSE Values for the Two Paths under IBS Controller Based on Quaternion Representation

It can be seen that the tracking behaviour of the IBS controller for all parameters of the quadrotor were satisfactory. Additionally, it is obvious that it was able to reject a bounded external disturbance and recover the change of half of the model parameter uncertainties, which indicated the activity of the controller.

#### 5.3.1.3 Team Formation

The IBS controllers were tested in simulation to track a desired path by the leader and maintain the desired distance, desired incidence angle and desired bearing angle between them for the follower. The two desired paths to be tracked

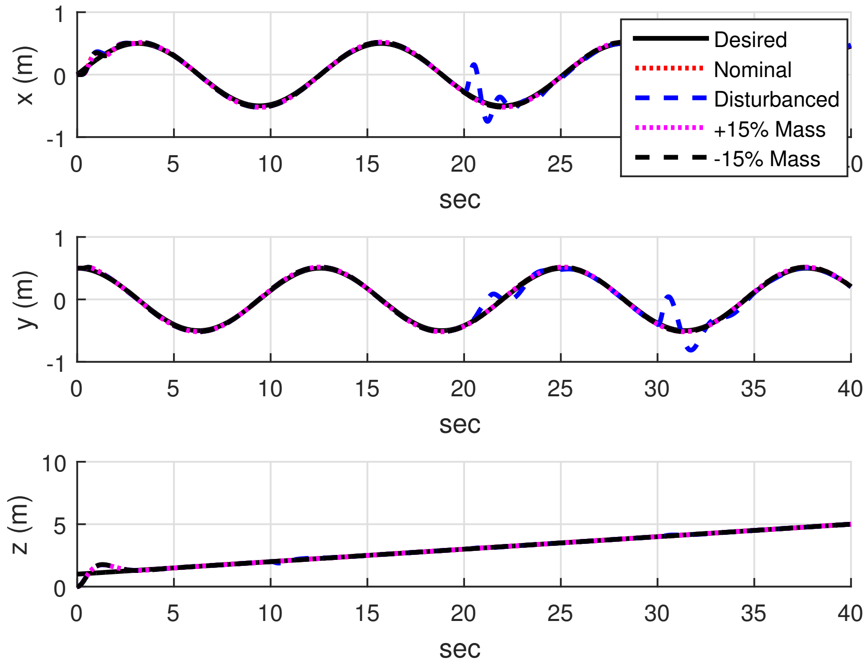

Figure 5.7: First Path Positions under IBS Controller Based on Quaternion Representation

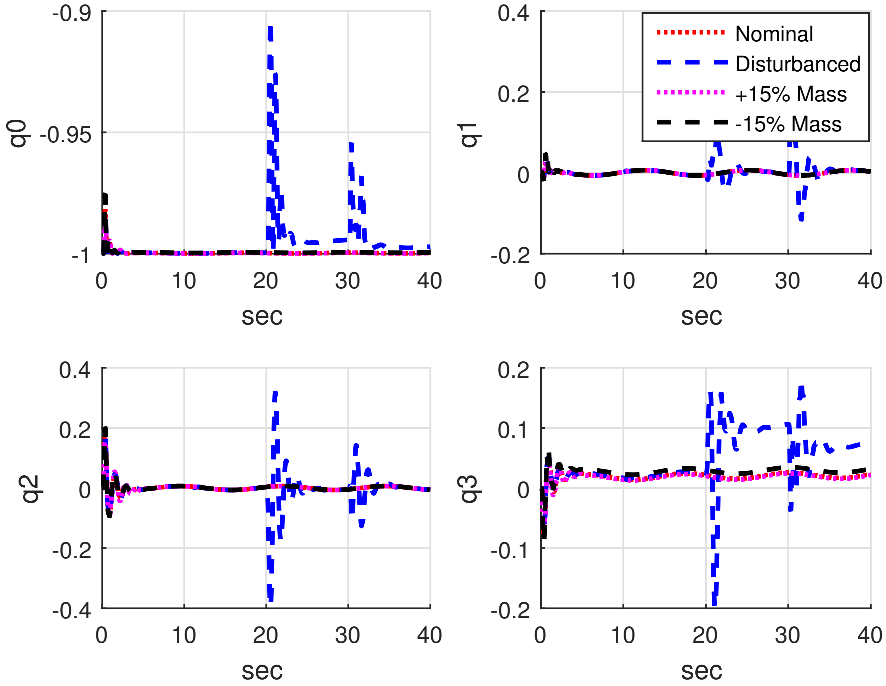

Figure 5.8: First Path Quaternion Components under IBS Controller Based on Quaternion Representation

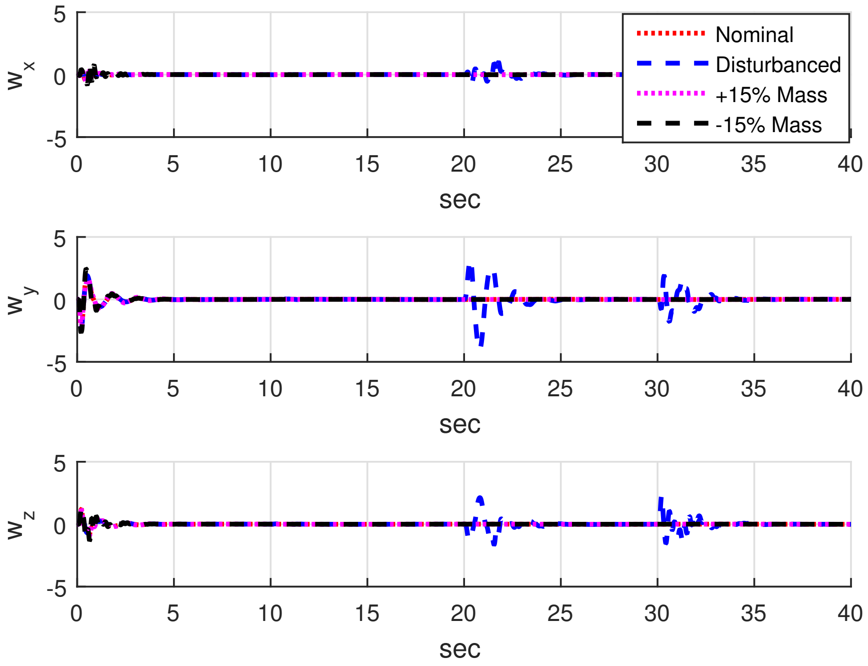

FIGURE 5.9: First Path Angular Velocities under IBS Controller Based on Quaternion Representation

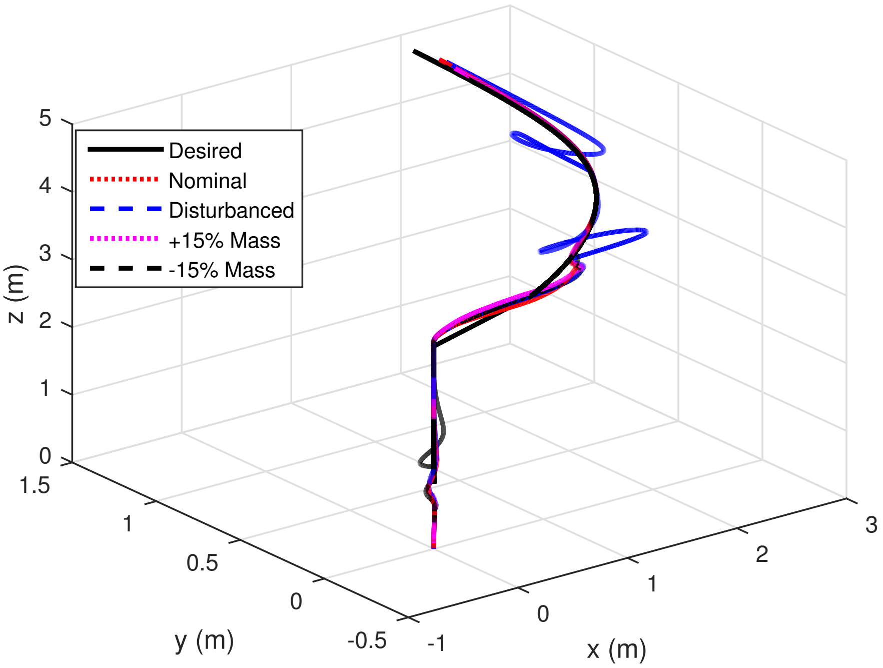

FIGURE 5.10: Second Path Tracking under IBS Controller Based on Quaternion Representation

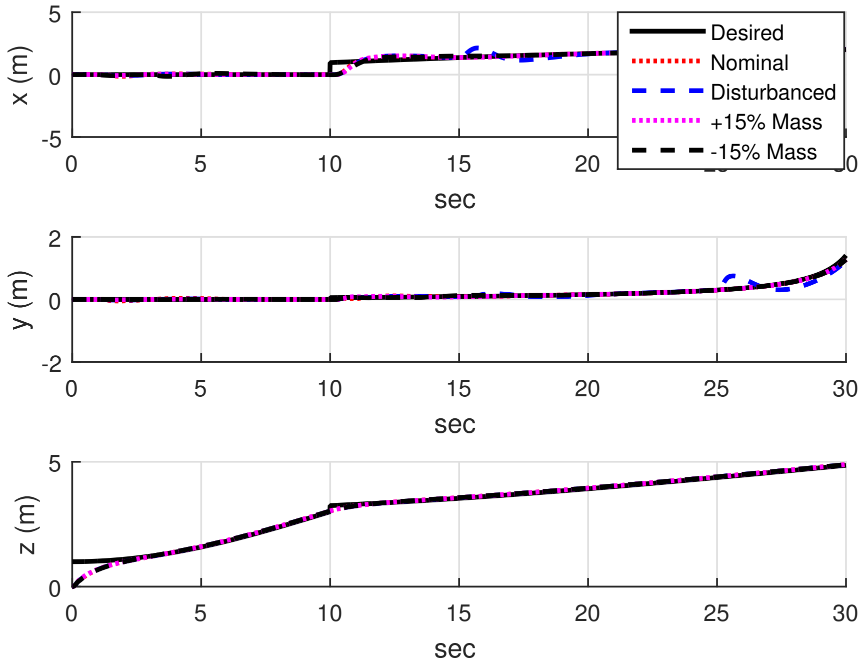

Figure 5.11: Second Path Positions under IBS Controller Based on Quaternion Representation

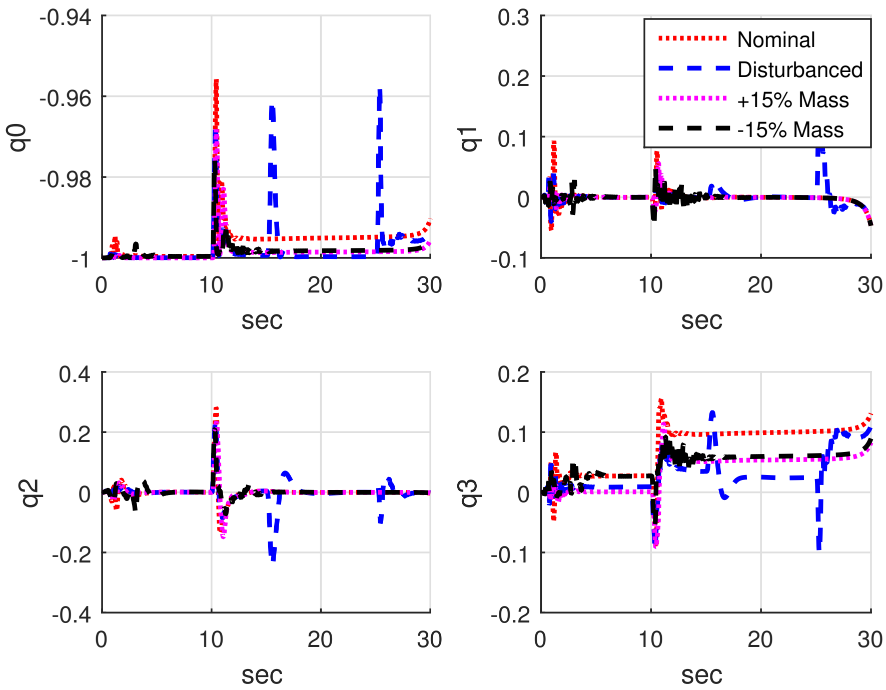

Figure 5.12: Second Path Quaternion Components under IBS Controller Based on Quaternion Representation

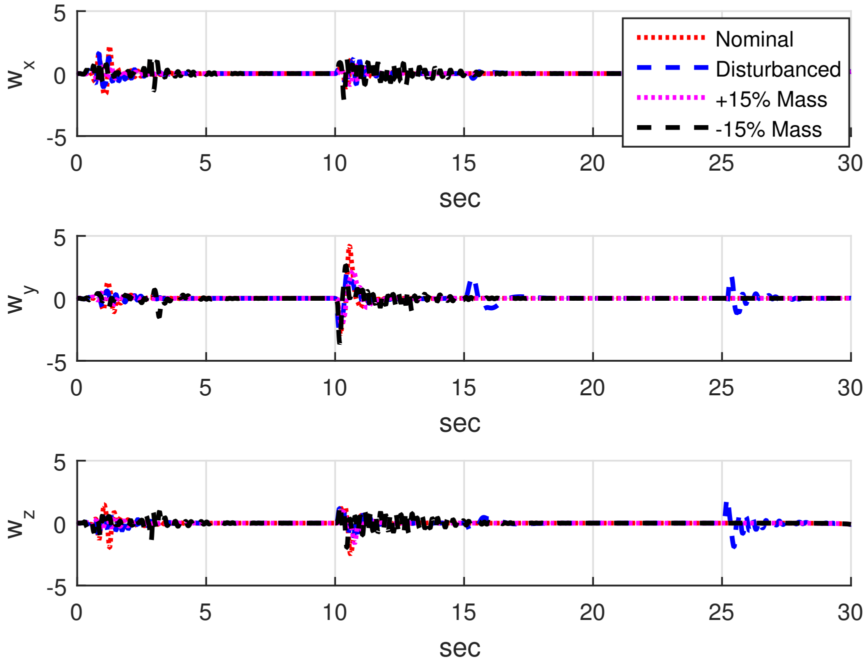

Figure 5.13: Second Path Angular Velocities under IBS Controller Based on Quaternion Representation

were described in Section 4.4. The parameters chosen for both paths were $b_L = diag(180, 0.34, 0.34)$, $c_L = diag(0.7, 0.02, 0.02)$, $k_L = diag(0.0516, 0.0081, 0.0081)$, $b_F = diag(12, 0.7, 0.7)$, $c_F = diag(1.4, 0.02, 0.02)$ and $k_F = diag(0.01, 0.001, 0.001)$.

Figures 5.14 and 5.17 show the formation trajectories of two quadrotors obtained by using the IBS controller when they tracked path 1 and path 2, respectively. From these figures we can see that the IBS controller performed with high error, large oscillation in disturbance rejection and model parameter uncertainty recovery.

The quaternions of the leader and the follower for path 1 and path 2 are shown in Figures 5.15, 5.16, 5.18 and 5.19, respectively. High oscillation is observed in all these figures. The distances between the leader and the follower for two paths are shown in Figure 5.20. Again, high oscillation can be observed from the results of both paths.

Figures 5.21 and 5.22 show the performance of both paths using the IBS controller when only the leader was affected by force disturbance $d_{v_{Lz}} = -4\text{Nm}$ during

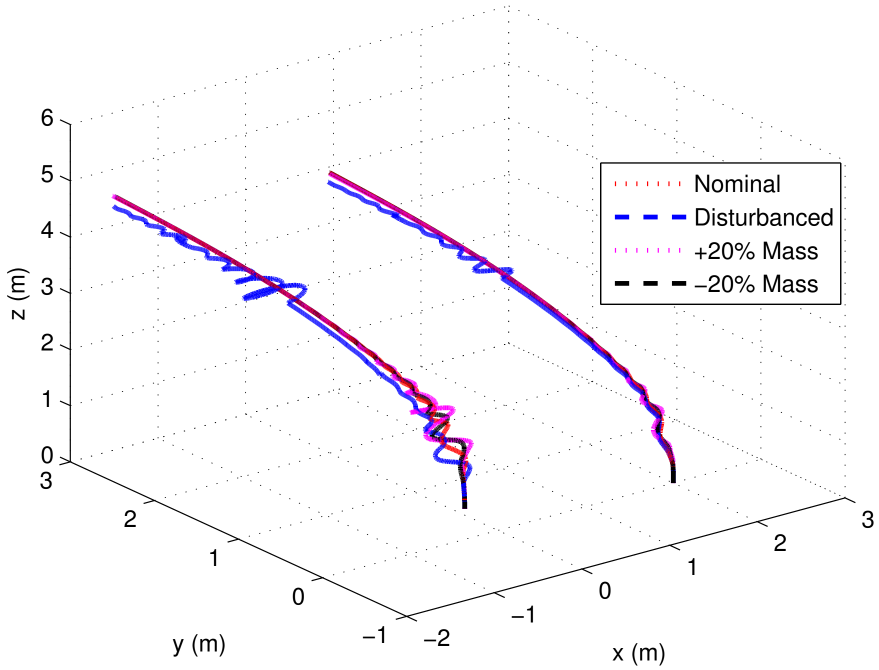

FIGURE 5.14: Leader-Follower Formation in First Path under IBS Controller Based on Quaternion Representation

$10 \leq t \geq 10.25$ seconds, $d_{v_{Lx}} = 4\text{Nm}$ during $20 \leq t \geq 20.25$ seconds, $d_{v_{Ly}} = 4\text{Nm}$ during $30 \leq t \geq 30.25$ seconds, and the attitude part is disturbance using (3.27).

It is clear that the follower tracked the leader and maintained the distance with high error and oscillation in all circumstances.

## 5.3.2 Based on Euler Angles Representation

### 5.3.2.1 Path Tracking

The IBS controller was tested in a MATLAB quadrotor simulator. The two paths described in Subsection 3.5.2 were used here for comparison purposes. The IBS controller constants were chosen to be $b = diag(3, 4, 1)$, $c = diag(18, 3.4, 1)$ and $k = diag(0.03, 1, 0.6)$.

The performance of tracking the first path is illustrated in Figures 5.23 - 5.25, while Figures 5.26 - 5.28 show the tracking performance of the second path.

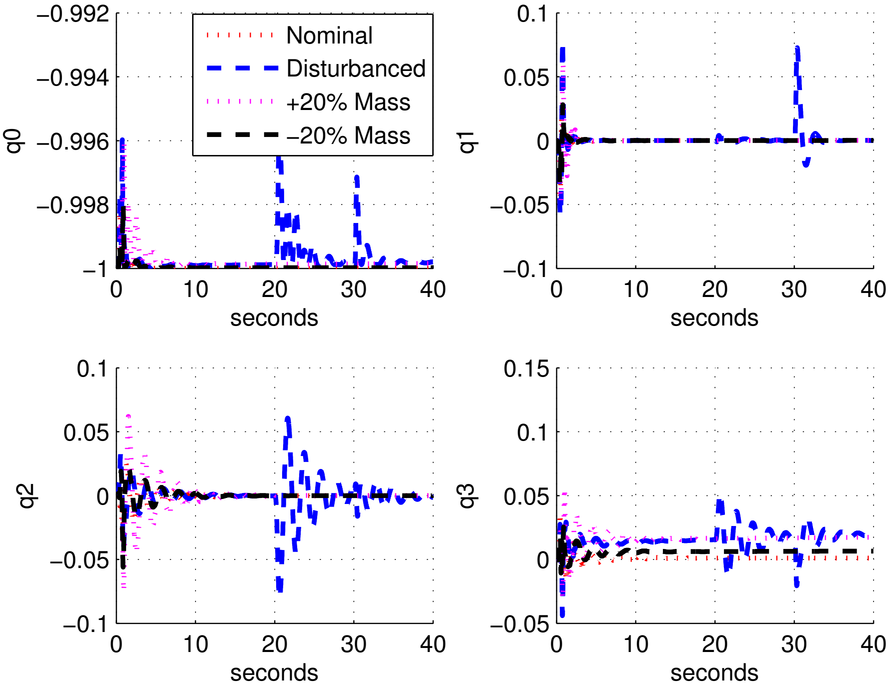

FIGURE 5.15: Leader Quaternions in First Path under IBS Controller Based on Quaternion Representation

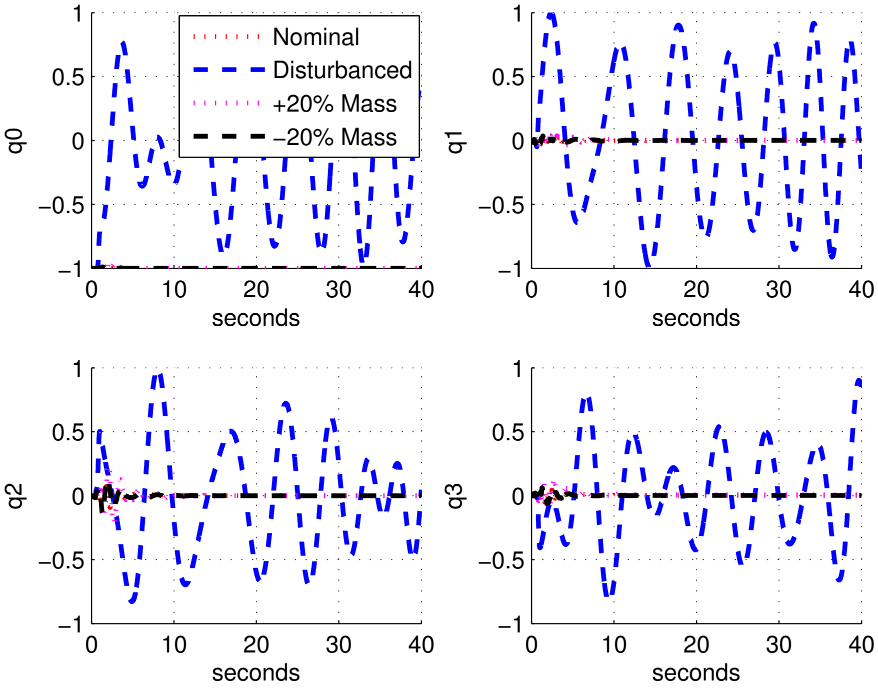

FIGURE 5.16: Follower Quaternions in First Path under IBS Controller Based on Quaternion Representation

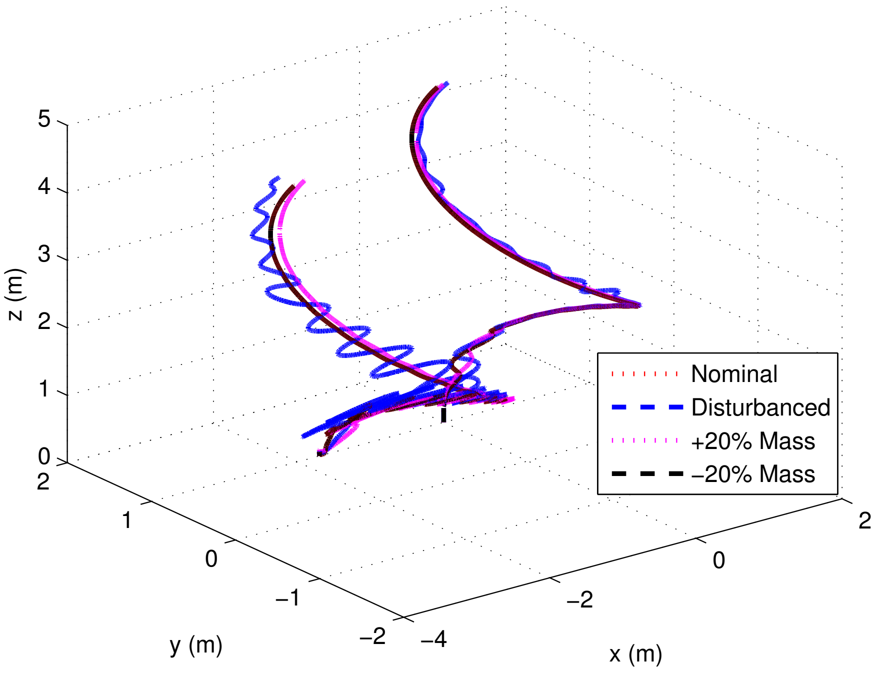

FIGURE 5.17: Leader-Follower Formation in Second Path under IBS Controller Based on Quaternion Representation

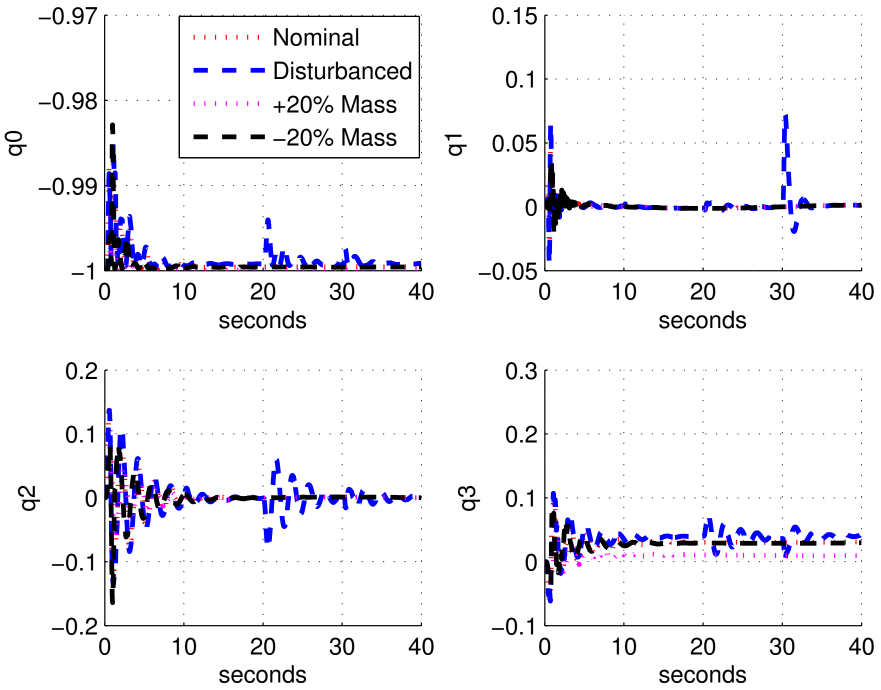

FIGURE 5.18: Leader Quaternions in Second Path under IBS Controller Based on Quaternion Representation

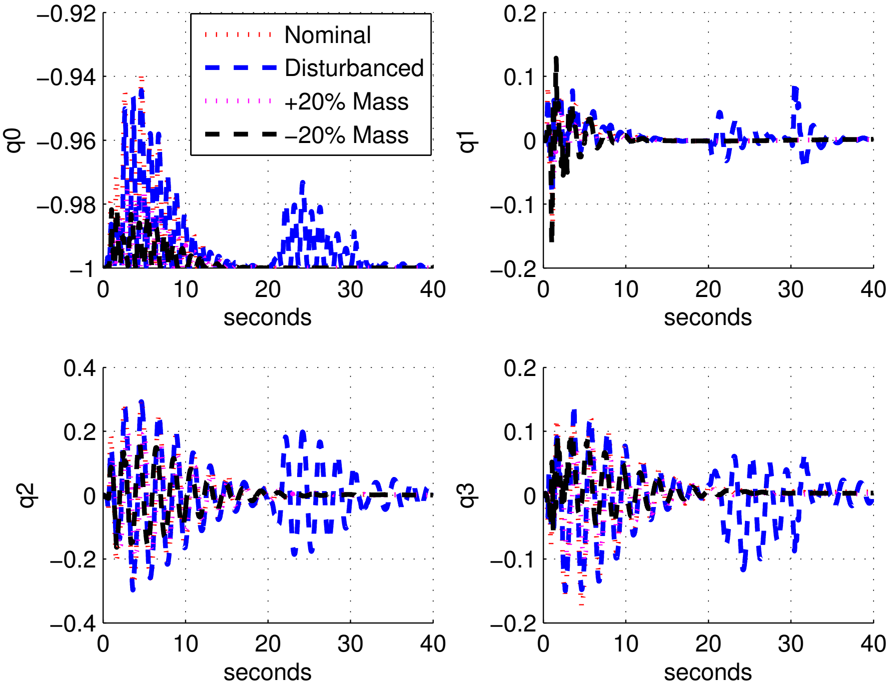

Figure 5.19: Follower Quaternions in Second Path under IBS Controller Based on Quaternion Representation

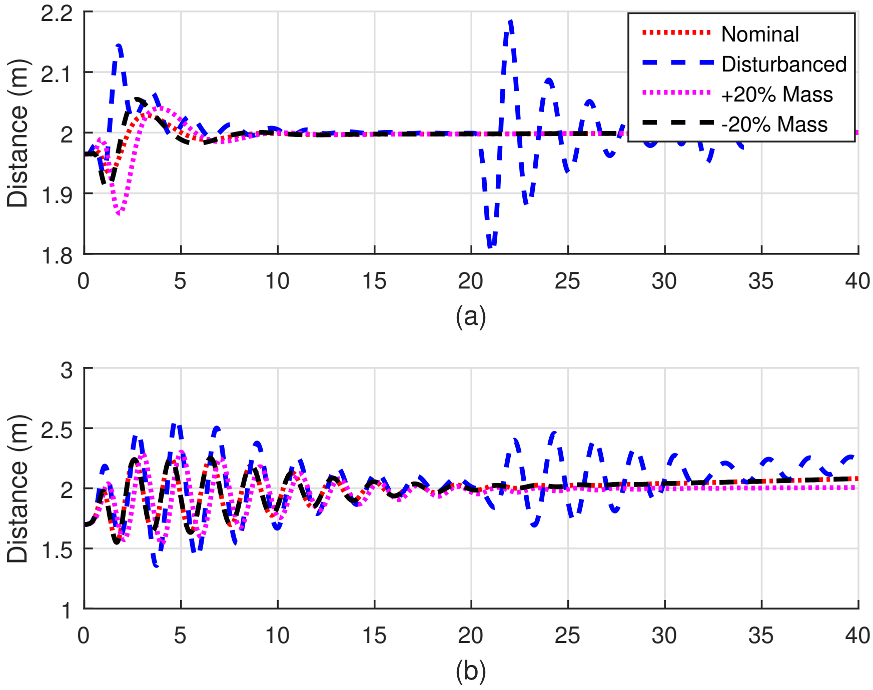

Figure 5.20: The Distance between the Leader and the Follower under IBS Controller Based on Quaternion Representation in, (a) The First Path, (b) The Second Path

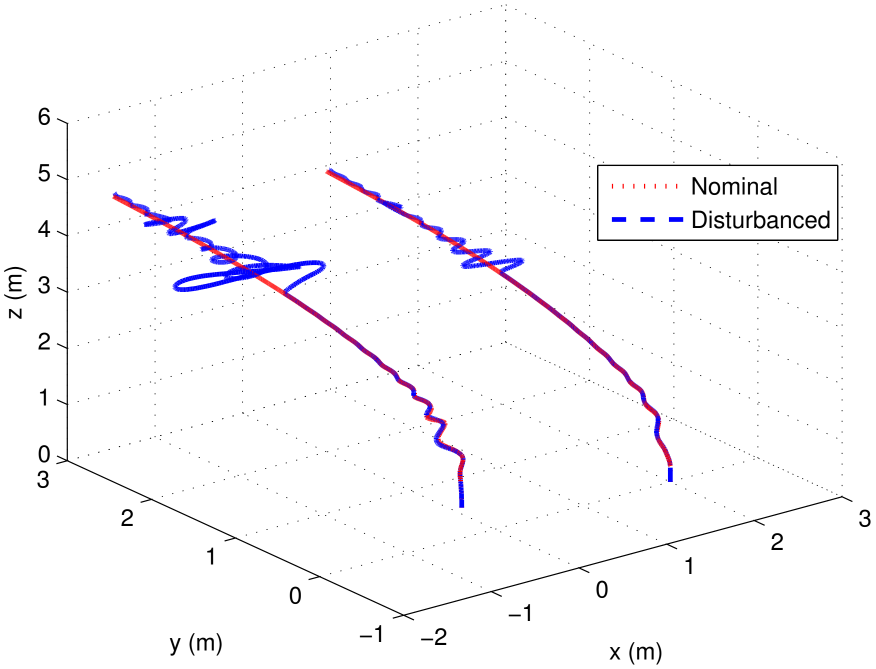

Figure 5.21: Leader-Follower Formation in First Path under IBS Controller Based on Quaternion Representation with Leader Disturbance Only

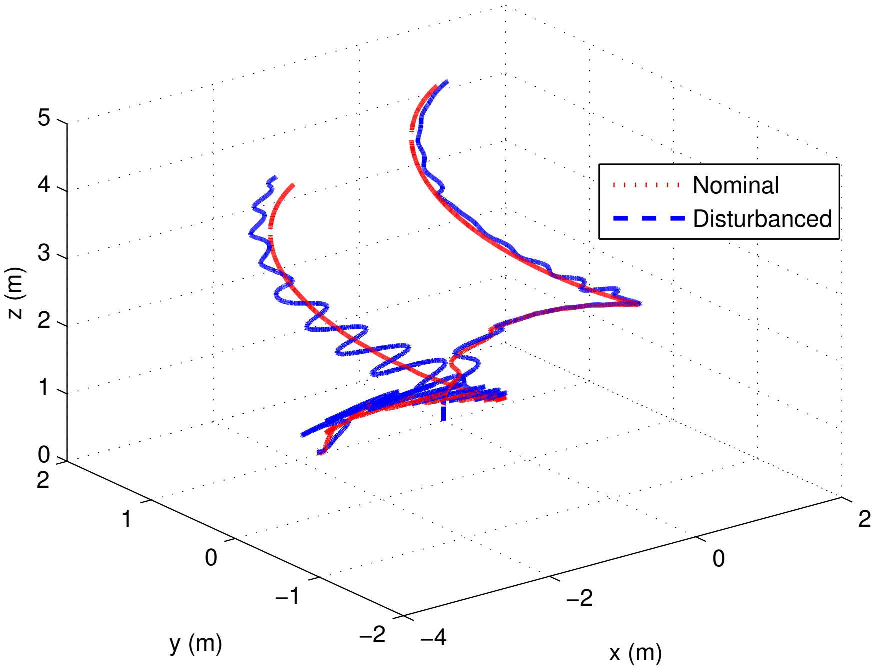

Figure 5.22: Leader-Follower Formation in Second Path under IBS Controller Based on Quaternion Representation with Leader Disturbance Only

| | Path 1 | | | | Path 2 | | | |
|---|---|---|---|---|---|---|---|---|
| **RMSE** | $x(m)$ | $y(m)$ | $z(m)$ | $\psi(deg.)$ | $x(m)$ | $y(m)$ | $z(m)$ | $\psi(deg.)$ |
| $IBS$ | 0.0005 | 0.0040 | 0.0936 | 0.0004 | 0.0030 | 0.0248 | 0.0936 | 0 |
| $IBS + \mathbf{d}$ | 0.0112 | 0.0772 | 0.0943 | 0.0004 | 0.1046 | 1.1018 | 0.3108 | $7e^{-6}$ |
| $IBS+30\%$ | 0.0005 | 0.0040 | 0.0936 | 0.0004 | 0.0030 | 0.0248 | 0.0936 | 0 |
| $IBS-30\%$ | 0.0005 | 0.0040 | 0.0936 | 0.0004 | 0.0030 | 0.0248 | 0.0936 | 0 |

Table 5.3: Position and $\psi$ RMSE Values for the Two Paths under IBS Controller Based on Euler Angles Representation

As shown in these figures, the IBS controller was able to track the desired path with more than 3 seconds and high position tracking errors. In addition, the controller under disturbance was able to track the desired path and recover from the disturbances within more than 4 seconds and long oscillations in the first path but it could not track the desired path in the second path. The controller under +30% and −30% model parameter uncertainty was capable of tracking the desired path with high errors as well.

Table 5.3 demonstrates the RMSE values of the two paths positions and yaw angle. It is clear that the RMSE values of the IBS controller were almost the same when using the IBS controller in normal conditions and with ±30% model parameter uncertainty in both paths, while they significantly increased with the disturbance and were higher in the second path. In general, it can be seen that the IBS controller was able to track the desired trajectories with high position tracking errors and low speed in disturbance rejection.

#### 5.3.2.2 Team Formation

Two paths were presented in the simulation to compare the performance of using the IBS with that of the $H_\infty$ controller with four different circumstances described in Section 4.7.

Figures 5.29 and 5.32 indicate the response of the IBS controller while the leader was tracking the first and second desired path, respectively. Figure 5.35 shows the distance between the leader and the follower via the two paths, and Figures

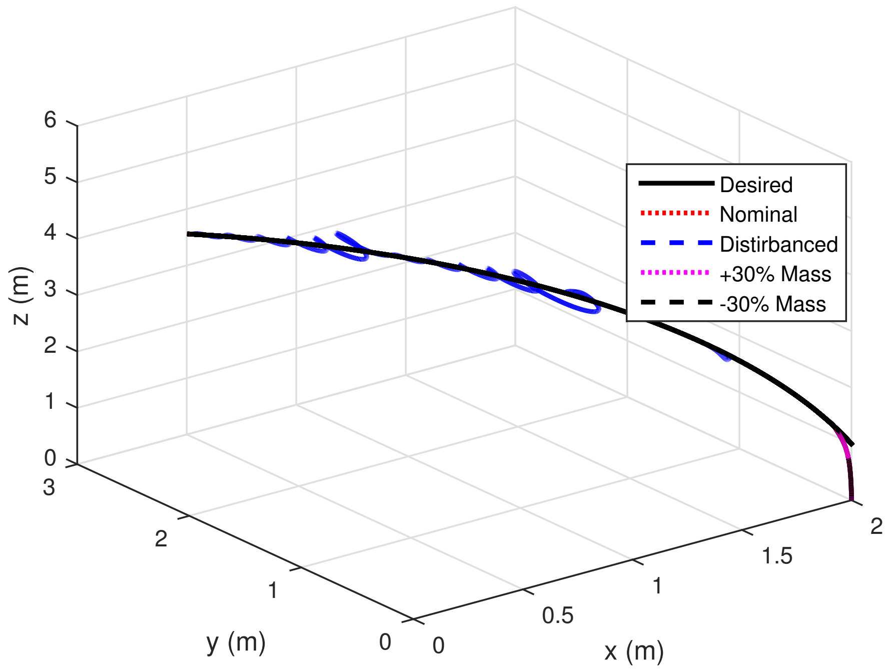

FIGURE 5.23: First Path Tracking under IBS Controller Based on Euler Angles Representation

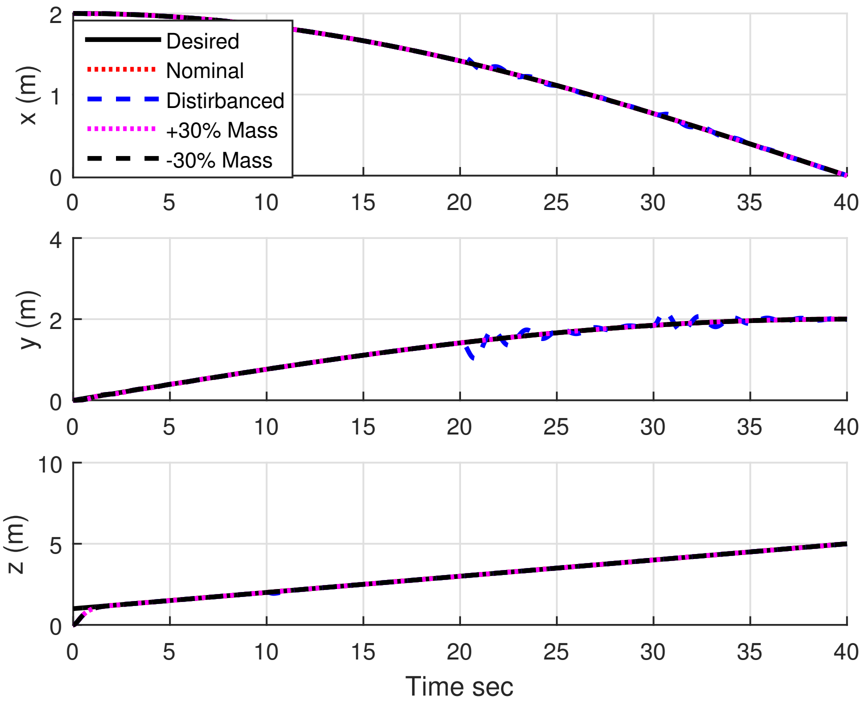

FIGURE 5.24: First Path Position under IBS Controller Based on Euler Angles Representation

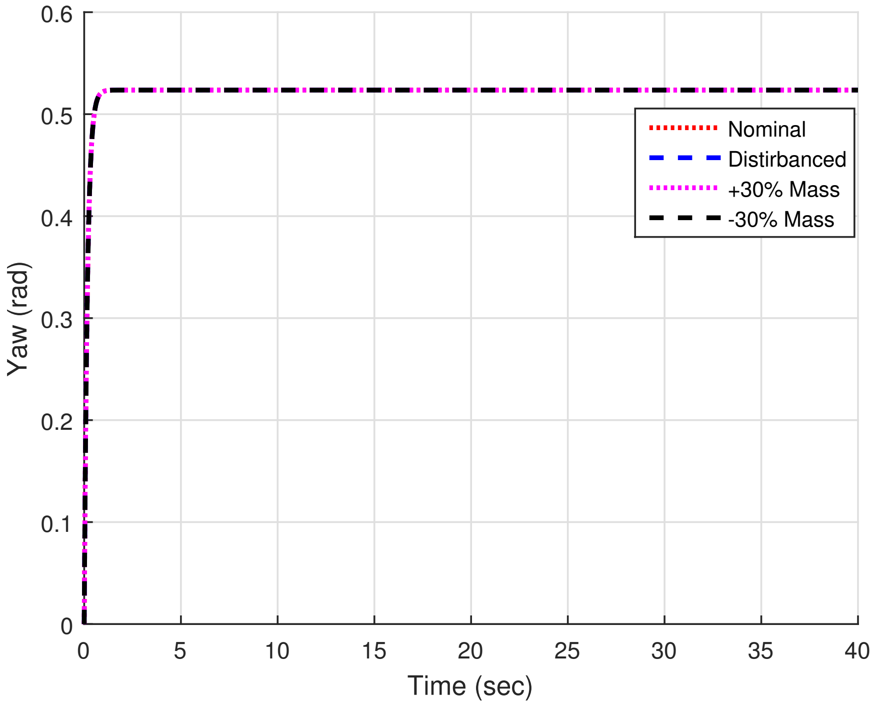

FIGURE 5.25: First Path Yaw Angle under IBS Controller Based on Euler Angles Representation

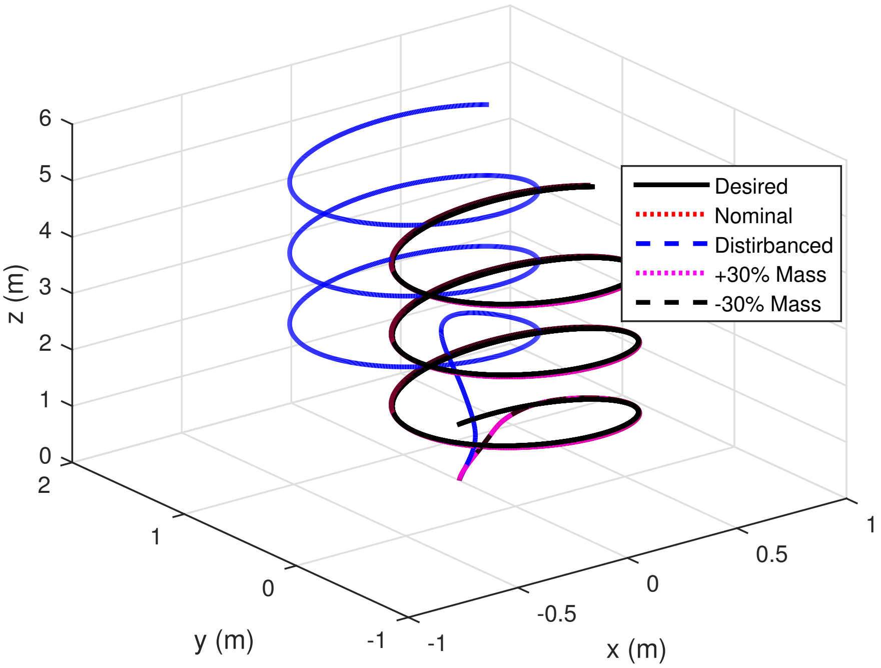

FIGURE 5.26: Second Path under IBS Controller Based on Euler Angles Representation

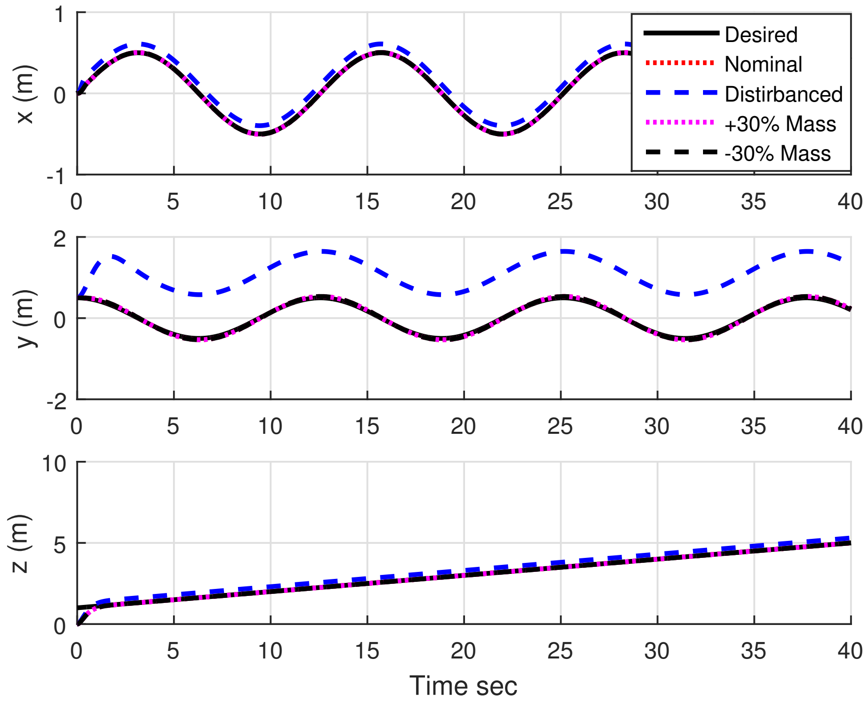

FIGURE 5.27: Second Path Position under IBS Controller Based on Euler Angles Representation

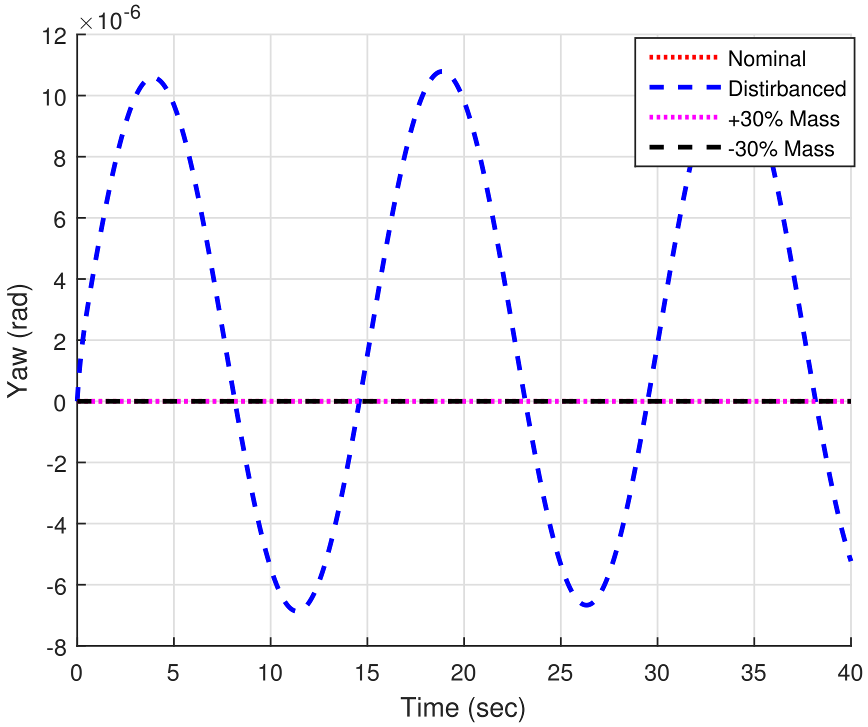

FIGURE 5.28: Second Path Yaw Angle under IBS Controller Based on Euler Angles Representation

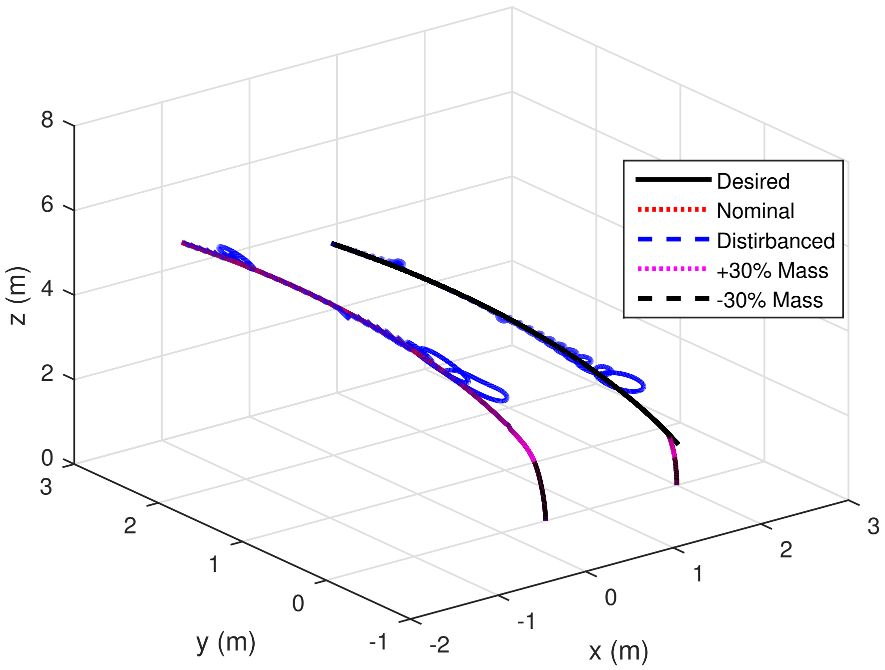

Figure 5.29: Leader-Follower Formation in First Path under IBS Controller Based on Euler Angles Representation

5.30, 5.31, 5.33 and 5.34 illustrate the yaw angles' behaviour for the leader and the follower via the two paths respectively.

It can be noticed from these figures that not only the overshoot but also the error in distance between the leader and the follower was high. It was also slower in rejecting the disturbances in the first path and was not able to reject them in the second path.

## 5.4 Discussions

This section presents a comparison between three different controllers: between $PD^2$ and $H_\infty$ in attitude stabilisation and between IBS and $H_\infty$ in path tracking and leader-follower formation control problems. The effect of the external disturbance and the model parameter uncertainties are also considered.

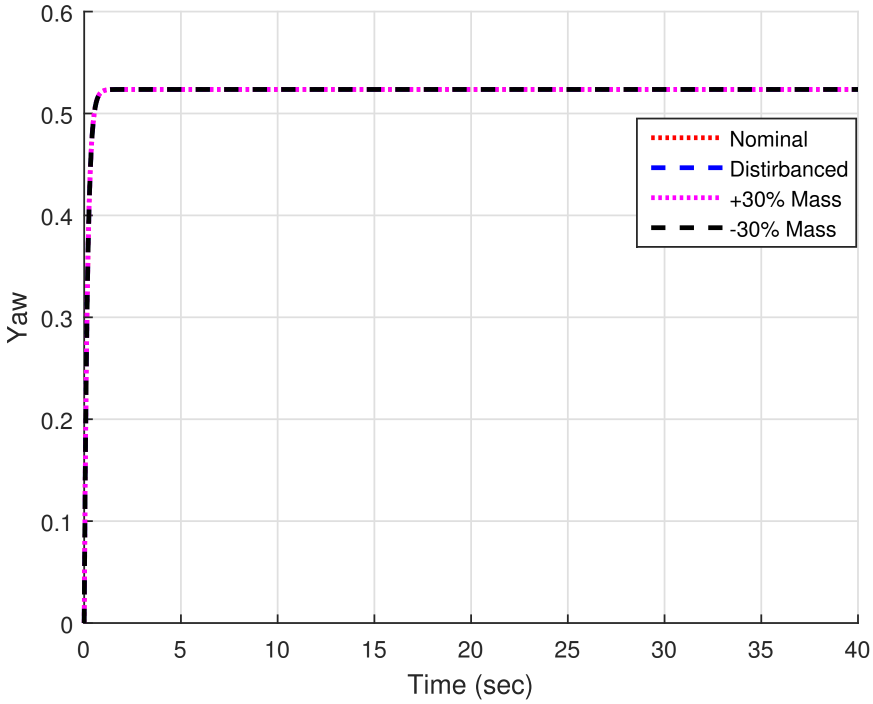

FIGURE 5.30: Leader Yaw Angle in First Path under IBS Controller Based on Euler Angles Representation

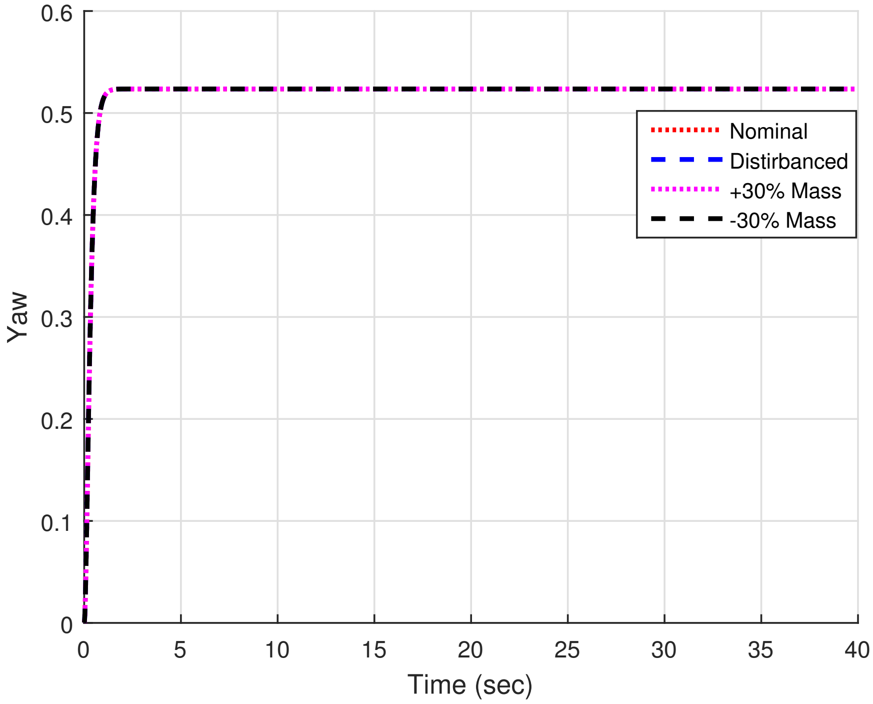

FIGURE 5.31: Follower Yaw Angle in First Path under IBS Controller Based on Euler Angles Representation

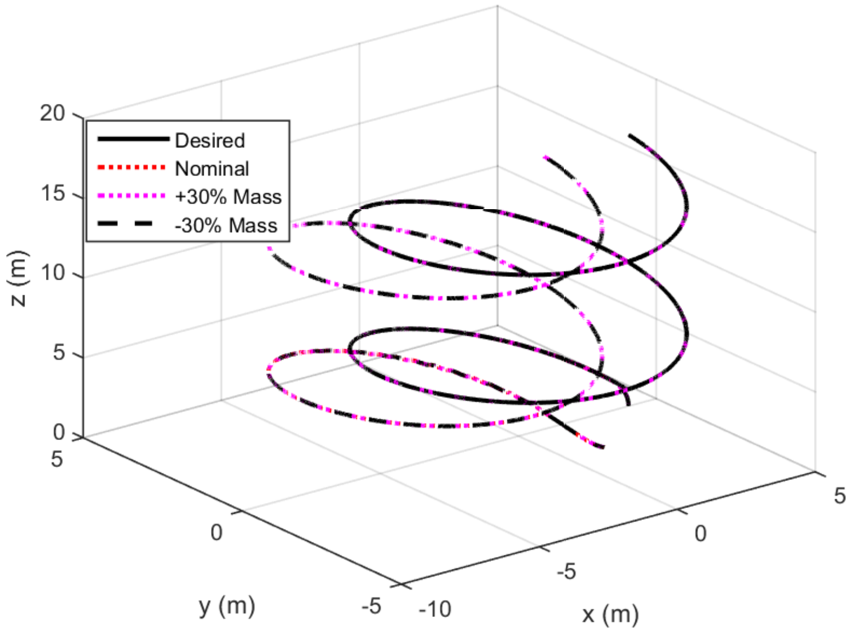

FIGURE 5.32: Leader-Follower Formation in Second Path under IBS Controller Based on Euler Angles Representation

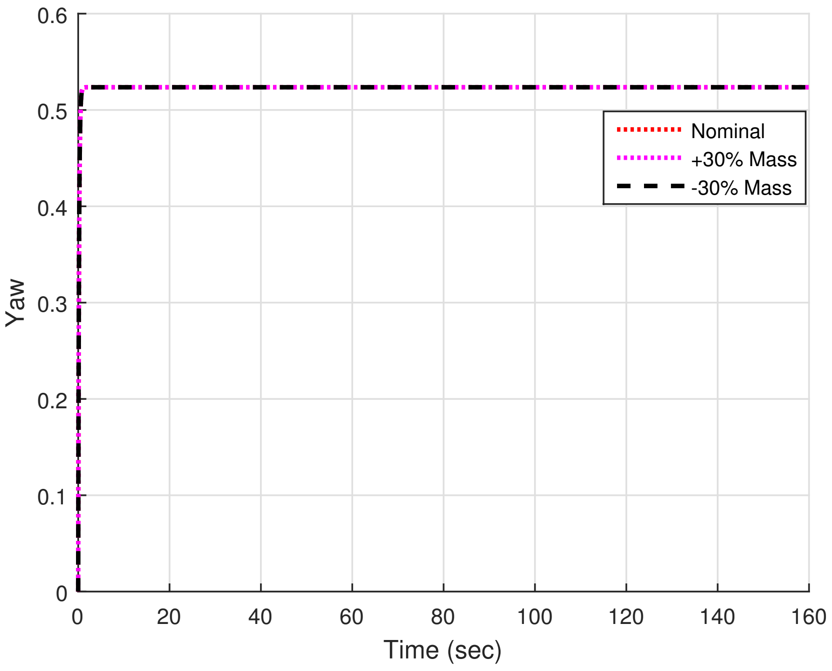

FIGURE 5.33: Leader Yaw Angle in Second Path under IBS Controller Based on Euler Angles Representation

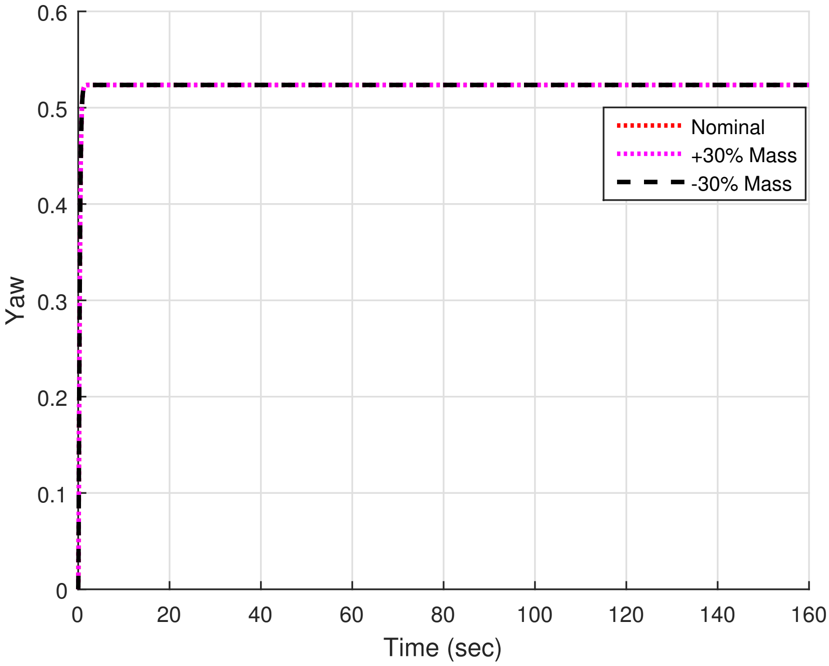

FIGURE 5.34: Follower Yaw Angle in Second Path under IBS Controller Based on Euler Angles Representation

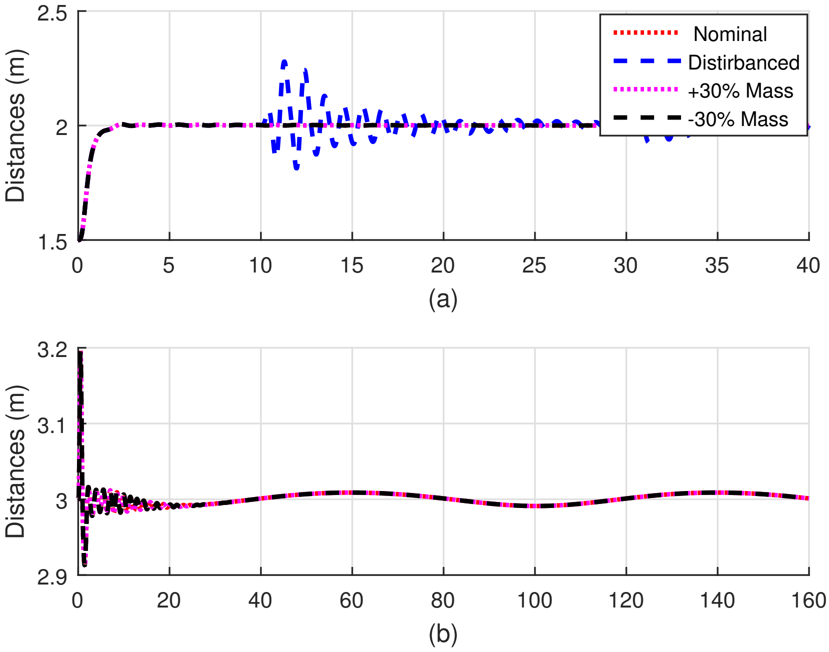

FIGURE 5.35: The Distance between the Leader and the Follower under IBS Controller Based on Euler Angles Representation in (a) The First Path, (b) The Second Path

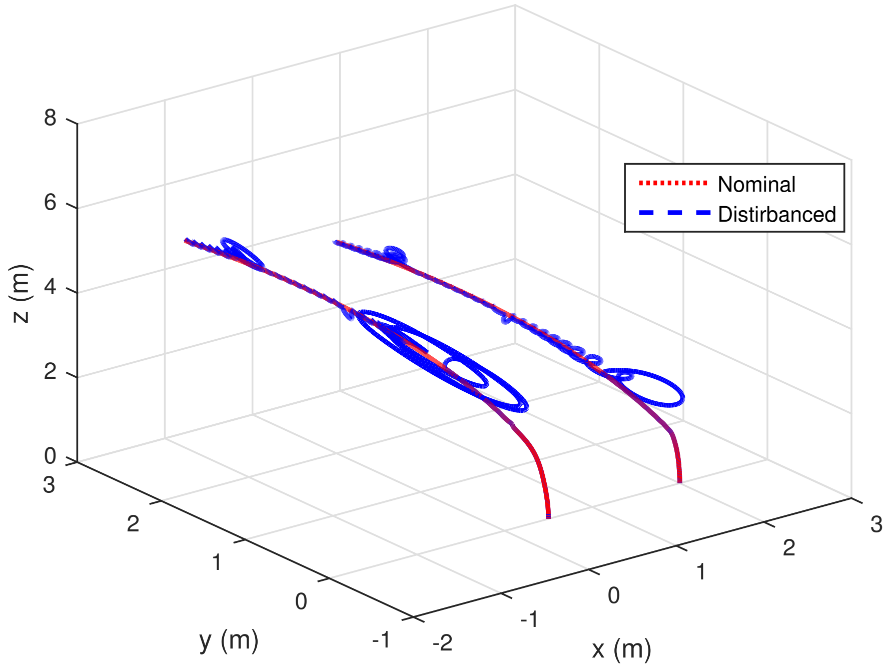

Figure 5.36: Leader-Follower Formation in First Path under IBS Controller Based on Euler Angles Representation with Leader Disturbance Only

The simulation results prove that the performance by using the $H_\infty$ controller had significantly smaller errors than that by using the $PD^2$ and the IBS, see Figures 3.2, 3.3, 3.4, 3.8, 3.12, 3.15, 3.18, 5.4, 5.5, 5.6 and 5.10. It is also obvious that using the $H_\infty$ controller led to a smooth and fast performance with very small overshoot compared with using the IBS, see Figures 4.2, 4.5, 4.11, 4.14, 5.14, 5.17 and 5.21. Moreover, the response of using $H_\infty$ controller in rejecting the external disturbances was faster than that of using the IBS; in fact, the IBS controller was not reject the disturbances when tracking an aggressive paths as in Figure 5.26 and 5.32.

It is well-known that IBS control is a methodical approach to build the Lyapunov function ahead with the control input design. Thus by the cancellation of the indefinite error terms, the stability of the derivative of the Lyapunov function can be secured. Although the stability of the Lyapunov function is guaranteed, this does not guarantee the performance of the system. On the other hand, the $H_\infty$ control technique converts the control problem to a mathematical optimisation problem and solves the latter by finding a suitable controller. The $H_\infty$ controller achieves the stabilisation with performance guaranteed.

In the current study, it can be noticed that adjusting the $H_\infty$ controller parameters was easier than adjusting those of the IBS. The former had two insulated parameters and these were calculated mathematically depending on the attenuation parameter value $\gamma$. In contrast, the latter had three coupling parameters to be tuned manually in order to find the suitable values which usually take longer time and greater effort.

As a result, the proposed $H_\infty$ controller indeed produced better control performance than the other two controllers in all circumstances.

# Chapter 6

# Iterative Linear Quadratic Regulator Controller

## 6.1 Introduction

Amongst the many control techniques, optimal control techniques are widely developed to solve the quadrotor control problem as a main approach of research, due to its successful history. The aim of using the optimal control technique is to calculate an optimal feedback control law to obtain the closed loop optimal solutions. LQR is an optimal control technique that utilises a linear/linearised dynamic model with a quadratic cost index to compute an optimal solution [124].

Based on the LQR technique, an iterative LQR is developed for controlling an individual quadrotor and two quadrotors for the leader-follower formation problem in this chapter. The main idea is that the nonlinear dynamic model is linearised around a nominal predefined path. Then the proposed iLQR technique calculates the optimal control law. Next, the iterative result is applied to the system to track the path. These steps are repeated with each sample of the predefined path.

## 6.2 Nonlinear Dynamic Model

The full quadrotor system has been presented as a translation-rotation dynamic model. We consider that the dynamic model is subjected to specific assumptions illustrated in Appendix A and the full mathematical model based on quaternion representation of (4.1) can be written in the state space form as:

$$\dot{\mathbf{x}}_i = f(\mathbf{x}_i, \mathbf{u}_i) = \begin{cases} \dot{x}_i \\ 2(q_{i1}q_{i3} + q_{i0}q_{i2})\frac{f_i}{m_i} \\ \dot{y}_i \\ 2(q_{i2}q_{i3} - q_{i0}q_{i1})\frac{f_i}{m_i} \\ \dot{z}_i \\ -g + (q_{i0}^2 - q_{i1}^2 - q_{i2}^2 + q_{i3}^2)\frac{f_i}{m_i} \\ \frac{1}{2}(-q_{i1}\omega_{ix} - q_{i2}\omega_{iy} - q_{i3}\omega_{iz}) \\ \frac{1}{2}(q_{i0}\omega_{ix} - q_{i3}\omega_{iy} + q_{i2}\omega_{iz}) \\ \frac{1}{2}(q_{i3}\omega_{ix} + q_{i0}\omega_{iy} - q_{i1}\omega_{iz}) \\ \frac{1}{2}(-q_{i2}\omega_{ix} + q_{i1}\omega_{iy} + q_{i0}\omega_{iz}) \\ \omega_{iy}\omega_{iz}\frac{J_{iy}-J_{iz}}{J_{ix}} - \frac{J_{ir}}{J_{ix}}\omega_{iy}\Omega_i + \frac{1}{J_{ix}}\tau_{iq_1} \\ \omega_{iz}\omega_{ix}\frac{J_{iz}-J_{ix}}{J_{iy}} + \frac{J_{ir}}{J_{iy}}\omega_{ix}\Omega_i + \frac{1}{J_{iy}}\tau_{iq_2} \\ \omega_{ix}\omega_{iy}\frac{J_{ix}-J_{ix}}{J_{iz}} + \frac{1}{J_{iz}}\tau_{iq_3} \end{cases} \tag{6.1}$$

where $\mathbf{x}_i = [x_i, \dot{x}_i, y_i, \dot{y}_i, z_i, \dot{z}_i, q_{i0}, q_{i1}, q_{i2}, q_{i3}, \omega_{ix}, \omega_{iy}, \omega_{iz}]^T$.

## 6.3 iLQR Control

iLQR is one of the optimal control strategies that is formulated to obtain the control signals that minimises a performance criterion to satisfy the physical model constraints. The iLQR strategy is utilised based on the LQR technique to design the full state quadrotor's controller. We will not use the notation $i$ in the coming equations for simplicity. Linearising the nonlinear dynamic model (6.1), we obtain

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) \tag{6.2}$$

with a quadratic cost function of the form

$$\mathbf{J} = \frac{1}{2}(\mathbf{x}_N - \mathbf{x}^*)^T \mathbf{Q}_N (\mathbf{x}_N - \mathbf{x}^*) + \frac{1}{2}\sum_{k=0}^{N-1}(\mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^T \mathbf{R}\mathbf{u}_k). \tag{6.3}$$

The proposed strategy starts with initial control signals $k = 0$, and the linearised nonlinear system around the control signal $\mathbf{u}_k$ and the state $\mathbf{x}_k$ then solves the LQR problem. Then these steps are repeated (iterated) until a good performance is achieved. Let the deviations from $\mathbf{u}_k$ and $\mathbf{x}_k$ be $\delta\mathbf{u}_k$ and $\delta\mathbf{x}_k$, respectively. The linearisation model is

$$\delta\mathbf{x}_{k+1} = A_k\delta\mathbf{x}_k + B_k\delta\mathbf{u}_k \tag{6.4}$$

where the matrices $A_k = \mathbf{J}_\mathbf{x} f(\mathbf{x}_k, \mathbf{u}_k)$ and $B_k = \mathbf{J}_\mathbf{u} f(\mathbf{x}_k, \mathbf{u}_k)$ are denoted by the Jacobians. These are evaluated along $\mathbf{x}_k$ and $\mathbf{u}_k$, respectively. Based on the linear model (6.4), the cost function (6.3) can be written as:

$$\begin{aligned}\mathbf{J} =& \frac{1}{2}(\mathbf{x}_N + \delta\mathbf{x}_N - \mathbf{x}^*)^T \mathbf{Q}_N (\mathbf{x}_N + \delta\mathbf{x}_N - \mathbf{x}^*) + \frac{1}{2}\sum_{k=0}^{N-1}((\mathbf{x}_k + \delta\mathbf{x})^T \mathbf{Q}(\mathbf{x}_k + \delta\mathbf{x}) \\ &+ (\mathbf{u}_k + \delta\mathbf{u})^T \mathbf{R}(\mathbf{u}_k + \delta\mathbf{u})). \end{aligned} \tag{6.5}$$

Adding a constraint to the cost function (6.5), the value function is

$$\begin{aligned}\mathbf{V} =& \frac{1}{2}(\mathbf{x}_N + \delta\mathbf{x}_N - \mathbf{x}^*)^T \mathbf{Q}_N (\mathbf{x}_N + \delta\mathbf{x}_N - \mathbf{x}^*) + \frac{1}{2}\sum_{k=0}^{N-1}((\mathbf{x}_k + \delta\mathbf{x})^T \mathbf{Q}(\mathbf{x}_k + \delta\mathbf{x}) \\ &+ (\mathbf{u}_k + \delta\mathbf{u})^T \mathbf{R}(\mathbf{u}_k + \delta\mathbf{u}) + \delta\lambda_{k+1}^T (A_k\delta\mathbf{x}_k + B_k\delta\mathbf{u}_k - \delta\mathbf{x}_{k+1})) \end{aligned} \tag{6.6}$$

The following Hamiltonian function is a first step to proceed towards the optimal control

$$\begin{aligned} \mathbf{H}_k =&(\mathbf{x}_k + \delta\mathbf{x}_k)^T\mathbf{Q}(\mathbf{x}_k + \delta\mathbf{x}_k) + (\mathbf{u}_k + \delta\mathbf{u}_k)^T\mathbf{R}(\mathbf{u}_k + \delta\mathbf{u}_k) \\ &+ \delta\lambda_{k+1}^T(A_k\delta\mathbf{x}_k + B_k\delta\mathbf{u}_k) \end{aligned} \tag{6.7}$$

and its derivatives with respect to $\delta\mathbf{x}_k$, $\delta\mathbf{u}_k$ and $\delta\mathbf{x}_N$ are

$$\begin{cases} \dfrac{\partial\mathbf{H}_k}{\partial(\delta\mathbf{x}_k)} = \delta\lambda_k \\ \dfrac{\partial\mathbf{H}_k}{\partial(\delta\mathbf{u}_k)} = 0 \\ \dfrac{\partial\mathbf{H}_k}{\partial(\delta\mathbf{x}_N)} = \delta\lambda_N, \end{cases}$$

which leads to the following conditions:

$$\delta\lambda_k = A_k^T\delta\lambda_{k+1} + \mathbf{Q}(\delta\mathbf{x}_k + \mathbf{x}_k) \tag{6.8}$$

$$0 = \mathbf{R}(\mathbf{u}_k + \delta\mathbf{u}_k) + B_k^T\delta\lambda_{k+1} \tag{6.9}$$

$$\delta\lambda_N = \mathbf{Q}_f(\mathbf{x}_N + \delta\mathbf{x}_N - \mathbf{x}^*). \tag{6.10}$$

Based on the boundary condition (6.10), $\delta\lambda_k$ is assumed to be

$$\delta\lambda_k = S_k\delta\mathbf{x}_k + \nu_k \tag{6.11}$$

for some unknown sequences $S_k$ and $\nu_k$. The boundary conditions for $S_k$ and $\nu_k$ are

$$\begin{cases} S_N = \mathbf{Q}_N \\ \nu_N = \mathbf{Q}_N(\mathbf{x}_N - \mathbf{x}_*) \end{cases} \tag{6.12}$$

and from the boundary condition (6.9), $\delta\mathbf{u}_k$ is obtained as:

$$\delta\mathbf{u}_k = -\mathbf{R}^{-1}B_k^T\delta\lambda_{k+1} - \mathbf{u}_k. \tag{6.13}$$

By solving equations (6.4), (6.9) and (6.11), we obtain

$$\delta \mathbf{u}_k = -K\delta \mathbf{x}_k - K_\nu \nu_{k+1} - K_{\mathbf{u}} \mathbf{u}_k \tag{6.14}$$

where

$$K = (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} B_k^T S_{k+1} A_k \tag{6.15}$$

$$K_\nu = (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} B_k^T \tag{6.16}$$

$$K_{\mathbf{u}} = (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} \mathbf{R}. \tag{6.17}$$

Backward recursion equations are used to solve the entire sequences $S_k$ and $\nu_k$ as:

$$S_k = A_k^T S_{k+1}(A_k - B_k K) + \mathbf{Q} \tag{6.18}$$

$$\nu_k = (A_k - B_k K)^T \nu_{k+1} - K^T \mathbf{R} \mathbf{u}_k + \mathbf{Q} \mathbf{x}_k \tag{6.19}$$

where the gains $K$ and $K_{\mathbf{u}}$ are built on the Riccati equation while the gain $K_\nu$ is reliant on auxiliary sequence (6.19). See Appendix C.

The entire sequences of $S_k$ and $\nu_k$ can be solved by the backward recursion (6.18) and (6.19) respectively, with the final state weighting matrix boundary condition $S_N$ stated in the cost function (6.5). The control law 6.14 includes three terms. The gains of the first and the third terms depend on the solution of the Riccati equation, while the second term gain depends on the auxiliary sequence $\nu_k$. In the first term, $\delta \mathbf{x}_k$ represents the error between the actual quadrotor state and the desired state, and in the third term, $\mathbf{u}_k$ represents the nominal control action. Once the modified LQR problem is solved, an improved nominal control sequence can be obtained: $\mathbf{u}_k^* = \mathbf{u}_k + \delta \mathbf{u}_k$, where $\mathbf{u}_k$ is the nominal control and $\mathbf{u}_k^*$ is the improved control. Then the total control laws are concluded as follows:

$$\begin{cases} \delta\mathbf{u}_{ik} = -K_i\delta\mathbf{x}_{ik} - K_{i\nu}\nu_{ik+1} - K_{i\mathbf{u}}\mathbf{u}_{ik} \\ K_i = (B_{ik}^T S_{ik+1} B_{ik} + \mathbf{R}_i)^{-1} B_{ik}^T S_{ik+1} A_{ik} \\ K_{i\nu} = (B_{ik}^T S_{ik+1} B_{ik} + \mathbf{R}_i)^{-1} B_{ik}^T \\ K_{i\mathbf{u}} = (B_{ik}^T S_{ik+1} B_{ik} + \mathbf{R}_i)^{-1} \mathbf{R}_i \\ S_{ik} = A_{ik}^T S_{ik+1}(A_{ik} - B_{ik}K_i) + \mathbf{Q}_i \\ \nu_{ik} = (A_{ik} - B_{ik}K_i)^T \nu_{ik+1} - K_i^T \mathbf{R}_i \mathbf{u}_{ik} + \mathbf{Q}_i \mathbf{x}_{ik} \\ \mathbf{u}_{ik}^* = \mathbf{u}_{ik} + \delta\mathbf{u}_{ik} \end{cases} . \tag{6.20}$$

### 6.3.1 Leader and Follower iLQR Controllers

By following the leader-follower formation control problem described in Subsection 4.2.1, the leader control law set is

$$\begin{cases} \delta f_{Lk} = -K_{Lz}\delta z_{Lk} - K_{Lz\nu}\nu_{zL_{k+1}} - K_{fL} f_{Lk} \\ \delta\tau_{Lqk} = -K_{Lq}\delta\mathbf{q}_{Lk} - K_{L\mathbf{q}\nu}\nu_{\mathbf{q}L_{k+1}} - K_{\tau qL}\tau_{Lqk} \\ f_{Lk} = \frac{m_L g}{q_{L0k}^2 - q_{L1k}^2 - q_{L2k}^2 + q_{L3k}^2} \\ f_{Lk}^* = f_{Lk} + \delta f_{Lk} \\ \tau_{Lqk}^* = \tau_{Lqk} + \delta\tau_{Lqk} \end{cases}$$

where

$$\delta z_{Lk} = \begin{bmatrix} z_{Lkd} - z_{Lk} \\ v_{Lzkd} - v_{Lzk} \end{bmatrix}$$

$$\delta\mathbf{q}_{Lk} = \begin{bmatrix} \mathbf{q}_{Lkd} - \mathbf{q}_{Lk} \\ \omega_{Lkd} - \omega_{Lk} \end{bmatrix}$$

and the follower control law set is

$$
\begin{cases}
\delta f_{Fk} = -K_{Fz}\delta z_{Fk} - K_{Fz\nu}\nu_{zF_{k+1}} - K_{fF} f_{Fk} \\
\delta \tau_{Fqk} = -K_{Fq}\delta \mathbf{q}_{Fk} - K_{F\mathbf{q}\nu}\nu_{\mathbf{q}F_{k+1}} - K_{\tau qF}\tau_{Fqk} \\
f_{Fk} = (g + \dot{v}_{Lz} - d(R_{q31}\cos\rho\cos\sigma + R_{q32}\cos\rho\sin\sigma + R_{q33}\sin\rho)) \\
\frac{m_F}{q_{F0k}^2 - q_{F1k}^2 - q_{F2k}^2 + q_{F3k}^2} \\
f_{Fk}^* = f_{Fk} + \delta f_{Fk} \\
\tau_{Fqk}^* = \tau_{Fqk} + \delta \tau_{Fqk}
\end{cases}
$$

where

$$
\delta z_{Fk} = \begin{bmatrix} z_{Fkd} - z_{Fk} \\ v_{Fzkd} - v_{Fzk} \end{bmatrix}
$$

$$
\delta \mathbf{q}_{Fk} = \begin{bmatrix} \mathbf{q}_{Lk} - \mathbf{q}_{Fk} \\ \omega_{Fkd} - \omega_{Fk} \end{bmatrix}.
$$

## 6.4 LQR Control

In this section, to linearise the dynamic model (6.1) to be used for the LQR controller, first order Tayler approximation around an operating point is used. The hovering point is chosen as an operating point for the linearisation purpose under the conditions, $Q = [q_0, q_1, q_2, q_3]^T = [1, 0, 0, 0]^T$, $\mathbf{v} = [v_x, v_y, v_z]^T = [0, 0, 0]^T$, and $\omega = [\omega_x, \omega_y, \omega_z]^T = [0, 0, 0]^T$ [88]. Linearisation at this operating point leads to a time invariant linear system, while using iLQR yields a time variant linear system. Then the linearised model can be written as:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) = \begin{cases} v_x \\ -2q_2 g \\ v_y \\ 2q_1 g \\ v_z \\ \frac{f}{m} \\ 0 \\ \frac{1}{2}\omega_x \\ \frac{1}{2}\omega_y \\ \frac{1}{2}\omega_z \\ \frac{1}{J_x}\tau_{q_1} \\ \frac{1}{J_y}\tau_{q_2} \\ \frac{1}{J_z}\tau_{q_3} \end{cases} . \tag{6.21}$$

The quadrotor is controlled by its altitude force $f$ and attitude torque vector $\tau$. The control vector can be defined as $\mathbf{u} = [f, \tau_{q_1}, \tau_{q_2}, \tau_{q_3}]^T$.

Consider the quadrotor a linearised time invariant system of a state-space form

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$$
$$\mathbf{y} = C\mathbf{x} + D\mathbf{u}$$

where

$$A = \begin{bmatrix} \mathbf{0}_{3\times3} & I_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & -2g & 0 & \mathbf{0}_{1\times3} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & 2g & \mathbf{0}_{1\times2} & \mathbf{0}_{1\times3} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{3\times3} & 0.5I_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}$$

$$B = \begin{bmatrix} \mathbf{0}_{5\times1} & \mathbf{0}_{5\times1} & \mathbf{0}_{5\times1} & \mathbf{0}_{5\times1} \\ 1/m & 0 & 0 & 0 \\ \mathbf{0}_{4\times1} & \mathbf{0}_{4\times1} & \mathbf{0}_{4\times1} & \mathbf{0}_{4\times1} \\ 0 & 1/J_x & 0 & 0 \\ 0 & 0 & 1/J_y & 0 \\ 0 & 0 & 0 & 1/J_z \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times8} \\ \mathbf{0}_{1\times2} & 1 & \mathbf{0}_{1\times10} \\ \mathbf{0}_{1\times4} & 1 & \mathbf{0}_{1\times8} \\ \mathbf{0}_{1\times9} & 1 & \mathbf{0}_{1\times3} \end{bmatrix}$$

$D = [\mathbf{0}_{4\times4}]$ and $\mathbf{y} = [x, y, z, q_3]^T$, with the cost function

$$\mathbf{J}_c = \frac{1}{2}\int_{t_o}^{t_f} (\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}) \tag{6.22}$$

and the linearised system state feedback

$$\mathbf{u} = -K_c\mathbf{x} = -\mathbf{R}^{-1}B^TP\mathbf{x}$$

where $P$ can be evaluated from the Ricatti equation

$$\dot{P}(t) + P(t)A + A^TP(t) - P(t)B\mathbf{R}^{-1}B^TP(t) + \mathbf{Q} = 0.$$

The goal is to find the gain $K_c$ that minimise the cost function (6.22), by applying the following law shown in (6.23), which controls the system states:

$$K_c = \mathbf{R}^{-1} B^T P. \tag{6.23}$$

Then the leader control law set is

$$\begin{bmatrix} f_L \\ \tau_{Lq} \end{bmatrix} = -K_{Lc} \begin{bmatrix} \delta z_L \\ \delta \mathbf{q}_L \end{bmatrix}$$

and the follower control law set is

$$\begin{bmatrix} f_F \\ \tau_{Fq} \end{bmatrix} = -K_{Fc} \begin{bmatrix} \delta z_F \\ \delta \mathbf{q}_F \end{bmatrix}.$$

## 6.5 Simulations

In this section, the performance of the proposed iLQR controller is compared with that of the LQR controller in attitude stabilisation, path tracking and leader-follower formation using a numerical MATLAB quadrotors simulater. The design parameters of the quadrotor used in the simulater are listed in Table 3.1.

### 6.5.1 Attitude Stabilisation

A Simulink simulation of a complete quadrotor dynamic model was performed to find the torque's parameters in order to stabilise the quaternion components. The simulated performance of using (1) iLQR for one iteration, (2) iLQR for five iterations and (3) LQR controller are shown in Figure 6.1. The RMSE values of the quaternion components are illustrated in Table 6.1.

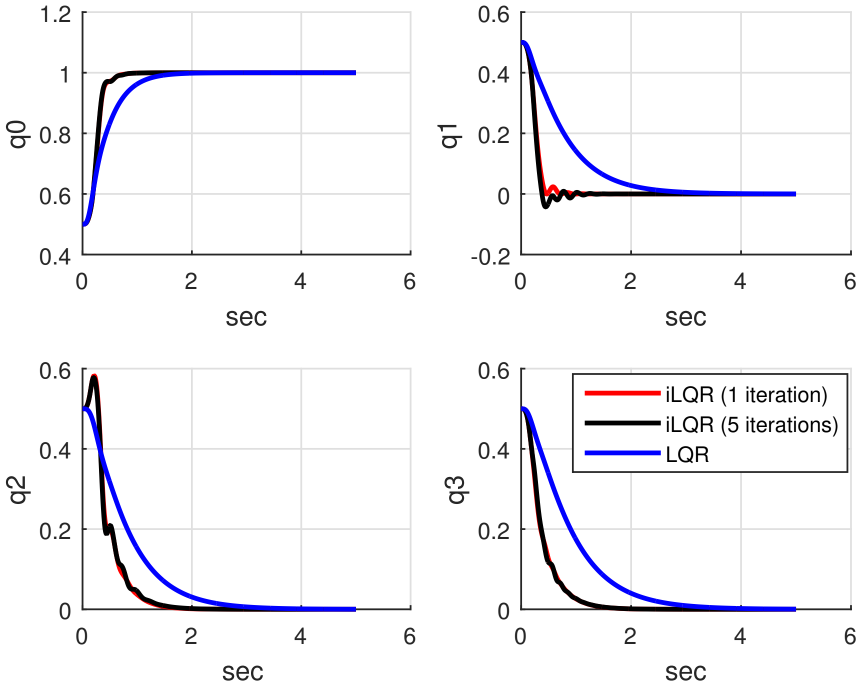

Figure 6.1: Quaternion Components under iLQR and LQR Controllers

| **RMSE** | $q_0$ | $q_1$ | $q_2$ | $q_3$ |
|---|---|---|---|---|
| iLQR (1) | 0.0047 | 0.0046 | 0.0067 | 0.0051 |
| iLQR (5) | 0.0047 | 0.0046 | 0.0066 | 0.0051 |
| LQR | 0.0053 | 0.0069 | 0.0072 | 0.0075 |

Table 6.1: Quaternion Parameter RMSE Values under iLQR and LQR Controllers

### 6.5.2 Path Tracking

The simulation results were obtained for the quadrotor to track two different desired trajectories by applying iLQR for one iteration and five iterations and LQR controllers. These trajectories are illustrated in Subsection 3.4.2.

The constant weighting matrices $\mathbf{R} = diag(0.0001, 10^-6, 10^-6, 10^-6)$, $\mathbf{Q} = diag$ $(17000, 2500, 1, 0.027, 1, 0.027, 1, 0.1)$ and $\mathbf{Q}_N = diag(0.1, 0.1, 100, 100, 100, 100, 100$ $100)$ were chosen for the iLQR controller, while $\mathbf{R} = diag(1, 10, 10, 10)$ and $\mathbf{Q} =$ $diag(17000, 2500, 35, 0.035, 35, 0.035, 0.001, 0.00015)$ were chosen for the LQR controller.

| | Path 1 | | | | Path 2 | | | |
|---|---|---|---|---|---|---|---|---|
| **RMSE** | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ | $x(m)$ | $y(m)$ | $z(m)$ | $q_3$ |
| iLQR (1) | 0.0041 | 0.0016 | 0.0041 | 2.02e-6 | 0.0370 | 0.0129 | 0.0025 | 6.04e-6 |
| iLQR (5) | 0.0040 | 0.0016 | 0.0039 | 2.01e-6 | 0.0370 | 0.0129 | 0.0017 | 6.03e-6 |
| LQR | 0.0178 | 0.0045 | 0.0250 | 1.06e-4 | 0.0272 | 0.0106 | 0.0261 | 1.5e-4 |

Table 6.2: Position and $q_3$ RMSE Values for the Two Paths under iLQR and LQR Controllers

Figures 6.2, 6.3, 6.5 and 6.6 show the 3D and 1D positions performance of tracking the two desired trajectories using the proposed iLQR (after 1 iteration and after 5 iterations) and the LQR controller, respectively. The quaternion components performance is illustrated in Figure 6.4 in the first path and Figure 6.7 in the second path. The RMSE values of the position and $q_3$ in the two paths are demonstrated in Table 6.2.

From these figures and table 6.2, comparing the behaviour of the two controllers when tracking the two paths, it can be seen that the nominal paths caught the desired paths faster when iLQR was used than when LQR was used. Moreover, the error between the nominal and desired paths when using iLQR was smaller than that when using LQR. It is also clear that the quaternion parameters using iLQR captured their desired path with less than one second and zero steady-state error compared with about three seconds to capture the desired quaternion parameters when LQR was used.

The improvement in performance from the LQR controller to one iteration for the iLQR controller is obvious. Therefore, it is clear that the iLQR controller (1 iteration) performed better than the LQR controller because the latter used a linearised model in a certain operation point while the iLQR controller linearised the model in each time step. It is also obvious that the performance improved slightly from one iteration to five iterations for the iLQR controller.

In general, it can be noticed that the tracking behaviour of the proposed iLQR controller was satisfactory compared with that of the LQR controller, and the figures show that the response of the iLQR controller was faster than that of the LQR controller with a smaller error.

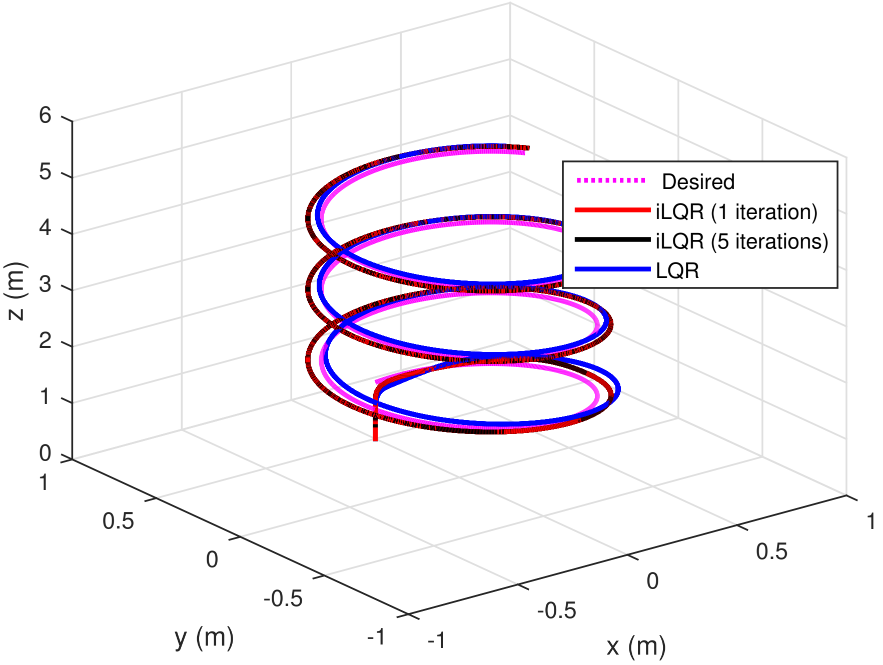

FIGURE 6.2: First Path Tracking under iLQR and LQR Controllers

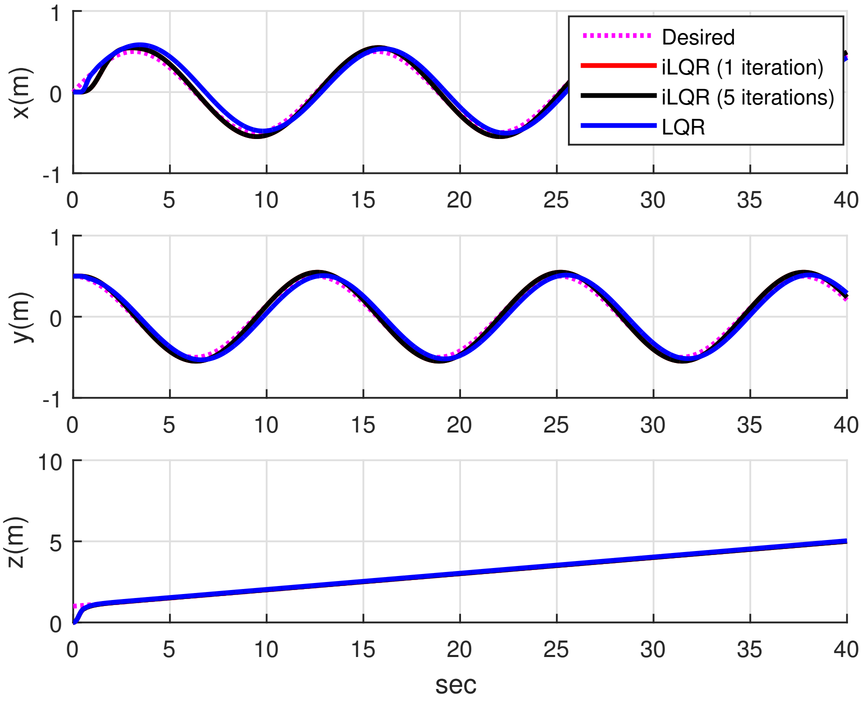

FIGURE 6.3: First Path Position under iLQR and LQR Controllers

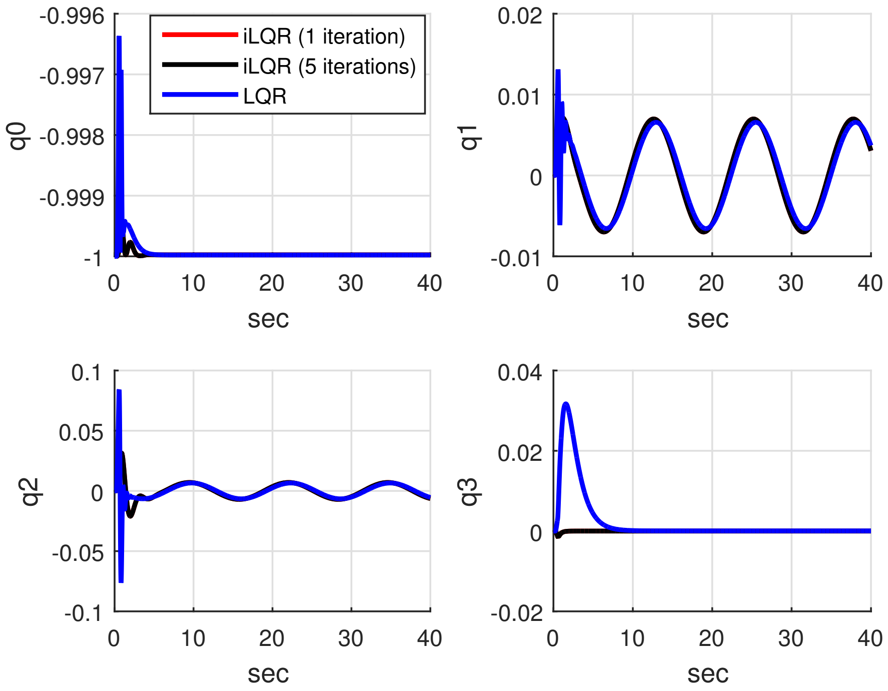

Figure 6.4: First Path Quaternion Components under iLQR and LQR Controllers

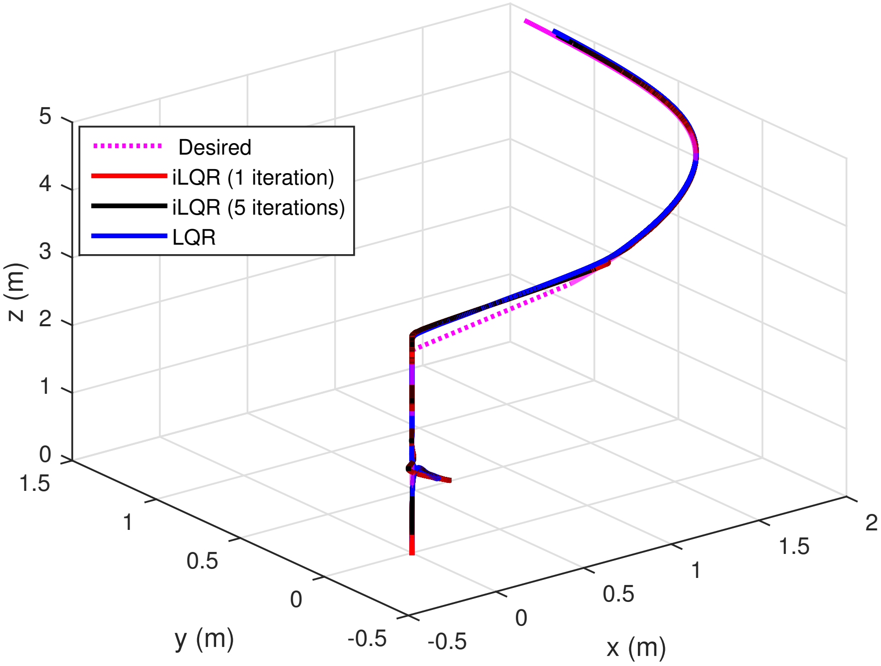

Figure 6.5: Second Path Tracking under iLQR and LQR Controllers

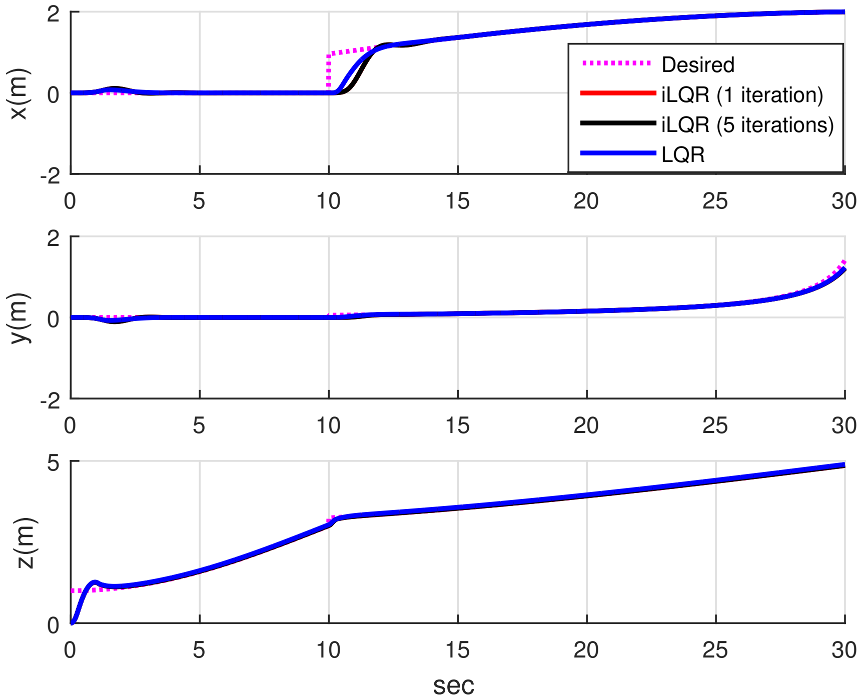

FIGURE 6.6: Second Path Position under iLQR and LQR Controllers

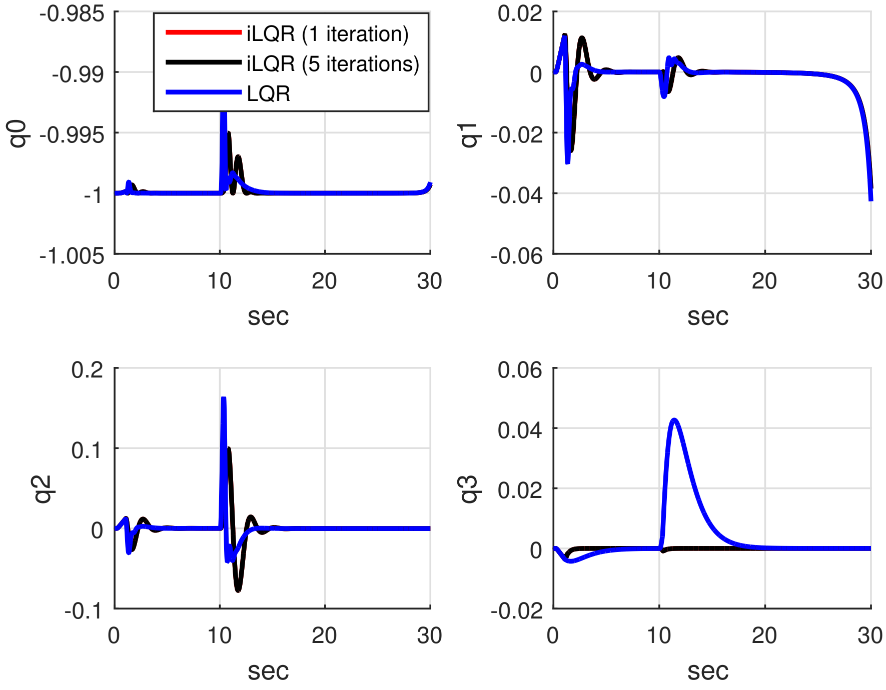

FIGURE 6.7: Second Path Quaternion Components under iLQR and LQR Controllers

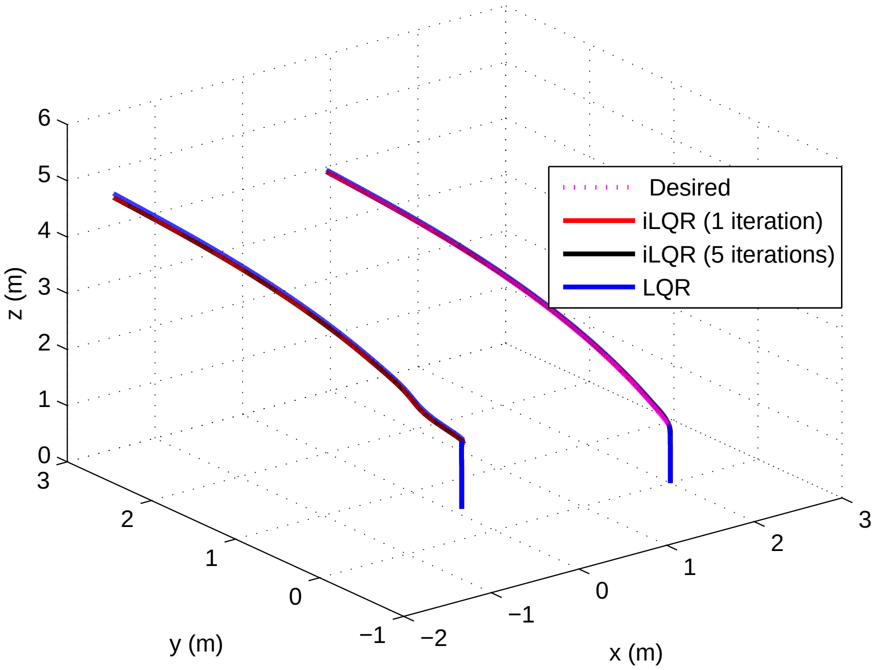

Figure 6.8: Leader-Follower Formation in First Path under iLQR and LQR Controllers

### 6.5.3 Team Formation

To validate the iLQR control strategy, it was tested in the simulation of two quadrotors in the leader-follower formation problem described in Subsection 4.2.1. Two predefined paths were used to test the proposed iLQR algorithm described earlier in Section 4.4. The two paths were also used to test the LQR control for comparison purposes.

Figures 6.8 and 6.11 show the response of the leader while tracking the predefined paths and the follower maintaining the desired distance, the bearing angle and the incidence angle, respectively. The quaternion components responses of the leader and the follower in tracking the two paths are shown in Figures 6.9, 6.10, 6.12 and 6.13, respectively. Figure 6.14 shows the distances between the leader and the follower in the two paths. The error in using the iLQR controller was smaller than that in using LQR. However, when the iLQR controller ran for five iterations, the response was slightly improved.

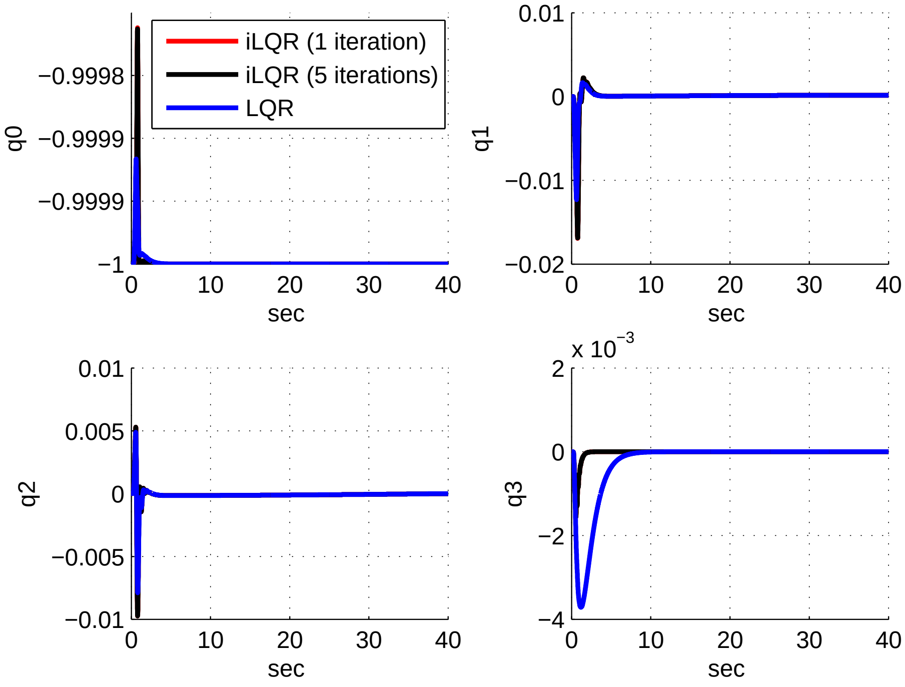

FIGURE 6.9: Leader Quaternions in First Path under iLQR and LQR Controllers

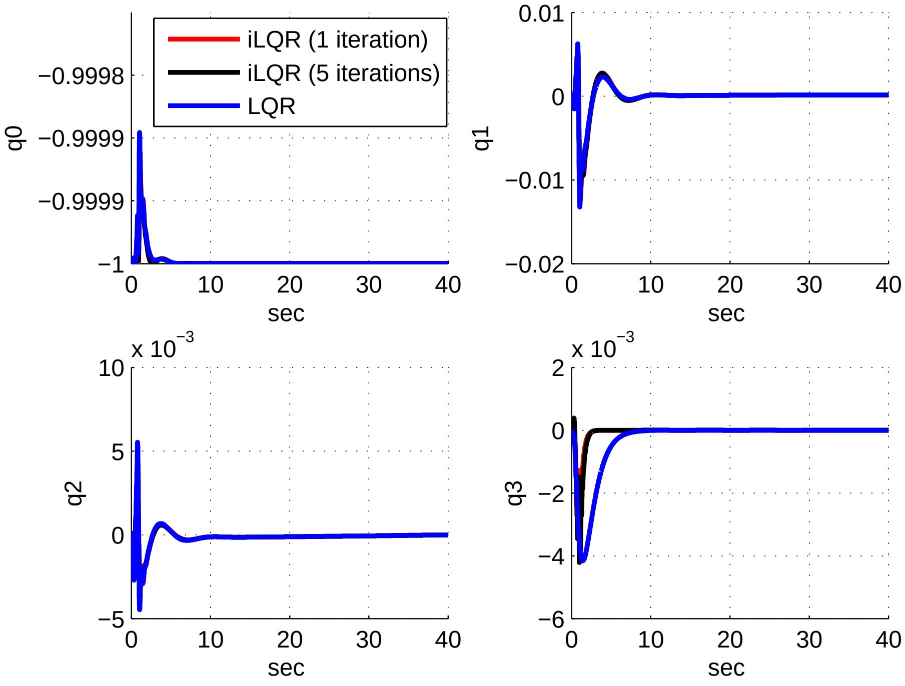

FIGURE 6.10: Follower Quaternions in First Path under iLQR and LQR Controllers

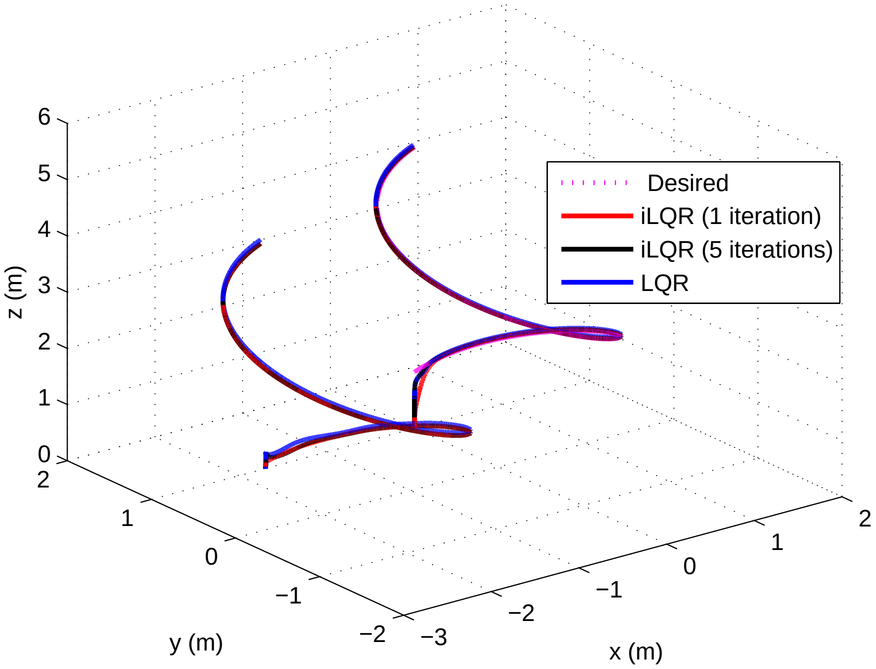

Figure 6.11: Leader-Follower Formation in Second Path under iLQR and LQR Controllers

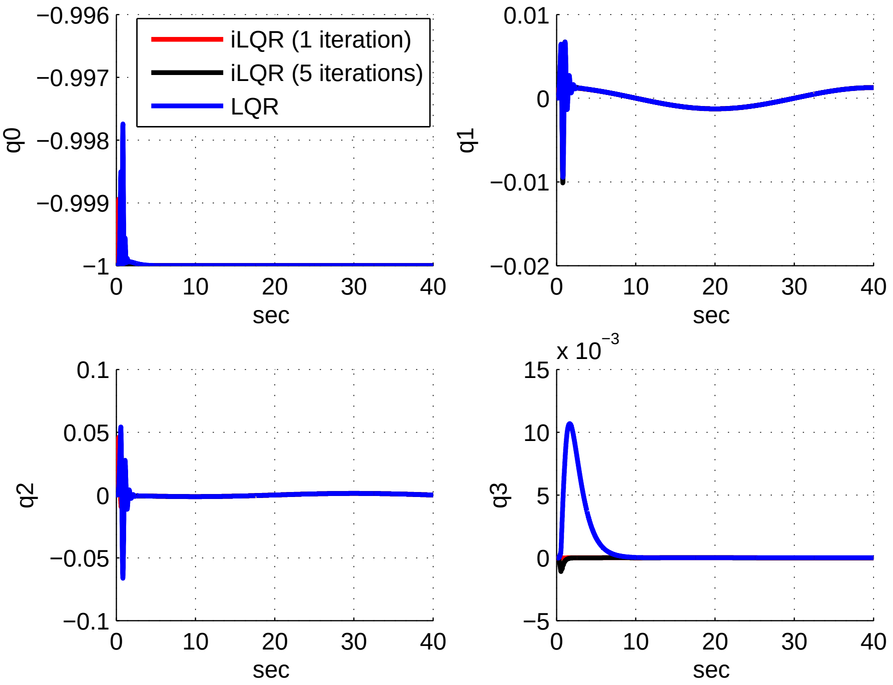

Figure 6.12: Leader Quaternions in Second Path under iLQR and LQR Controllers

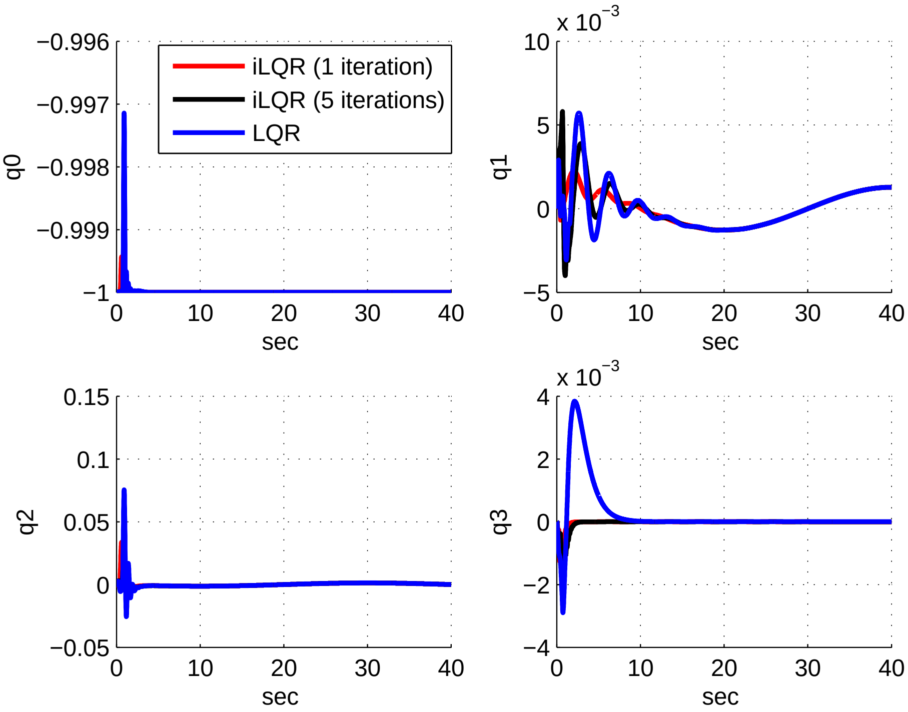

Figure 6.13: Follower Quaternions in Second Path under iLQR and LQR Controllers

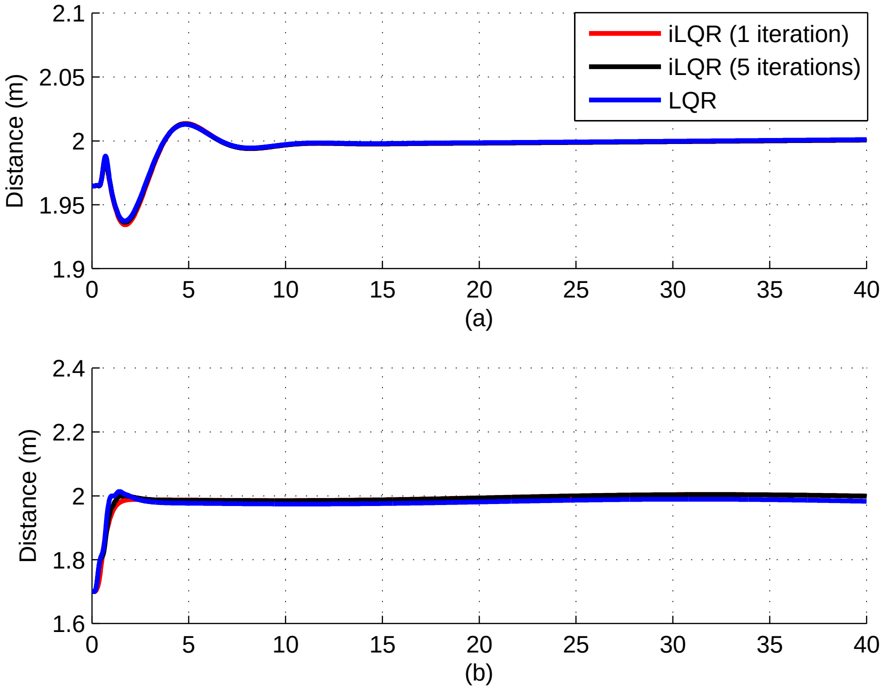

Figure 6.14: The Distance between the Leader and the Follower under iLQR and LQR Controllers in, (a) First Path, (b) Second Path

In conclusion, it is obvious that the proposed iLQR controller maintained the distance between the leader and the follower faster than LQR controller.

## 6.6 Discussions

This chapter presented an iLQR controller based on the LQR controller for quadrotor UAVs attitude stabilisation, path tracking and leader-follower team formation problems. One loop control scheme was used in simulation to find the total thrust and torques. The proposed iLQR controller was based on finding a linearised system at each time step of the operation, while the LQR controller was based on obtaining a linearised system at the operating point (hovering point). The solutions of the two controllers establish the potential of the proposed iLQR law by improving the tracking accuracy and the speed of catching the desired paths and maintaining the distances between the leader and the follower compared with the LQR controller. The iLQR controller performed better than the LQR controller, especially in quaternion components performance.

# Chapter 7

# Conclusions and Future Work

The main goal of this research was to investigate the control of stabilisation, path tracking and leader-follower systems. The research focused on designing two controllers for these purposes: the $H_\infty$ controller to reject the external disturbances and recover the model parameter uncertainties change and the iLQR controller to reach the target fast and minimise the steady state error. Each controller was supported by theoretical derivation, stability analysis and simulation results. To verify these controllers' performance, their simulation results were compared with those of $PD^2$, IBS and LQR controllers. The $H_\infty$ controller was tested practically to verify the simulation results.

The dynamic model of the quadrotor was implemented with nonlinear effects consideration. Then a one loop control scheme was used when $H_\infty$, iLQR, $PD^2$ and LQR controllers were applied and a two loop control scheme was used when the IBS controller was applied. Contributions and results of this research are concluded in this chapter. A future work plan is presented in this chapter as well.

## 7.1 Research Summary

**H$_\infty$ Controller** Van der Schaft [15] illustrated a suboptimal solution of the nonlinear $H_\infty$ output feedback robust control problem for affine nonlinear systems.

In this thesis his theory was developed to the quadrotor dynamic system with external disturbances consideration in its mathematical derivation. In particular, necessary and sufficient conditions were established for the controller stability by solving the HJI with a selection of a smooth appropriate Lyapunov function. The controller was implemented and tested in simulation and real vehicles to perform different paths and the results show the ability of the proposed controller to reject the external disturbances and recover the change in model parameter uncertainties. The main advantage of the $H_\infty$ controller over the other controllers is that the external disturbances were included in its mathematical form. Calculation of the controller gains were dependent on finding an appropriate attenuation constant $\gamma$.

**PD$^2$ Controller** When the $PD^2$ controller was used for attitude stabilisation, it failed to reject the external disturbances due to the fact that the $PD^2$ controller was a linear controller and the external disturbances were not included in its mathematical form, in addition to the fact that it was performed with higher RMS errors in the other circumstances compared with the performance of $H_\infty$ controller. Figures 3.2, 3.3 and 3.12 together with Tables 3.2 and 5.1 show a comparison between the attitude control performance of the $H_\infty$ and $PD^2$ controllers in terms of their response and RMSE of the quaternion parameters, respectively.

**IBS Controller** The IBS controller was developed based on BS control theory with adding an integral action to minimise the steady state error which appeared when the BS controller was used for path tracking and leader follower formation problems. The main drawback of the IBS controller is that its stability is guaranteed but the performance is not, and it has three coupling parameters to be tuned compared with a guaranteed stability and performance of the $H_\infty$ controller. Another noteworthy drawback of the IBS controller that was noticed in the current study is the considerable overshoot in its response due to the effect of the integral parameter and high oscillations when external disturbances were applied to the system dynamics or when the controller failed to recover in leader-follower formation.

**iLQR Controller** An iLQR controller is essentially based on a LQR controller with an iteration technique. It has a set of gains equal to the number of operating samples by linearising the system in each sample of operation. In the current study, the one iteration iLQR controller performed better than the LQR controller when tracking the quaternion components, tracking different paths and maintaining the distance between the leader and the follower in terms of the speed of catching the desired target and minimising steady state errors. These findings are related to the linearisation method used for the iLQR controller compared with a single operation point linearisation technique utilised in the LQR algorithm. A slight improvement in the performance was noticed after five iterations. The comparison is very clear in Tables 6.1 and 6.2.

**LQR Controller** Again a good performance was achieved when the LQR controller was simulated. The LQR controller performance showed an optimal solution with no overshoot and a very small steady state error. The main advantage of using the LQR controller is that stability is guaranteed and, by carefully choosing a couple of parameters, the controller is automatically guaranteed. The disadvantage, however, is that the system is linearised in one operating condition. In this thesis, an iLQR technique was proposed to overcome this problem.

In conclusion, all the controllers mentioned in this work performed very well in normal conditions with some differences. The $H_\infty$ controller performance showed high robustness to the external disturbances effect and model uncertainties change, in addition to the fact that its stability was guaranteed. The $PD^2$ controller failed to reject the external disturbances. The IBS controller rejected a limited external disturbance and sometimes it failed to do so. However, the iLQR controller performed faster than the LQR controller with less steady state errors.

Of particular interest is the fact that the performance the $H_\infty$ and iLQR controllers was further improved in comparison with previous robust nonlinear approaches. In simulations, the maximum error was less than 2 cm compared with about 10 cm error obtained by [86] with the presence of external disturbances, and the two controllers were able to catch the target with less than 3 seconds. In real tests, the proposed $H_\infty$ controller obtained a less than 5 cm error and less than 5 seconds to achieve the formation compared with more than 8 seconds in [107].

## 7.2 Future Work

In spite of our promising results, represented by the stability and robustness of the proposed linear and nonlinear controllers and the effectiveness of the leader-follower formation scheme presented in this thesis, there are many possibilities on how to extend the demonstrated work.

**Extending the proposed control stability and robustness**

The proposed control techniques considered an obstacle-free flight environment. The future work toward this direction is to enhance the stability and robustness of the proposed controllers to avoid static and moving obstacles.

**Implementing the control techniques in real vehicles**

As mentioned in this thesis, only the $H_\infty$ controller was tested practically. Firstly, it would be very interesting to practically implement the iLQR controller on a real vehicle to further verify its simulation results. Secondly, to further extend the real quadrotors flight stability, especially in leader-follower team formation laser range finders, cameras or laser sonars to find the pose of the leader quadrotor are to be used by the follower to maintain the distance between the leader and the follower. Thirdly, it is beneficial to test the proposed controllers with flying outside the Vicon system.

**Team formation shape selection**

In terms of the shape selection of the leader-follower team formation and the increase in the number of follower robots, the future work is to relax the desired incidence and bearing angles. This objective may require more complex control laws, particularly when the follower has to maintain the distances with several robots. Upon using the Vicon system to calculate the pose of the quadrotors, sending and receiving the pose among the robots and the control law calculation are likely to consume very long time and cause too much delay. Therefore, sensors like cameras and laser sonars may prove very useful in this case.

### Robustness and performance balancing

As a robust control law with the use of the $H_\infty$ controller was found and guaranteed and optimal performance was achieved by the iLQR controller, the future step toward this issue is to find a balanced controller by gathering both the $H_\infty$ and iLQR controllers to control the quadrotors.

# Appendix A

# Dynamical Model Based on Euler Angles

## A.1 Introduction

The quadrotor UAV has four motors with four propellers mounted over the motors, each opposite pair working together in an opposite direction of rotation to the other pair. The take-off or vertical motion is generated by increasing or decreasing the four motors' speed.

Increasing the speed of the motor of one of each pair and decreasing the other generate the rotation and motion of the vehicle. Namely, increasing the number (1) motor's speed and decreasing the number (3) motor's speed and vice versa produce a pitch rotation and a sloping motion. Increasing the number (2) motor's speed and decreasing the number (4) motor's speed and vice versa produce a roll rotation and a sloping motion, while the difference between each pair propellers torque produces the yaw rotation [3][40].

## A.2 Mathematical Model

To control the motion and rotation of the quadrotor UAV, first the mathematical dynamic model should be achieved. The quadrotor UAV system has a nonlinear dynamic system and complicated structure; therefore, it is difficult to represent its motion and rotation in a simple model. The dynamic model of the quadrotor UAV depends on some assumptions [1]:

- The structure of the quadrotor is rigid and symmetrical;
- The propellers are rigid;
- The centre of mass and body fixed frame are coincides;
- Thrust and drugs are proportional to the square of the propellers; and
- The difference of gravity by altitude or the spin of the earth is minor.

According to these assumptions, the mathematical model can be derived to perform the quadrotor UAV fuselage dynamics in space, where it will be easy to add to it the effects of aerodynamic forces generated by the rotation of the propellers. The coordinate reference system of the quadrotor includes two frames of reference, the inertial (earth fixed) frame mentioned $\mathcal{I}(x_{\mathcal{I}}, y_{\mathcal{I}}, z_{\mathcal{I}})$ and the rigid (body fixed) frame mentioned $\mathcal{B}(x_{\mathcal{B}}, y_{\mathcal{B}}, z_{\mathcal{B}})$. Several techniques can be used to perform the rigid body rotation in space such as Euler angles, Quaternions and Tait-Bryan angles [102]. The rotation matrix in the space will be described below using the Euler angles method, in which the front direction is mentioned to the $x - axis$ and the side direction is mentioned to the $y - axis$ while the $z - axis$ is mentioned by the vertical direction. And $(x, y, z)$ directions follow the right-hand rule. These three directions generate the roll, pitch and yaw angles, respectively.

### A.2.1 Euler Angle Representation

The orientation of the Cartesian coordinate system is used to parameterise the Euler angle, one with respect to another, by creating three positive rotations using the Right-hand rotation technique. The first rotation is around $x - axis$, the second rotation is around the new $y - axis$ whereas the third rotation is around the new $z - axis$. The three axes rotations can be depicted by the following three matrices [102][78].

$$R_\theta(x,\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{bmatrix} \tag{A.1}$$

$$R_\theta(y,\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \tag{A.2}$$

$$R_\theta(z,\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.3}$$

Multiplication of these three above matrices (A.1)-(A.3) produces the rotational matrix from the fixed frame to the body frame;

$$R_\theta(\psi,\theta,\varphi) = R_\theta(z,\varphi)R_\theta(y,\theta)R_\theta(x,\psi). \tag{A.4}$$

This results in

$$R_\theta = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\varphi - \sin\psi\cos\varphi & \cos\psi\sin\theta\cos\varphi + \sin\psi\sin\varphi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\varphi + \cos\psi\cos\varphi & \sin\psi\sin\theta\cos\varphi - \sin\varphi\cos\psi \\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi \end{bmatrix}. \tag{A.5}$$

### A.2.2 Coriolis Equation

Derivation of the Coriolis equation depends on the two frames $\mathcal{I}$ and $\mathcal{B}$ with steps given in [125], assuming that the $P$ vector is moving in the $\mathcal{B}$ frame and the $\mathcal{B}$ frame is not rotating w.r.t. $\mathcal{I}$ frame. The time derivative of the vector $P$ in $\mathcal{I}$ and

$\mathcal{B}$ frames is

$$\frac{d}{dt_{\mathcal{I}}}P = \frac{d}{dt_{\mathcal{B}}}P. \tag{A.6}$$

Assuming that the vector $P$ is stationary in the $\mathcal{B}$ frame and the frame itself is rotating w.r.t. $\mathcal{I}$ frame, and then we obtain the equation of this rotation as:

$$P + \delta P = (1 - \cos\delta\varphi)\varpi(\varpi \cdot P) + \cos(-\delta\varphi)P - \sin(-\delta\varphi)\varpi \times P. \tag{A.7}$$

Dividing Equation (A.7) by $\delta t$ and using the approximation of small angle we obtain

$$\frac{\delta P}{\delta t} \approx \frac{\delta\varphi}{\delta t}\varpi \times P \tag{A.8}$$

where $\varpi$ is the instantaneous axis of rotation and $\delta\varphi$ is the right hand rotation angle. If we take the limit $\delta t = 0$ and the angular velocity of the $\mathcal{B}$ frame w.r.t. $\mathcal{I}$ frame is $\omega_{\mathcal{B}/\mathcal{I}} = \varpi\dot{\varphi}$ we get

$$\frac{d}{dt_{\mathcal{I}}}P = \omega_{\mathcal{B}/\mathcal{I}} \times P. \tag{A.9}$$

Then the Coriolis equation can be obtained by combining Equations (A.6) and (A.9);

$$\frac{d}{dt_{\mathcal{I}}}P = \frac{d}{dt_{\mathcal{B}}}P + \omega_{\mathcal{B}/\mathcal{I}} \times P. \tag{A.10}$$

### A.2.3 Quadrotor Kinematics and Dynamics

The Newton Euler formula of the dynamics of a solid shape under the effect of external forces applied to the centre mass is distinct in the body fixed frame as shown in [72].

#### A.2.3.1 For Translational Motion

$$m\frac{d\mathbf{v}}{dt_{\mathcal{I}}} = f. \tag{A.11}$$

Applying the Coriolis equation to (A.11) we have

$$m\frac{d\mathbf{v}}{dt_{\mathcal{I}}} = m(\frac{d\mathbf{v}}{dt_{\mathcal{I}}} + \omega_{\mathcal{B}/\mathcal{I}} \times \mathbf{v}) = f. \tag{A.12}$$

Applying Equation (A.12) in body coordinates with $\mathbf{v}^{\mathcal{B}} = (u, v, w)^T$ and $\omega^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} = (\omega_x, \omega_y, \omega_z)^T$ it will be:

$$m \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = m(0 + \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix}) = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \tag{A.13}$$

or

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \omega_z v - \omega_y w \\ \omega_x w - \omega_z u \\ \omega_y u - \omega_x v \end{bmatrix} + \frac{1}{m} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}. \tag{A.14}$$

#### A.2.3.2 For Rotational Motion

From Newton's second law

$$\frac{dh^{\mathcal{B}}}{dt_{\mathcal{I}}} = m. \tag{A.15}$$

Applying the equation of Coriolis to Equation (A.15) we get

$$\frac{dh}{dt_{\mathcal{I}}} = \frac{dh}{dt_{\mathcal{B}}} + \omega_{\mathcal{B}/\mathcal{I}} \times h = m. \tag{A.16}$$

From the body coordinate we have $h^{\mathcal{B}} = J\omega^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}$, then Equation (A.16) can be resolved in the body coordinate frame. The equations of motion of the quadrotor UAVs depend on the two frames which can be written as in [126].

$$\begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = 0 + \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \times \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \tag{A.17}$$

or

$$\begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} \frac{1}{J_x} & 0 & 0 \\ 0 & \frac{1}{J_y} & 0 \\ 0 & 0 & \frac{1}{J_z} \end{bmatrix} ( \begin{bmatrix} \omega_y \omega_z (J_y - J_z) \\ \omega_x \omega_z (J_x - J_z) \\ \omega_x \omega_y (J_x - J_y) \end{bmatrix} + \begin{bmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} ). \tag{A.18}$$

or

$$\begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} \frac{\omega_y \omega_z (J_y - J_z)}{J_x} \\ \frac{\omega_x \omega_z (J_x - J_z)}{J_y} \\ \frac{\omega_x \omega_y (J_x - J_y)}{J_z} \end{bmatrix} + \begin{bmatrix} \frac{\tau_\varphi}{J_x} \\ \frac{\tau_\theta}{J_y} \\ \frac{\tau_\psi}{J_z} \end{bmatrix}. \tag{A.19}$$

The relationship between position and velocities is given by

$$\frac{d}{dt} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = R_\theta^T \begin{bmatrix} u \\ v \\ w \end{bmatrix}. \tag{A.20}$$

The relationship between absolute angles $[\varphi, \theta, \psi]$ and the angular rates $[\omega_x, \omega_y, \omega_z]$ is

$$\begin{aligned} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} &= J \begin{bmatrix} \dot{\varphi} \\ 0 \\ 0 \end{bmatrix} + R_\theta(x, \varphi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_\theta(x, \varphi) R_\theta(y, \theta) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\varphi & \sin\varphi\cos\theta \\ 0 & -\sin\varphi & \cos\varphi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}. \end{aligned} \tag{A.21}$$

By inverting we obtain

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\varphi\tan\theta & -\cos\varphi\tan\theta \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi\sec\theta & \cos\varphi\sec\theta \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \tag{A.22}$$

Then the six-degree-of-freedom model for the quadrotor kinematics and dynamics can be summarised as follows [126]:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\varphi - \sin\psi\cos\varphi & \cos\psi\sin\theta\cos\varphi + \sin\psi\sin\varphi \\ \cos\theta\sin\psi & \sin\varphi\sin\theta\sin\psi + \cos\varphi\cos\psi & \sin\psi\sin\theta\cos\varphi - \sin\varphi\cos\psi \\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{A.23}$$

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \omega_z v - \omega_y w \\ \omega_x w - \omega_z u \\ \omega_y u - \omega_x v \end{bmatrix} + \frac{1}{m} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \tag{A.24}$$

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\varphi\tan\theta & -\cos\varphi\tan\theta \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi\sec\theta & \cos\varphi\sec\theta \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{A.25}$$

$$\begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} \frac{\omega_y\omega_z(J_y-J_z)}{J_x} \\ \frac{\omega_x\omega_z(J_x-J_z)}{J_y} \\ \frac{\omega_x\omega_y(J_x-J_y)}{J_z} \end{bmatrix} + \begin{bmatrix} \frac{\tau_\varphi}{J_x} \\ \frac{\tau_\theta}{J_y} \\ \frac{\tau_\psi}{J_z} \end{bmatrix}. \tag{A.26}$$

The total force applied to the quadrotor is given by $f = f_1 + f_2 + f_3 + f_4$ and the torque applied on the UAV's body which is created by the propellers $\tau$ and is equal to the difference between each pair of opposite propellers is

$$\begin{bmatrix} \tau_\varphi \\ \tau_\theta \\ \tau\psi \end{bmatrix} = \begin{bmatrix} l(f_4 - f_2) \\ l(f_1 - f_3) \\ f_2 + f_4 - f_1 - f_3 \end{bmatrix}. \tag{A.27}$$

The gravity is another body force effect on the centre of mass of the quadrotor and it can be written as:

$$f_g = R_\theta^T \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} = \begin{bmatrix} mg\sin\theta \\ -mg\cos\theta\sin\varphi \\ -mg\cos\theta\cos\varphi \end{bmatrix}. \tag{A.28}$$

Then Equation (A.24) can be rewritten as:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \omega_z v - \omega_y w \\ \omega_x w - \omega_z u \\ \omega_y u - \omega_x v \end{bmatrix} + \begin{bmatrix} mg\sin\theta \\ -mg\cos\theta\sin\varphi \\ -mg\cos\theta\cos\varphi \end{bmatrix} + \frac{1}{m} \begin{bmatrix} 0 \\ 0 \\ f \end{bmatrix}. \tag{A.29}$$

Assuming that $[\varphi, \theta, \omega_x, \omega_y, \omega_z]$ are small, we obtain the following equations from Equations (A.25) and (A.26):

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{A.30}$$

$$\begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} \frac{\tau_\varphi}{J_x} \\ \frac{\tau_\theta}{J_y} \\ \frac{\tau_\psi}{J_z} \end{bmatrix}. \tag{A.31}$$

Differentiating Equation (A.30) and substituting Equation (A.31) we obtain:

$$\begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{\tau_\varphi}{J_x} \\ \frac{\tau_\theta}{J_y} \\ \frac{\tau_\psi}{J_z} \end{bmatrix}. \tag{A.32}$$

Other moment effects on the body of the quadrotor are the body gyro effect

$$G(\omega) = \begin{bmatrix} \frac{J_r}{J_x}\dot{\theta}\Omega \\ \frac{-J_r}{J_y}\dot{\varphi}\Omega \\ 0 \end{bmatrix} \tag{A.33}$$

and the propeller gyro effect

$$G(p) = \begin{bmatrix} \dot{\theta}\dot{\psi}\frac{J_y - J_z}{J_x} \\ \dot{\varphi}\dot{\psi}\frac{J_z - J_x}{J_y} \\ \dot{\varphi}\dot{\theta}\frac{J_x - J_y}{J_z} \end{bmatrix}. \tag{A.34}$$

By adding them to (A.32) we get

$$\begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \dot{\theta}\dot{\psi}\frac{J_y - J_z}{J_x} \\ \dot{\varphi}\dot{\psi}\frac{J_z - J_x}{J_y} \\ \dot{\varphi}\dot{\theta}\frac{J_x - J_y}{J_z} \end{bmatrix} + \begin{bmatrix} \frac{J_r}{J_x}\dot{\theta}\Omega \\ \frac{-J_r}{J_y}\dot{\varphi}\Omega \\ 0 \end{bmatrix} + \begin{bmatrix} \frac{\tau_\varphi}{J_x} \\ \frac{\tau_\theta}{J_y} \\ \frac{\tau_\psi}{J_z} \end{bmatrix}. \tag{A.35}$$

Differentiating Equation (A.23), substituting Equation (A.29), ignoring $\dot{R}_\theta$ and neglecting the Coriolis terms we obtain

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \begin{bmatrix} \cos\varphi\sin\theta\cos\varphi + \sin\psi\sin\varphi \\ \cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi \\ \cos\varphi\cos\theta \end{bmatrix} \frac{f}{m}. \tag{A.36}$$

The full mathematical model is

$$\begin{cases} \ddot{x} = (\cos\varphi\cos\psi\sin\theta + \sin\varphi\sin\psi)\frac{f}{m} \\ \ddot{y} = (\cos\varphi\sin\psi\sin\theta - \sin\varphi\cos\psi)\frac{f}{m} \\ \ddot{z} = -g + (\cos\varphi\cos\theta)\frac{f}{m} \\ \ddot{\varphi} = \dot{\theta}\dot{\psi}\frac{J_y - J_z}{J_x} - \frac{J_r}{J_x}\dot{\theta}\Omega + \frac{l}{J_x}\tau_\varphi \\ \ddot{\theta} = \dot{\varphi}\dot{\psi}\frac{J_z - J_x}{J_y} - \frac{J_r}{J_y}\dot{\varphi}\Omega + \frac{l}{J_y}\tau_\theta \\ \ddot{\psi} = \dot{\theta}\dot{\varphi}\frac{J_x - J_y}{J_z} + \frac{l}{J_z}\tau_\psi \end{cases} \tag{A.37}$$

where

$$\begin{cases} f = B(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ \tau_\varphi = B(\Omega_4^2 - \Omega_2^2) \\ \tau_\theta = B(\Omega_3^2 - \Omega_1^2) \\ \tau_\psi = D(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \\ \Omega = \Omega_2 + \Omega_4 - \Omega_1 - \Omega_3 \end{cases} . \tag{A.38}$$

# Appendix B

# Quaternion Representation

## B.1 Quaternion Mathematics

An alternative method used to describe the position and orientation of the quadrotor is the quaternion method. This method is used to overcome the singularity problem which encounter researchers who use the Euler angles representation. It is a singularity outcome of the so called gimbal lock and it appears when dividing the pitch angles $\theta = \pm 90^o$ by zero. It is a hyper complex number of 4-tuple $(q_0, q_1, q_2, q_3) \in \mathbb{R}^4$ which can be written in many ways as $Q = q_0 + q_1 i + q_2 j + q_3 k$ and $Q = [q_0, \mathbf{q}^T]^T$ [127][128][37].

The north east down (NED) coordinate system is used to parametrise the dynamic model of the quadrotor with an angle of one-axis rotation $\alpha$ around the Euler axis of unit vector $\mathrm{k} \in \mathbb{R}^3$ which has a direct physical connection and can be written as:

$$Q = \begin{bmatrix} \cos\frac{\alpha}{2} \\ \mathrm{k}\sin\frac{\alpha}{2} \end{bmatrix} \tag{B.1}$$

where $\mathrm{k} = \frac{\mathbf{q}}{\|\mathbf{q}\|}$ and $\alpha = 2\arccos q_0$. Moreover, as any complex number the norm, complex conjugate and inverse of the quaternion can be defined as:

$$\|Q\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \tag{B.2}$$

$$\bar{Q} = \begin{bmatrix} q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix} \tag{B.3}$$

$$Q^{-1} = \frac{\bar{Q}}{\|Q\|}. \tag{B.4}$$

The unit quaternion can be used to represent the coordinate transformation between the inertial frame $\mathcal{I}$ and the body frame $\mathcal{B}$ by defining the multiplication and the inverse quaternion. The multiplication of two quaternions $Q = [q_0, \mathbf{q}^T]^T$ and $Q' = [{q'}_0, \mathbf{q'}^T]^T$ is defined as:

$$\begin{aligned} Q \otimes Q' &= \begin{bmatrix} q_0 & -\mathbf{q}^T \\ \mathbf{q} & q_0 I + S(\mathbf{q}) \end{bmatrix} \begin{bmatrix} {q'}_0 \\ \mathbf{q}' \end{bmatrix} \\ &= \begin{bmatrix} q_0 {q'}_0 - \mathbf{q}^T \mathbf{q}' \\ {q'}_0 \mathbf{q} + q_0 \mathbf{q}' + S(\mathbf{q})\mathbf{q}' \end{bmatrix}. \end{aligned}$$

The inverse unit quaternion is defined as $Q^{-1} = [q_0, -\mathbf{q}^T]^T$ for $Q = [q_0, \mathbf{q}^T]^T$. A vector $\mathrm{x}_{\mathcal{I}} \in \mathbb{R}^3$ in the inertial frame can be expressed as a vector $\mathrm{x}_{\mathcal{B}} \in \mathbb{R}^3$ in the body frame via $\mathrm{x}_{\mathcal{B}} = R^T \mathrm{x}_{\mathcal{I}}$. Using $\bar{\mathrm{x}} = [0, \mathrm{x}^T]^T$, the transformation from the inertial frame to the body frame is expressed as $\bar{\mathrm{x}}_{\mathcal{B}} = Q^{-1} \otimes \bar{\mathrm{x}}_{\mathcal{I}} \otimes Q$.

And if the norm of the quaternion is equal to one $\|Q\| = 1$, it means that the inverse is the same as the conjugate, which is the case used to represent the coordinate transformation between the inertial frame $\mathcal{I}$ and the body frame $\mathcal{B}$ by defining the multiplication and the inverse quaternion. The multiplication of two quaternions $Q = [q_0, \mathbf{q}^T]^T$ and $Q' = [{q'}_0, \mathbf{q'}^T]^T$ is defined as:

$$\begin{aligned} Q \otimes Q' &= \begin{bmatrix} q_0 & -\mathbf{q}^T \\ \mathbf{q} & q_0 I + S(\mathbf{q}) \end{bmatrix} \begin{bmatrix} {q'}_0 \\ \mathbf{q}' \end{bmatrix} \\ &= \begin{bmatrix} q_0 {q'}_0 - \mathbf{q}^T \mathbf{q}' \\ {q'}_0 \mathbf{q} + q_0 \mathbf{q}' + S(\mathbf{q})\mathbf{q}' \end{bmatrix} \end{aligned} \tag{B.5}$$

where $S : \mathbb{R}^4 \rightarrow \mathbb{R}^{3\times3}$ is the skew-symmetric cross product matrix, and $Q_S : \mathbb{R}^4 \rightarrow \mathbb{R}^{4\times4}$ is the quaternion skew-symmetric cross matrix and they are defined as:

$$S(x) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \tag{B.6}$$

$$Q_S(Q) = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & q_3 & -q_2 \\ q_2 & -q_3 & q_0 & q_1 \\ q_3 & q_2 & -q_1 & q_0 \end{bmatrix} \tag{B.7}$$

$$\bar{Q}_S(Q) = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix}. \tag{B.8}$$

The derivative of the quaternion $Q$ is linked with the quadrotor angular velocity as follows:

$$\dot{Q}'_\omega(Q, \omega') = \frac{1}{2} \begin{bmatrix} 0 \\ \omega' \end{bmatrix} \otimes Q = \frac{1}{2}\bar{Q}_S(Q) \begin{bmatrix} 0 \\ \omega' \end{bmatrix} \tag{B.9}$$

$$\dot{Q}_\omega(Q, \omega) = \frac{1}{2} Q \otimes \begin{bmatrix} 0 \\ \omega \end{bmatrix} = \frac{1}{2} Q_S(Q) \begin{bmatrix} 0 \\ \omega \end{bmatrix}. \tag{B.10}$$

However, as mentioned above, the quaternion is a unit vector which is utilised as a rotation operator. Then the rotation from the fixed frame to the body frame requires a rotational matrix which is the same as in the Euler angles method but it does not contain trigonometric functions which can be evaluated by rotating a vector from the fixed frame to the body frame as follows:

$$\begin{aligned} \begin{bmatrix} 0 \\ k' \end{bmatrix} &= Q \otimes \begin{bmatrix} 0 \\ k \end{bmatrix} \otimes Q^{-1} = Q \otimes \begin{bmatrix} 0 \\ k \end{bmatrix} \otimes \bar{Q} \\ &= \bar{Q}_S(Q)^T Q_S(Q) \begin{bmatrix} 0 \\ k \end{bmatrix} = \begin{bmatrix} 1 & 0^T \\ 0 & R_q \end{bmatrix} \begin{bmatrix} 0 \\ k \end{bmatrix} \end{aligned} \tag{B.11}$$

where $k \in \mathbb{R}^3$ is a vector to be rotated from the fixed frame to the body frame and

$$R_q = \begin{bmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix}; \tag{B.12}$$

that is $k' = R_q k$ and $k = R_q^T k'$.

Computing the quaternion parameters from Euler angles or computing the Euler angles from the quaternion parameters can be presented using the relationships [35]:

$$Q = \begin{bmatrix} \cos(\frac{\varphi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2})+\sin(\frac{\varphi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2}) \\ \sin(\frac{\varphi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2})-\cos(\frac{\varphi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2}) \\ \cos(\frac{\varphi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2})+\sin(\frac{\varphi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2}) \\ \cos(\frac{\varphi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2})-\sin(\frac{\varphi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2}) \end{bmatrix} \tag{B.13}$$

$$\begin{bmatrix} \varphi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \arctan 2(2(q_0q_1+q_2q_3), q_0^2-q_1^2-q_2^2+q_3^2) \\ \arcsin(2(q_0q_2-q_1q_3)) \\ \arctan 2(2(q_0q_3+q_1q_2), q_0^2+q_1^2-q_2^2-q_3^2) \end{bmatrix}. \tag{B.14}$$

### B.1.1 Quaternion Kinematics

The kinematic equations of the movements of a unit quaternion $Q(t)$ can be driven by rotating the quadrotor with its angular velocity vector $\omega$ in the three directions to make a slight change in the movement of the quadrotor $\Delta t$ and the change will be as follows [88]:

$$Q(t+\Delta t) = \begin{bmatrix} \cos(\frac{\Delta\alpha}{2})I + \sin(\frac{\Delta\alpha}{2}) \begin{bmatrix} 0 & n_3 & -n_2 & n_1 \\ -n_3 & 0 & n_1 & n_2 \\ n_2 & -n_1 & 0 & n_3 \\ -n_1 & -n_2 & -n_3 & 0 \end{bmatrix} \end{bmatrix} Q(t) \tag{B.15}$$

where $\Delta\alpha = \omega\Delta t$. Then if $\Delta t$ is considered small, these expressions hold, $\cos(\frac{\Delta\alpha}{2}) \cong 1$, $\sin(\frac{\alpha}{2}) \cong \frac{1}{2}\omega\Delta t$. According to these assumptions, Equation (B.15) can be written as:

$$Q(t+\Delta t) = \left[\ 1 + \tfrac{1}{2}S(\omega)\Delta t\ \right] Q(t). \tag{B.16}$$

Thus the kinematic quaternion movement is

$$\dot{Q} = \lim_{\Delta t \to 0} \frac{Q(t+\Delta t) - Q(t)}{\Delta t} = \frac{1}{2}S(\omega)Q \tag{B.17}$$

where

$$S_S(\omega) = \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix}. \tag{B.18}$$

Then the time derivative of the quaternion kinematics can be written in the following two forms:

$$\dot{Q} = \frac{1}{2}Q \otimes \begin{bmatrix} 0 \\ \omega \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 \\ \omega \end{bmatrix} \otimes Q. \tag{B.19}$$

### B.1.2 Quadrotor Kinematics and Dynamics

The quaternion formula of the dynamics of a solid shape under the effect of external forces applied to the centre mass which is distinct in the body fixed frame can be separated into translational and rotational motions and it can be defined by recall (A.11)-(A.20) and (A.27) with the use of the rotation matrix $R_q$; hence, the effect of gravity can be written as:

$$f_g = R_q \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} = \begin{bmatrix} -2mg(q_1q_3 + q_0q_2) \\ -2mg(q_2q_3 - q_0q_1) \\ -mg(q_0^2 - q_1^2 - q_2^2 + q_3^2) \end{bmatrix}. \tag{B.20}$$

Then Equations (A.29) and (A.36) can be rewritten as:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \omega_z v - \omega_y w \\ \omega_x w - \omega_z u \\ \omega_y u - \omega_x v \end{bmatrix} + \begin{bmatrix} -2mg(q_1 q_3 + q_0 q_2) \\ -2mg(q_2 q_3 - q_0 q_1) \\ -mg(q_0^2 - q_1^2 - q_2^2 + q_3^2) \end{bmatrix} + \frac{1}{m} \begin{bmatrix} 0 \\ 0 \\ f \end{bmatrix} \tag{B.21}$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \begin{bmatrix} 2(q_1 q_3 + q_0 q_2) \\ 2(q_2 q_3 - q_0 q_1) \\ q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \frac{f}{m}. \tag{B.22}$$

In the rotational motion part, instead of using the Euler angles, two differential equations hold: the quaternion and the angular velocity differential equation. The quaternion rate equation can be rewritten as:

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} 0 \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \tag{B.23}$$

Then the full model for the quadrotor kinematics and dynamics can be summarised as follows:

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \begin{bmatrix} 2(q_1 q_3 + q_0 q_2) \\ 2(q_2 q_3 - q_0 q_1) \\ q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \frac{f}{m} \tag{B.24}$$

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} 0 \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{B.25}$$

$$\begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} \frac{\omega_y \omega_z (J_y - J_z)}{J_x} \\ \frac{\omega_x \omega_z (J_x - J_z)}{J_y} \\ \frac{\omega_x \omega_y (J_x - J_y)}{J_z} \end{bmatrix} + \begin{bmatrix} \frac{\tau_{q1}}{J_x} \\ \frac{\tau_{q2}}{J_y} \\ \frac{\tau_{q3}}{J_z} \end{bmatrix}. \tag{B.26}$$

The full mathematical model is

$$\begin{cases} \ddot{x} = 2(q_1q_3 + q_0q_2)\frac{f}{m} \\ \ddot{y} = 2(q_2q_3 - q_0q_1)\frac{f}{m} \\ \ddot{z} = -g + (q_0^2 - q_1^2 - q_2^2 + q_3^2)\frac{f}{m} \\ \dot{q}_0 = \frac{1}{2}(-q_1\omega_x - q_2\omega_y - q_3\omega_z) \\ \dot{q}_1 = \frac{1}{2}(q_0\omega_x - q_3\omega_y + q_2\omega_z) \\ \dot{q}_2 = \frac{1}{2}(q_3\omega_x + q_0\omega_y - q_1\omega_z) \\ \dot{q}_3 = \frac{1}{2}(-q_2\omega_x + q_1\omega_y + q_0\omega_z) \\ \dot{\omega}_x = \omega_y\omega_z\frac{J_y - J_z}{J_x} - \frac{J_r}{J_x}\omega_y\Omega + \frac{l}{J_x}\tau_{q_1} \\ \dot{\omega}_y = \omega_z\omega_x\frac{J_y - J_z}{J_x} + \frac{J_r}{J_x}\omega_x\Omega + \frac{l}{J_x}\tau_{q_2} \\ \dot{\omega}_z = \omega_x\omega_y\frac{J_y - J_z}{J_x} + \frac{l}{J_x}\tau_{q_3} \end{cases} . \tag{B.27}$$

# Appendix C

# iLQR Control Law Derivation

Equations (6.14)-(6.19) can be calculated by first substituting Equation (6.8) into (6.4), which yields

$$\delta\mathbf{x}_{k+1} =(I + B_k\mathbf{R}^{-1}B_k^T S_{k+1})^{-1}(A_k\delta\mathbf{x}_k - B_k\mathbf{R}^{-1}B_k^T\nu_{k+1} - B_k\mathbf{u}_k) \tag{C.1}$$

and then substituting Equations (6.11) and (C.1) into (6.8), we get

$$\begin{aligned}S_k\delta\mathbf{x}_k + \nu_k =&\mathbf{Q}\delta\mathbf{x}_k + A_k^T S_{k+1}(I + B_k\mathbf{R}^{-1}B_k^T S_{k+1})^{-1}(A_k\delta\mathbf{x}_k - B_k\mathbf{R}^{-1}B_k^T\nu_{k+1}\\ &- B_k\mathbf{u}_k) + A_k^T\nu_{k+1} + \mathbf{Q}\mathbf{x}_k\end{aligned}$$

$S_k$ and $\nu_k$ can be obtained by applying the matrix inversion as:

$$S_k =A_k^T S_{k+1}(I - B_k(B_k^T S_{k+1}B_k + \mathbf{R})^{-1}B_k^T S_{k+1})A_k + \mathbf{Q}$$

and

$$\begin{aligned}\nu_k =&A_k^T\nu_{k+1} - A_k^T S_{k+1}(I - B_k(B_k^T S_{k+1}B_k + \mathbf{R})^{-1}B_k^T S_{k+1})B_k\mathbf{R}^{-1}B_k^T\nu_{k+1} - A_k^T S_{k+1}\\ &(I - B_k(B_k^T S_{k+1}B_k + \mathbf{R})^{-1}B_k^T S_{k+1})B_k\mathbf{R}^{-1}B_k^T S_{k+1})B_k\mathbf{u}_k + \mathbf{Q}\mathbf{x}_k\end{aligned}$$

By applying $(B_k^T S_{k+1} B_k + \mathbf{R})^{-1} = \mathbf{R}^{-1} - (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} B_k^T S_{k+1} B_k \mathbf{R}^{-1}$, the second term in $\nu_k$ is

$$-A_k^T S_{k+1} B_k (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} B_k^T \nu_{k+1},$$

and the third term is

$$-A_k^T S_{k+1} B_k (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} \mathbf{R}\mathbf{u}_k..$$

By using (6.15), $S_k$ and $\nu_k$ can be written as in (6.18) and (6.19) respectively.

Now, the $\delta\mathbf{u}_k$ control law can be obtained by substituting (6.11) and (C.1) into (6.13) as:

$$\begin{aligned}\delta\mathbf{u}_k = &- (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} B_k^T S_{k+1} A_k \delta\mathbf{x}_k - (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} B_k^T \nu_{k+1} \\ &- (B_k^T S_{k+1} B_k + \mathbf{R})^{-1} \mathbf{R}\mathbf{u}_k,\end{aligned}$$

then $\delta\mathbf{u}_k$ can be written as in (6.14), by replacing the gains in (6.15)-(6.17).

# Bibliography

[1] B. C. Min, C. H. Cho, K. M. Choi, and D. H. Kim, "Development of a micro quad-rotor UAV for monitoring an indoor environment," *Advances in Robotics, Springer-Verlag Berlin Heidelberg*, vol. 5744, pp. 262–271, 2009.

[2] M. B. Srikanth, Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "A robust environment for simulation and testing of adaptive control for mini-UAVs," in *American Control Conference ACC '09*, (Hyatt Regency Riverfront, St. Louis, MO, USA), pp. 5398–5403, 2009.

[3] B. Erginer and E. Altug, "Modeling and PD control of a quadrotor vtol vehicle," in *Proceeding of The 2007 IEEE Intelligent Vehicles Symposium*, (Istanbul, Turkey), pp. 894–899, 2007.

[4] A. Jaimes, S. Kota, and J. Gomez, "An approach to surveillance an area using swarm of fixed wing and quad-rotor unmanned aerial vehicles UAVs," in *IEEE International Conference on System of Systems Engineering SoSE '08*, (Singapore), pp. 1–6, 2008.

[5] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *Proceeding of The 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, (San Diego, CA, USA), pp. 153–158, 2007.

[6] P. T. Nathan, H. A. F. Almurib, and T. N. Kumar, "A review of autonomous multi-agent quad-rotor control techniques and applications," in *2011 4th International Conference on Mechatronics (ICOM)*, (Kuala Lumpur, Malaysia), pp. 1–7, 2011.

[7] U. Pilz, A. Popov, and H. Werner, "Robust controller design for formation flight of quad-rotor helicopters," in *Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, (Shanghai, P.R. China), pp. 8322–8327, 2009.

[8] J. A. Guerrero, P. Castillo, and Y. Challal, "Quadrotors formation control a wireless medium access aware approach," *Journal of Intelligent and Robotic Systems*, vol. 4, pp. 221–231, 2013.

[9] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: Theories and applications," *Control Systems Technology, IEEE Transactions on*, vol. 23, pp. 340–348, Jan 2015.

[10] M. Turpin, N. Michael, and V. Kumar, "Capt: Concurrent assignment and planning of trajectories for multiple robots," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 98–112, 2014.

[11] N. Roy, P. Newman, and S. Srinivasa, *Towards A Swarm of Agile Micro Quadrotors*, pp. 504–. MIT Press, 2013.

[12] T. Nägeli, C. Conte, A. Domahidi, M. Morari, and O. Hilliges, "Environment-independent formation flight for micro aerial vehicles," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1141–1146, Sept 2014.

[13] M. Saska, Z. Kasl, and L. Preucil, "Motion planing and control of formations of micro aerial vehicles," in *Preprints of the 19th World Congress, The International Federation of Automatic Control IFAC*, (Cape Town, South Africa), pp. 1228–1233, 2014.

[14] A. Eskandarpour and V. J. Majd, "Cooperative formation control of quadrotors with obstacle avoidance and self collisions based on a hierarchical mpc approach," in *Robotics and Mechatronics (ICRoM), 2014 Second RSI/ISM International Conference on*, pp. 351–356, Oct 2014.

[15] A. van der Schaft, "$L_2$-gain analysis of nonlinear systems and nonlinear state feedback $H_\infty$ control," *IEEE Transactions on Automatic Control*, vol. 37, no. 6, pp. 770–784, 1992.

[16] M. J. Kim, Y. Choi, and W. K. Chung, "Bringing nonlinear $H_\infty$ optimality to robot controllers," *IEEE Transaction on Robotics*, vol. 31, pp. 682–698, June 2015.

[17] A. Ataka, "Path planning and formation control via potential function for uav quadrotor," *PhD. Thesis, Faculity of Engineering Gadjah Mada University Yogyakarta*, 2014.

[18] G. V. Raffo, M. G. Ortega, and F. R. Rubio, "MPC with nonlinear $H_\infty$ control for path tracking of a quadrotor helicopter," in *The 17th World Congress, The international Federation of Automatic Control*, pp. 8564–8569, July 2008.

[19] G. V. Raffo, M. G. Ortega, and F. R. Rubio, "Backstepping/nonlinear $H_\infty$ control for path tracking of a quadrotor unmanned arrial vehicle," in *American Control Conference (ACC), 2008*, pp. 3356–3361, June 2008.

[20] M. Dalsmo and O. Egeland, "State feedback $H_\infty$ suboptimal control of a rigid spacecraft," *IEEE Transactions on Automatic Control*, vol. 42, no. 8, pp. 1186–1189, 1997.

[21] T. Emanuel and L. Weiwei, "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *2005 American Control Conference (ACC)*, (Portland, OR, USA), pp. 300–306, 2005.

[22] M. Bharatheesha, W. Caarls, W. Wolfslag, and M. Wisse, "Distance metric approximation for state-space rrts using supervised learning," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pp. 252–257, Sept 2014.

[23] H.-j. Zhang, J.-w. Gong, Y. Jiang, G.-m. Xiong, and H.-y. Chen, "An iterative linear quadratic regulator based trajectory tracking controller for wheeled mobile robot," *Journal of Zhejiang University SCIENCE C*, vol. 13, no. 8, pp. 593–600, 2012.

[24] L. Weiwei and T. Emanuel, "Iterative linear quadratic regulator design for nonlinear biological movement systems," in *Informatics in Control, Automation and Robotics (ICINCO), 1st International Conference on*, pp. 222–229, Aug. 2004.

[25] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," in *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 3, pp. 2451–2456 vol.3, Sept 2004.

[26] E. Stingu and F. Lewis, "Design and implementation of a structured flight controller for a 6dof quadrotor using quaternions," in *Control and Automation, 2009. MED '09. 17th Mediterranean Conference on*, pp. 1233–1238, June 2009.

[27] T. Zhang, Y. Kang, M. Achtelik, K. Kuhnlenz, and M. Buss, "Autonomous hovering of a vision/imu guided quadrotor," in *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, pp. 2870–2875, Aug 2009.

[28] M. D. Schmidt, "Simulation and control of a quadrotor unmanned aerial vehicle," in *Master's Thesis, University of Kentucky*, 2011.

[29] S. Gonzalez-Vazquez and J. Moreno-Valenzuela, "A new nonlinear pi/pid controller for quadrotor posture regulation," in *Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2010*, pp. 642–647, Sept 2010.

[30] J. Dvorak, M. Lellis, and Z. Hurak, "Advanced control of quadrotor using eigenaxis rotation," in *2011 IEEE International Conference on Control Applications CCA*, (Denver, CO, USA), pp. 153–158, 2011.

[31] A. Tayebi and S. McGilvray, "Attitude stabilization of a VTOL quadrotor aircraft," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 3, pp. 562–571, 2006.

[32] M. N. Duc, T. N. Trong, and Y. S. Xuan, "The quadrotor mav system using pid control," in *Mechatronics and Automation (ICMA), 2015 IEEE International Conference on*, pp. 506–510, Aug 2015.

[33] I. Sadeghzadeh, A. Mehta, Y. Zhang, and C. Rabbath, "Fault-tolerant trajectory tracking control of a quadrotor helicopter using gain-scheduled pid and model reference adaptive control," in *Annual Conference of the (PHM) Society*, (Monttereal, QC,Canada), 2011.

[34] D. Esteves, A. Moutinho, and J. Azinheira, "Stabilization and altitude control of an indoor low-cost quadrotor: Design and experimental results," in *Autonomous Robot Systems and Competitions (ICARSC), 2015 IEEE International Conference on*, pp. 150–155, April 2015.

[35] E. Fresk and G. Nikolakopoulos, "Full quaternion based attitude control for a quadrotor," in *2013 European Control Conference (ECC)*, pp. 3864–3869, 2013.

[36] R. Zhang, X. Wang, and K.-Y. Cai, "Quadrotor aircraft control without velocity measurements," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pp. 5213–5218, Dec 2009.

[37] M. Schreier, "Modeling and adaptive control of a quadrotor," in *Proceeding of 2012 IEEE International Conference on Mechatronics and Automation ICMA*, (Chengdu, China), pp. 383–390, 2012.

[38] J. F. Guerrero-Castellanos, N. Marchand, A. Hably, S. Lesecq, and J. Delamare, "Bounded attitude control of rigid bodies: Real-time experimentation to a quadrotor mini-helicopter," *Elsevier, Control Engineering Practice*, vol. 19, no. 8, pp. 790–797, 2011.

[39] L. Garcia-Carrillo, E. Rondon, A. Dzul, A. Sanche, and R. Lozano, "Hovering quad-rotor control: A comparison of nonlinear controllers using visual feedback," in *Decision and Control (CDC), 2010 49th IEEE Conference on*, pp. 1662–1667, Dec 2010.

[40] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 2247–2252, April 2005.

[41] S. Zhao, W. Dong, and J. Farrell, "Quaternion-based trajectory tracking control of vtol-uavs using command filtered backstepping," in *American Control Conference (ACC), 2013*, pp. 1018–1023, June 2013.

[42] A. Azzam and X. Wang, "Quadrotor arial robot dynamic modeling and configuration stabilization," in *Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on*, vol. 1, pp. 438–444, March 2010.

[43] D. Cabecinhas, R. Naldi, L. Marconi, C. Silvestre, and R. Cunha, "Robust take-off and landing for a quadrotor vehicle," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 1630–1635, May 2010.

[44] A. Sanchez, V. Parra-Vega, C. Tang, F. Oliva-Palomo, and C. Izaguirre-Espinosa, "Continuous reactive-based position-attitude control of quadrotors," in *American Control Conference (ACC), 2012*, pp. 4643–4648, June 2012.

[45] M. Heidarian and A. Memon, "Attitude control of vtol-uavs," in *2012 UKACC International Conference on Control*, (Cardiff, UK), pp. 363–368, 2012.

[46] A. Tayebi, A. Roberts, and A. Benallegue, "Inertial vector measurements based velocity-free attitude stabilization," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2893–2898, 2013.

[47] J. Guerrero-Castellanos, J. Tellez-Guzman, S. Durand, N. Marchand, and J. Alvarez-Munoz, "Event-triggered nonlinear control for attitude stabilization of a quadrotor," in *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*, pp. 584–591, May 2013.

[48] Y. Al-Younes and M. Jarrah, "Attitude stabilization of quadrotor uav using backstepping fuzzy logic and backstepping least-mean-square controllers," in *Mechatronics and Its Applications, 2008. ISMA 2008. 5th International Symposium on*, pp. 1–11, May 2008.

[49] C. Coza and C. Macnab, "A new robust adaptive-fuzzy control method applied to quadrotor helicopter stabilization," in *Fuzzy Information Processing Society, 2006. NAFIPS 2006. Annual meeting of the North American*, pp. 454–458, June 2006.

[50] W. G. Syed Ali Raza 2010-01-01.

[51] A. Sharma and A. Brave, "Controlling of quadrotor UAV using PID controller and fuzzy logic controller," *International Journal of Electrical, Electronics and Computer Engineering*, vol. 1, no. 2, pp. 38–41, 2012.

[52] M. Santos, V. Lopez, and F. Morata, "Intelligent fuzzy controller of a quadrotor," in *Intelligent Systems and Knowledge Engineering (ISKE), 2010 International Conference on*, pp. 141–146, Nov 2010.

[53] L. S. S. M. S. Abeywardena, D.M.W.; Amaratunga, "A velocity feedback fuzzy logic controller for stable hovering of a quadrotor UAV," in *Industrial and Information Systems (ICIIS), 2009 International Conference on*, pp. 558–562, Dec. 2009.

[54] M. Y. Amir and V. Abbas, "Modeling and neural control of quadrotor helicopter," *Yanbu Journal of Engineering and Science*, vol. 2, pp. 35–49, April 2011.

[55] M. Efe, “Neural network assisted computationally simple $pi^{\lambda}D^{\mu}$ control of a quadrotor UAV,” *Industrial Informatics, IEEE Transactions on*, vol. 7, pp. 354–361, May 2011.

[56] M. K. K. Miller and C. Tomlin, “Path tracking control for quadrotor helicopters,” July 2008.

[57] S. L. W. G. M. Hoffmann and C. J. Tomlin, “Quadrotor helicopter trajectory tracking control,” in *American Institute of Aeronautics and Astronautics, AIAA Guidance, Navigation and Control Conference and Exhibit*, (Honolulu, Hawaii, USA), August 2008.

[58] M. S. D. Mellinger and V. Kumar, “Control of quadrotors for robust perching and landing,” in *International Powered Lift Conference*, (Philadelphia, PA, USA), October 2010.

[59] N. I. A. Eresen and M. O. Efe, “Autonomous quadrotor flight with vision-based obstacle avoidance in virtual environment,” vol. 39, pp. 894–905, 2012.

[60] I. Palunko, R. Fierro, and P. Cruz, “Trajectory generation for swing-free maneuvers of a quadrotor with suspended payload: A dynamic programming approach,” in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 2691–2697, May 2012.

[61] M. Zhaowei, H. Tianjiang, S. Lincheng, K. Weiwei, Z. Boxin, and Y. Kaidi, “An iterative learning controller for quadrotor uav path following at a constant altitude,” in *Control Conference (CCC), 2015 34th Chinese*, pp. 4406–4411, July 2015.

[62] X. Heng, D. Cabecinhas, R. Cunha, C. Silvestre, and X. Qingsong, “A trajectory tracking lqr controller for a quadrotor: Design and experimental evaluation,” in *TENCON 2015 - 2015 IEEE Region 10 Conference*, pp. 1–7, Nov 2015.

[63] Y. Al Younes, A. Drak, H. Noura, A. Rabhi, and A. El Hajjaji, "Model-free control of a quadrotor vehicle," in *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on*, pp. 1126–1131, May 2014.

[64] I. Sa and P. Corke, "Estimation and control for an open-source quadcopter," in *Australasian Conference on Robotics and Automation (ACRA 2011)*, December 2011.

[65] H. Bouadi, M. Bouchoucha, and M. Tadjine, "Sliding mode control based on backstepping approach for an UAV type-quadrotor," *International Journal of Applied Math. and Computer Science*, vol. 4, no. 2, 2007.

[66] L. Carrillo, G. Flores, G. Sanahuja, and R. Lozano, "Quad-rotor switching control: An application for the task of path following," in *American Control Conference (ACC), 2012*, pp. 4637–4642, June 2012.

[67] J. C. Goldin, "Perching using quadrotor with onboard sensing," *Mater Theses, Utah State University*, 2011.

[68] M. Oliveira, "Modeling, identification and control of a quadrotor aircraft," *Master Thesis, Czech Technical University in Prague*, 2011.

[69] D. Cabecinhas, R. Cunha, and C. Silvestre, "A nonlinear quadrotor trajectory tracking controller with disturbance rejection," in *American Control Conference (ACC), 2014*, pp. 560–565, June 2014.

[70] D. Cabecinhas, R. Cunha, and C. Silvestre, "A globally stabilizing path following controller for rotorcraft with wind disturbance rejection," *Control Systems Technology, IEEE Transactions on*, vol. 23, pp. 708–714, March 2015.

[71] Y.-C. Choi and H.-S. Ahn, "Nonlinear control of quadrotor for point tracking: Actual implementation and experimental tests," *Mechatronics, IEEE/ASME Transactions on*, 2014.

[72] T. Madani and A. Benallegue, “Backstepping control for a quadrotor helicopter,” in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 3255–3260, Oct 2006.

[73] P. De Monte and B. Lohmann, “Trajectory tracking control for a quadrotor helicopter based on backstepping using a decoupling quaternion parametrization,” in *Control Automation (MED), 2013 21st Mediterranean Conference on*, pp. 507–512, June 2013.

[74] K. Do and J. Paxman, “Global tracking control of quadrotor vtol aircraft in three dimensional space,” in *Control Conference (AUCC), 2013 3rd Australian*, pp. 26–33, Nov 2013.

[75] A. Honglei, L. Jie, W. Jian, W. Jianwen, and M. Hongxu, “Backstepping-based inverse optimal attitude control of quadrotor,” *International Journal of Advanced Robotic Systems*, vol. 10, no. 223, pp. 1–9, 2013.

[76] V. Parra-Vega, A. Sanchez, and C. Izaguirre, “Toward force control of a quadrotor UAV in se(3),” in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pp. 1802–1809, Dec 2012.

[77] A. Sanchez, V. Parra-Vega, O. Garcia, F. Ruiz-Sanchez, and L. Ramos-Velasco, “Time-parametrization control of quadrotors with a robust quaternion-based sliding mode controller for aggressive maneuvering,” in *Control Conference (ECC), 2013 European*, pp. 3876–3881, July 2013.

[78] A. F. Sorensen, “Autonomous control of a miniature quadrotor following fast trajectory,” *Master Thesis, Aalborg University*, 2010.

[79] S. Barros dos Santos, S. Givigi, and C. Nascimento, “Autonomous construction of multiple structures using learning automata: Description and experimental validation,” *Systems Journal, IEEE*, vol. PP, no. 99, pp. 1–12, 2015.

[80] J. Guadarrama-Olvera, H. Rodriguez-Cortes, and R. Castro-Linares, "Robust trajectory tracking control of a quadrotor helicopter," in *Control Conference (ECC), 2014 European*, pp. 908–913, June 2014.

[81] S. Islam, P. Liu, and A. El Saddik, "Robust control of four-rotor unmanned aerial vehicle with disturbance uncertainty," *Industrial Electronics, IEEE Transactions on*, vol. 62, pp. 1563–1571, March 2015.

[82] G. Vasan, A. K. Singh, and K. M. Krishna, "Model predictive control for micro aerial vehicle systems (MAV) systems," *CoRR*, vol. abs/1412.2356, 2014.

[83] M. Beul, R. Worst, and S. Behnke, "Nonlinear model-based position control for quadrotor uavs," in *ISR/Robotik 2014; 41st International Symposium on Robotics; Proceedings of*, pp. 1–6, June 2014.

[84] D. Zhou and M. Schwager, "Vector field following for quadrotors using differential flatness," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 6567–6572, May 2014.

[85] O. Fritsch, P. De Monte, M. Buhl, and B. Lohmann, "Quasi-static feedback linearization for the translational dynamics of a quadrotor helicopter," in *American Control Conference (ACC), 2012*, pp. 125–130, June 2012.

[86] T. Lee, M. Leok, and N. McClamroch, "Nonlinear robust tracking control of a quadrotor UAV on se(3)," in *American Control Conference (ACC), 2012*, pp. 4649–4654, June 2012.

[87] T. Lee, "Robust adaptive attitude tracking on so(3) with an application to a quadrotor UAV," *Control Systems Technology, IEEE Transactions on*, vol. 21, pp. 1924–1930, Sept 2013.

[88] E. Reyes-Valeria, R. Enriquez-Caldera, S. Camacho-Lara, and J. Guichard, "Lqr control for a quadrotor using unit quaternions: Modeling and simulation," in *2013 IEEE International Conference on Electronics, Communications and Computing (CONIELECOMP)*, (Cholula), pp. 172–178, 2013.

[89] P. Castillo, A. Dzul, and R. Lozano, “Real-time stabilization and tracking of a four-rotor mini rotorcraft,” *Control Systems Technology, IEEE Transactions on*, vol. 12, pp. 510–516, July 2004.

[90] W. C. T. Sangyam, P. Laohapiengsak and I. Nilkhamhang, “Path tracking of UAV using self-tuning PID controller based on fuzzy logic,” in *SICE Annual Conference*, (The Grand Hotel, Taipei, Taiwan), pp. 1265–1269, August 2010.

[91] F. Yacef, O. Bouhali, and M. Hamerlain, “Adaptive fuzzy tracking control of unmanned quadrotor via backstepping,” in *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pp. 40–45, June 2014.

[92] H. Lee and H. J. Kim, “Robust control of a quadrotor using takagi-sugeno fuzzy model and an LMI approach,” in *Control, Automation and Systems (ICCAS), 2014 14th International Conference on*, pp. 370–374, Oct 2014.

[93] M. Zareb, R. Ayad, and W. Nouibat, “Fuzzy-PID hybrid control system to navigate an autonomous mini-quadrotor,” in *3rd International Conference on Systems and Control*, pp. 906–913, Oct 2013.

[94] D. Gautam and C. Ha, “Control of a quadrotor using a smart self-tuning fuzzy PID controller,” *International Journal of Advanced Robotic Systems*, vol. 10, no. 380, pp. 1–9, 2013.

[95] Y. Wang, Y. Chenxie, J. Tan, C. Wang, Y. Wang, and Y. Zhang, “Fuzzy radial basis function neural network pid control system for a quadrotor uav based on particle swarm optimization,” in *Information and Automation, 2015 IEEE International Conference on*, pp. 2580–2585, Aug 2015.

[96] A. Dib, N. Zaidi, and H. Siguerdidjane, “Robust control and visual servoing of an UAV,” in *The 17th World Congress, The international Federation of Automatic Control*, pp. 5730–5735, July 2008.

[97] A. Chamseddine, Y. Zhang, C. Rabbath, C. Join, and D. Theilliol, "Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 48, pp. 2832–2848, October 2012.

[98] J. Ferruz, V. M. Vega, A. Ollero, and V. Blanco, "Reconfigurable control architecture for distributed systems in the hero autonomous helicopter," *Industrial Electronics, IEEE Transactions on*, vol. 58, no. 12, pp. 5311–5318, December 2011.

[99] T. Dierks and S. Jagannathan, "Neural network control of quadrotor UAV formations," in *American Control Conference ACC '09*, (Hyatt Regency Riverfront, St. Louis, MO, USA), pp. 2990–2996, 2009.

[100] G. Damien, "Unmanned aerial vehicle formation flight using sliding mode disturbance observers aerial vehicles," in *Thanh Mung Lam (Ed.),ISBN: 978-953-7619-41-1, InTech*, 2009.

[101] D. Lee, "Distributed backstepping control of multiple thrust-propelled vehicles on balanced graph," in *18th IFAC Congress*, (Milano, Italy), pp. 8872–8877, 2011.

[102] G. Regula and B. Lantos, "Formation control of quadrotor helicopters with guaranteed collision avoidance via safe path," *Electrical Engineering and Computer Science*, vol. 4, no. 2012, pp. 113–124, 2013.

[103] I. Bayezit and B. Fidan, "Distributed cohesive motion control of flight vehicle formations," *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 12, pp. 5763–5772, December 2013.

[104] H. Cai and J. Huang, "The leader-following consensus of multiple rigid spacecraft systems," in *American Control Conference ACC '13*, (Washington, DC, USA), pp. 824–829, 2013.

[105] A. Abdessameud, A. Tayebi, and I. G. Polushin, "Motion coordination of thrust-propelled underactuated vehicles in the presence of communication

delays," in *Preprints of the 19th World Congress, The International Federation of Automatic Control IFAC*, (Cape Town, South Africa), pp. 3170–3175, 2014.

[106] F. Poiesi and A. Cavallaro, "Distributed vision-based flying cameras to film a moving target," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 2453–2459, Sept 2015.

[107] A. Abdessameud and A. Tayebi, "Formation control of vtol-uavs," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pp. 3454–3459, Dec 2009.

[108] A. Karimoddini, M. Karimadini, and H. Lin, "Decentralized hybrid formation control of unmanned aerial vehicles," *arXiv preprint arXiv:1403.0258*, no. 1, 2014.

[109] L. C. A. Pimenta, G. A. S. Pereira, N. Michael, R. Mesquita, M. M. Bosque, L. Chaimowics, and V. Kumer, "Swarm coordination based on smoothed particle hydrodynamics technique," *IEEE Transaction on Robotics*, vol. 29, no. 2, pp. 383–399, 2013.

[110] M. Turpin, N. Michael, and V. Kumar, "Decentralized formation control with variable shapes for aerial robots," in *2012 IEEE International Conference on Robotics and Automation*, (RiverCentre, Saint Paul, Minnesota, USA), pp. 23–30, 2012.

[111] M. Turpin, N. Michael, and V. Kumar, "Trajectory design and control for aggressive formation flight with quadrotors," *Autonomous Robots*, vol. 33, no. 1-2, pp. 143–156, 2012.

[112] A. Bemporad and C. Rocchi, "Decentralized hybrid model predictive control of a formation of unmanned aerial vehicles," in *18th IFAC World Congress*, (Milano, Italy), 2011.

[113] N. Koksal, B. Fidan, and K. Buyukkabasakal, "Real-time implementation of decentralized adaptive formation control on multi-quadrotor systems," in *Control Conference (ECC), 2015 European*, pp. 3162–3167, July 2015.

[114] K. Ghamry and Y. Zhang, "Formation control of multiple quadrotors based on leader-follower method," in *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, pp. 1037–1042, June 2015.

[115] Y. X. Min, K.-C. Cao, H. H. Sheng, and Z. Tao, "Formation tracking control of multiple quadrotors based on backstepping," in *Control Conference (CCC), 2015 34th Chinese*, pp. 4424–4430, July 2015.

[116] F. B. Choi and D. Lee, "Teleoperation control of formation among multiple under-actuated quadrotor UAVs," *2013 IEEE RO-MAN, The 22nd IEEE International Symposium on Robot and Human Interactive Communication*, no. 3, pp. 306–307, 2013.

[117] R. Tjahjana and P. Sasongko, "Multi flying vehicle control for saving fuel," *Applied Mathematical Sciences*, vol. 8, no. 15, pp. 733–742, 2014.

[118] B. Yu, X. Dong, Z. Shi, and Y. Zhong, "Formation control for quadrotor swarm systems: Algorithms and experiments," in *32nd Chines Control Conference CCC*, (Xi'an, China), pp. 7099–7104, 2013.

[119] A. Mancini, A. Benini, E. Frontoni, P. Zingaretti, and S. Longhi, "Coalition formation for unmanned quadrotors," in *Volume 3: 2011 ASME/IEEE International Conference Mechatroncs and Embedded Systems and Applications, Parts A and B*, (Washington, DC, USA), pp. 745–752, 2011.

[120] L. Rosa, "Elective in robotics, asctec hummingbird quadrotor," in *Sapienza University Di Roma*, 2011.

[121] I. Digi International, "Xbee/xbee-pro 802.15.4 module," in *http://www.digi.com*, 2011.

[122] D. Mellinger, "Trajectory generation and control for quadrotors," *PhD Dissertation, University of Pennsylvania*, 2012.

[123] H. J. Marquez, *Nonlinear control systems.* Hoboken, New Jersey, 2003.

[124] J. Betts, *Practical methods for optimal control and estimation using nonlinear programming*, vol. 19. SIAM, 2009.

[125] B. Stevens and F. Lewis, *Aircraft Control and Simulation.* Hoboken, New Jersey: John Wiley and Sons,Inc., 2nd edition, 2003.

[126] R. Beard, “Quadrotor dynamics and control,” *Brigham Young University Report*, 2008.

[127] J. Kuipers, “Quaternions and rotation sequences,” in *Geometry, Integrability and Quantization, Ivailo M. Mladenov and Gregory L. Naber, Editors, Coral Press, Sofia 2000*, (Vama, Bulgaria), pp. 127–143, 1999.

[128] J. Diebel, “Representing attitude: Euler angles, unit quaternions, and rotation vectors,” *Stanford University Report*, 2006.