Optimal Control of Multiple Quadrotors for Transporting a Cable Suspended Payload

By

Yaser N. Ibrahem Alothman

Abstract

In this thesis, the main aim is to improve the flight control performance for a cable suspended payload with single and two quadrotors based on optimised control techniques. The study utilised optimal controllers, such as the Linear Quadratic Regulator LQR, the Iterative based LQR (ILQR), the Model Predictive Control MPC and the dynamic game controller to solve tracking control problems in terms of stabilisation, accuracy, constraints and collision avoidance. The LQR control was applied to the system as the first control method and compared with the classical Proportional-Derivative controller PD. It was used to achieve the load path tracking performance for single and two quadrotors with a cable slung load. The second controller was ILQR, which was developed based on the LQR control method to deal with the model nonlinearity. The MPC technique was also applied to the linearised nonlinear model LMPC of two quadrotors with a payload suspended by cables and compared with a nonlinear MPC (NMPC). Both MPC controllers LMPC and NMPC considered the constraints imposed on the system states and control inputs. The dynamic game control method was developed based on an incentive strategy for a leader-follower framework with the consideration of different optimal cost functions. It was applied to the linearised nonlinear model. Selecting these control techniques led to a number of achievements. Firstly, they improved the system performance in terms of achieving the system stability and reducing the steady-state errors. Secondly, the system parameter uncertainties were taken into consideration by utilising the ILQR controller. Thirdly, the MPC controllers guaranteed the handling of constraints and external disturbances in linear and nonlinear systems. Finally, avoiding collision between the leader and follower robots was achieved by applying the dynamic game controller. The controllers were tested in MATLAB simulation and verified for various desired predefined trajectories. In real experiments, these controllers were used as high-level controllers, which produce the optimised trajectory points. Then a low-level controller (PD controller) was used to follow the optimised trajectory points.

Publication List

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

1. 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.

2. Y. Alothman and D. Gu, "Quadrotor Transporting Cable-suspended Load Using Iterative Linear Quadratic Regulator (ILQR) Optimal Control." Computer Science and Electronic Engineering (CEEC), 2016 8th. IEEE, 2016.

3. Y. Alothman, M. Guo, and D. Gu,"Using Iterative LQR to Control Two Quadrotors Transporting a Cable-suspended Load," Proceeding of IFAC, vol. 50, no. 1, pp. 4324-4329, 2017.

4. Y. Alothman and D. Gu,"Using Constrained Model Predictive Control to Control Two Quadrotors Transporting a Cable-Suspended Payload," WCICA-PapersOnLine, July 4-7, Changsha, China 2018.

5. Y. Alothman and D. Gu,"Using Constrained NMPC to Control a Cable-Suspended Payload With Two Quadrotors," The 24th IEEE International Conference on Automation and Computing (ICAC'18), September 6-7, Newcastle,UK, 2018.

6. Y. Alothman and D. Gu,"Incentive Stackelberg Dynamic Game Approach to Transporting a Cable-Suspended Load with Two Quadrotors," in 10th Computer Science and Electronic Engineering Conference (CEEC 2018), September 19-21, Colchester,UK, 2018.

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 heartfelt gratitude goes to my supervisor Prof. Dongbing Gu for his continuous guidance and valuable feedback.

My utmost appreciation goes to my mother, father and siblings, who, despite being so far away, have supported me in every step of the way through their immense love and care.

I am deeply indebted to her, without whom everything I have made would have seemed meaningless, my dear wife. She has given me all the love and relieved all the pain.

Last but not least, my thanks are reserved for all my friends and colleagues at the University of Essex. Every one of you has marked this journey with an unforgettable contribution.. . .

Contents

List of Figures

List of Tables

Abbreviations

AscTec.	Ascending Technology
LQR	Linear Quadratic Regulator
ILQR	Iterative Linear Quadratic Regulator
MPC	Model Predictive Control
LMPC	Linear Model Predictive Control
NMPC	Nonlinear Model Predictive Control
PD	Proportional Derivative
UAVs	Unmanned Aerial Vehicles
SMC	Sliding Model Control
PID	Proportional Integral Derivative
MFC	Model Free Control
ILQG	Iterative Linear Quadratic Gaussian
PWA	Piecewise Affine
LBMPC	Learning Base Model Predective Control
GPC	Generalized Predictive Control
ESO	Extended State Observer
ATM	Air Traffic Management
MAVs	Micro Air Vehicles
SCSP	small cells service provider
MSP	Microcell provider
COP	Constrained Optimisation Problem
HEVs	Hybrid Electric Vehicles
V2V	Vehicle 2 Vehicle
RMSE	The Root Mean Square Error

Symbols

$\boldsymbol{S_e} : \boldsymbol{x_e y_e z_e}$	Inertial frame
$\boldsymbol{S^K} : \boldsymbol{xyz}$	Intermediate frame: translation from $\boldsymbol{S_e}$ to the center of the $\boldsymbol{K}$th quadrotors $\boldsymbol{K} = 1, 2$
$\boldsymbol{S_b^K} : \boldsymbol{x_b^K y_b^K z_b^K}$	Body-fixed frame for quadrotor,
$\boldsymbol{E_\mathcal{I}} \in \mathbb{R}^3, \mathcal{I} = 1, 2, 3$	Unit orthogonal vectors of $\boldsymbol{S_e}$
$\boldsymbol{e_\mathcal{I}^K} \in \mathbb{R}^3, \mathcal{I} = 1, 2, 3$	Unit orthogonal vectors of $\boldsymbol{S_b^K}$
$\boldsymbol{\eta^K} = \left[\phi^K, \theta^K, \psi^K\right]^T \in \mathbb{R}^3$	Euler angles of quadrotor defined in $Z - Y - X$
$\boldsymbol{T_{e2b}^K} \in \mathbb{R}^{3 \times 3}$	Transformation matrix from $\boldsymbol{S_e}$ to $\boldsymbol{S_b^K}$
$\boldsymbol{\Omega^K} \in \mathbb{R}^3$	Angular velocity of quadrotor in $\boldsymbol{S_b^K}$
m_Q^K	Mass of the quadrotor
m_P	Mass of the payload
$\boldsymbol{I_Q^K} \in \mathbb{R}^3$	Inertial matrix of the quadrotor with respect to $\boldsymbol{S_b^K}$
$\boldsymbol{\xi_Q^K} \in \mathbb{R}^3$	Position of the center of quadrotor in $\boldsymbol{S_e}$
$\boldsymbol{\xi_P} \in \mathbb{R}^3$	Position of the payload in $\boldsymbol{S_e}$
x_P, y_P, z_P	Three elements of $\boldsymbol{\xi_P}$
L_r^K	Length of the rope
L_Q^K	Length of the quadrotor arm
$\alpha^K, \beta^K \in \mathbb{R}$	Angles of the rope with respect to $\boldsymbol{S^K}$
$\boldsymbol{\rho^K} \in \mathbb{R}^3$	Unit vector from the payload to the attached point

K, K_v, K_u	Positive proportional gains
$\boldsymbol{f}_{iz}^{\boldsymbol{K}}, \boldsymbol{f}_{ih}^{\boldsymbol{K}} \in \mathbb{R}^3$	Vertial and horizontal forces generated
	by ith propeller, $i = 1, 2, 3, 4$
$k_F, k_M \in \mathbb{R}$	Propeller aerodynamic parameters
$\boldsymbol{F}_Q^{\boldsymbol{K}} = \left[0, 0, F_z^{\boldsymbol{K}}\right]^T$	Thrusts of two quadrotors in $\boldsymbol{S}_{\boldsymbol{b}}^{\boldsymbol{K}}$
$\boldsymbol{M}_Q^{\boldsymbol{K}} = \left[M_x^{\boldsymbol{K}}, M_y^{\boldsymbol{K}}, M_z^{\boldsymbol{K}}\right]^T$	Torques of two quadrotors in $\boldsymbol{S}_{\boldsymbol{b}}^{\boldsymbol{K}}$
$\mathbf{Q}$	The generalised forces
C	Cosine function
S	Sine function
Q and Q_f	Semi-definite symmetric positive matrices
	state and final cost weighting matrices
R	Symmetric positive definite control cost matrix
$\boldsymbol{x}_N$	The final step of each state
$\boldsymbol{x}^*$	The target state
$\boldsymbol{u}^*$	The improved nominal control
$\delta\lambda_{\boldsymbol{k+1}}^T$	Lagrange multiplier
$S_{\boldsymbol{k}} and \mathcal{S}_{\boldsymbol{k}}$	Symmetric cross product matrix
$\delta\boldsymbol{u}_{\boldsymbol{k}}$ and $\delta\boldsymbol{x}_{\boldsymbol{k}}$	control and state deviations
$\mathbb{X}$	State constraints
$\mathbb{U}$	Control input constraints
$\boldsymbol{u}_{\boldsymbol{1}}^{\boldsymbol{t}}$ and $\boldsymbol{u}_{\boldsymbol{2}}^{\boldsymbol{t}}$	leader and follower team optimal control

Chapter 1

Introduction

1.1 Introduction

Autonomous flying objects have been the subject of extensive research in control systems of Unmanned Aerial Vehicle (UAV) in terms of the latter's cost and efficient design. The quadrotors have been designed in a way that provides them with effective prioritisation to perform many applications safely and precisely compared with fixed-wing UAVs. The advantages of these quadrotors, in comparison with those of fixed wing plane, are presented by position and orientations in terms of their capability to perform specific actions in a limited space, such as taking off, landing and hovering vertically over a static point or dynamic targets [1], [2]. This makes these quadrotors best able to perform many complicated tasks with various applications. Among these applications are critical situations in civil and military fields, such as dam cracks, oil and power lines inspections, injured soldiers rescue from a war field, fire rescue, mail transport, and medical transportation such as blood, kidneys, hearts and emergency products between hospitals. These quadrotor applications are still in progress due to the fields' increasing requirements worldwide, which has encouraged the development of the system technology and control automation protocols [3], [4].

Furthermore, utilising cable suspended payloads with single and two quadrotors has advantages over the individual quadrotor in that they implement their transportation missions that cannot be performed with an individual quadrotor. Moreover, they achieve their tasks with high performance in terms of system stability and high accuracy, while a minimum number of sensors are utilised to disseminate them to the quadrotors [5], [6],[7]. This progress is presented in many ways such as localisation achievement, obstacles avoidance, path planning, and control design. Research has been mainly focused on controller development in the quadrotors carrying a payload with cables autonomously. This field is still facing challenges in carrying load controlled by cables with single and two quadrotors. These challenges have persisted because of the modelling complexities due to system structure design. The system dynamic equations are represented in four input forces and six output states with the system under actuation. Moreover, there have been considerable difficulties in utilising Lagrange equation considering suspended payload and control estimation architecture, and limitations in communication are not uncommon. In the past few years, researchers have been motivated to design and develop quadrotors' components, such as sensors, electronics and other parts in terms of reducing cost value, size and weight.

Transporting a suspended load with two quadrotors has advantages over utilising a quadrotor with carrying a load. While autonomous flying control of a cable suspended payload with a quadrotor has several restrictions, such as large swing angles, system high nonlinearity and constrained environments, the control problems of multiple quadrotors carrying a payload by cables include the initial equilibrium point, communication delay between two quadrotors, large ropes angles and stability, as addressed by [8], [9], in addition to avoiding collisions between the quadrotors while maintaining the operational point and transporting the load, as well as avoiding immobile or moving obstacles. To the best of my Knowledge, this is the first use of two quadrotors with a suspended load by cables. In this thesis, the focus is on control design of the single, two quadrotors with suspended payload by cables, while avoiding collisions between two quadrotors. Furthermore, the study examines the possibility of maintaining the distance between them through

selecting a proper operational point. The first scenario introduces a single quadrotor and a suspended payload with its centre of gravity by a cable length $1m$. The control architecture is applied on the slung load to track a desired trajectory. This controller is implemented in order to prevent large angles of the load resulting from payload swinging during hovering and transporting tasks. The controller aims are comprised of different main goals to improve the required results. This can be summarised as realising a small error, gaining better performance, achieving stability, avoiding collisions, in addition to having constraints and robustness. Advanced control theory is reviewed in different types which have been used to control the autonomous tasks.

In the second scenario, however, two quadrotors are carrying a payload by cables. Both quadrotors have a centralised controller considering the suspended load while keeping the desired safe distance between them by relying on the equilibrium point. This operational position is considered to be an initial stage to proceed to the next stage, which relies on the selected appropriate swing angle values. Two quadrotors with the payload collaboratively maintain the system formation based on eight control inputs' forces, 26 states' positions and orientations in 13 degrees of freedom.

In the third scenario, the leader-follower quadrotors with the suspended load by cables are almost similar to the case in the second approach. One of the quadrotors functions as a leader quadrotor while the other vehicle is the follower. The leader quadrotor is in charge of tracking a reference trajectory, whereas the follower quadrotor is responsible for preserving small swing angles to achieve load stability. Moreover, both quadrotors keep a certain distance between each other in the same plane. In addition, both quadrotors have their own control input in the same architecture, where each control input is affected by the other but with a separate decision-making ability because of the values for their weight matrices. Each vehicle has its own flexibility in terms of considering the state, but increasing the mean of the follower control weight and decreasing the leading one can minimise the cost when achieving the system performance.

In order to guarantee the performance, an efficient controller must be proposed to handle the constraints and then achieve the system stability while transporting the suspended load. Both accuracy and stability are the important performance to be considered in this thesis. Therefore, in this thesis many control techniques are addressed for the performance in the MATLAB simulation and C++ experiment. Firstly, the LQR controller has been nominated in this thesis due to the high demand for flight control to handle the system optimal performance by achieving stability, and capturing accuracy with minimum cost value. Employing the optimal LQR controller has several advantages. First, this controller is able to tune the weight parameters easily compared with the classical PID control, which leads to determining the better parameters. Second, in the LQR optimal control objective, the performance criterion is included to test the system efficiency. Due to these reasons, this controller is developed to stabilise the suspended payload and minimise the swing angles and quadrotor attitudes while tracking a predefined trajectory by determining the best ratio between the states cost and control input cost via the Lyapunov function. Another advantage is that this controller overcomes the model nonlinearity by performing the optimisation recursively. For this reason, an optimal iterative linear quadratic regulator ILQR controller is suggested based on the LQR control approach.

The optimal ILQR control is applied to the nonlinear dynamic model and then it is linearised at each sampling time with an operational point. The main feature of this controller is that it recursively performs the tracking control at each sample in order to find the best optimised payload position, reduce swing angles and achieve the system stability while the quadrotors orientations are being considered.

In addition, a constrained model predictive optimal control MPC is chosen then applied to two quadrotors with suspended payload via cables in both linear and nonlinear dynamic model representations. The beneficial points of the optimal MPC are focused on handling the states and control inputs constraints, optimising the tracking problem for all time steps along the predefined path, and improving the system performance. Finally, a cooperative incentive dynamic game controller is designed for the leader-follower technique with a non-centralised control scenario,

where each quadrotor makes its own decision based on its knowledge from the other. This provides the system with formation change due to the individual control decision and improves its tracking accuracy, as well as avoiding collision. All these aforementioned controllers are tested utilising simulations and evaluated indirectly in experiments via planning the trajectory. The control design depends on the system model. The dynamic model is a high-level complex model and is presented based on using the Euler angles technique.

The pitch and roll angles can be represented in a way that evaluates the infinite value (singularity situation), which comes from the pitch angle divided by zero for a single quadrotor. In the case of two quadrotors with a suspended load, however, there is no need to consider avoiding singularity due to the model formation since the payload is not in the centre of gravity for either quadrotor. The dynamic model is comprised of all the states' parameters and control input including payload position, swing angles and quadrotor's orientation, while the control input includes forces and moments which can affect one another while achieving the transportation task.

In 2003 the Ascending Technology company was established by a group of students in Germany and started the first step to build a simple model of the quadrotor. This quadrotor is called Hummingbird and consists of the low-level microcontroller which is in charge of the stabilisation of its attitude and orientation while the high-level microcontroller is responsible for the quadrotor's position (see Chapter 7). Moreover, this quadrotor is able to carry a payload with limited weight and it has been used in this thesis (see Figures 1.1 and 1.2), where the ropes are used as cables and the payload is a car toy.
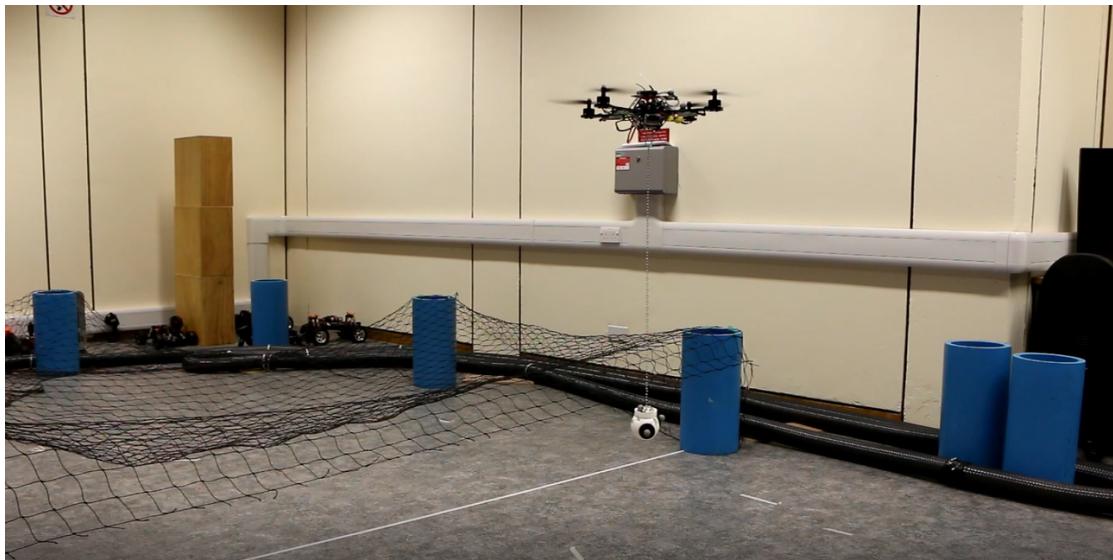
FIGURE 1.1: Single quadrotor with a suspended payload with

FIGURE 1.2: Two quadrotors carrying payload with cables

1.2   Motivations

Recently, a cable suspended payload with the quadrotor model has achieved significant growth due to the high demand on single and cooperative two quadrotors to handle the transporting applications that the naked quadrotors cannot be able to achieve. Thus, the researchers have been focusing on a control design to control a payload's stable or moving position and swing angles and the quadrotor's orientations in order to improve the system performance so that the tasks can be achieved efficiently. The control of cable suspended load with a quadrotor can solve the problem of reducing the swing angles, achieving the system stability and preserving the two quadrotors while implementing their transportation.

In this study, the advantages are taken by the permissibility of increasing the weight, sensors sharing, formation maintaining and energy compatibility. The control design of single, two quadrotors transporting a cable suspended load are introduced. First, a single quadrotor is presented to improve the controllability of accurate transportation to address the previous limitations and difficulties. Second, a control system is constructed which is comprised of multiple quadrotors connected with a rigid body suspended by links to perform transportation of payload. These controllers detect the errors to improve the required performance in order to promote and realise the final goal. This thesis focuses on developing an optimal controller in order to improve the performance of the non-linear dynamic model through achieving quadrotor-payload position and orientation stabilisation during the transportation task. In the case of leader-follower quadrotors with a suspended load, one of the quadrotors is selected as a leader to track a desired predefined trajectory while the second is considered as a follower quadrotor to follow the leader. Moreover, all the three relative distances represented by two ropes' lengths and the relative space between the leader-follower quadrotors are maintained according to payload positions, swing angles and leader-follower orientations, where the computer gives the reference trajectory position information to both the leader and the follower quadrotors to be tracked by the payload. Each leader-follower behaviour can be obtained based on the relative weighted control

action in order to maintain the system formation, collision avoidance and accuracy achievement.

In this thesis, the control design for a cable suspended payload with a single, two quadrotors is introduced to achieve the system transportation performance. This is achieved through a reducing steady-state errors, making the payload track a predefined path, stabilising the quadrotors' orientations, considering constraints, and avoiding collision. The efficient control design for the cable suspended payload with quadrotors relies on the quadrotors-load aerodynamics, which includes a number of rotors and the payload's weight, and position.

1.3   Thesis Scope

The control of the quadrotors with a suspended payload is a challenging task. Firstly, the location of the quadrotors and the payload should be known and provided to the control system. However, obtaining the location information in nature environments is still an unsolved problem. there are various algorithms using different sensors to the problem, but they are far from being employed in transportation applictions where high accuracy and reliability are required. The work in this thesis assumes the location information is available and in our experiment the system operates in indoor environment where a vision tracking system (Vicon) is available to provide the required information. Secondly, both system parameter uncertainty and external environment disturbances are not considered in this thesis. With the consideration of these uncertainties, it would definitly make gains on the system robustness. But they are out of the scope of this thesis, and left for future work. Thirdly, the operating environment is assumed to be static, no moving obstacles. Only the collision between two quadrotors and payload are consideredin the cost function. This thesis is focuses on the design of the controllers to control the system. The stability, optimal cost and constraints are the main factors to be investigated.

1.4   Contributions

This thesis is focused on four optimisation controllers, which are LQR, ILQR, MPC and dynamic game controllers. The substantial contributions in this work are the development of a dynamic model for cable suspended payload with two quadrotors and the implementation of the aforementioned controllers in simulation and experimental tests. The thesis contributions are summarised in the following points:

1. Two dynamic models have been established based on Lagrange equation, and these models are presented by (i) a cable suspended load with a single quadrotor and (ii) two quadrotors carrying a payload by cables. In the second model, each constant state and control input is taken into consideration to guarantee a satisfying operational point. As these models are in high level of complexity, and a linearisation procedure is executed based on a nonlinear model system to be convenient for the optimised controllers.

2. An LQR controller is developed for the systems as one of the contributions made in this thesis. This controller is to show less state errors can be achieved for the systems compared with a popular PD controller. These errors are clarified as a payload position RMSE values and improved to be 0.0123(m) for LQR controller compared with 0.086(m) for PD controller. It is implemented by linearising the nonlinear system models and applying the standard LQR control technique.

3. Due to the high nonlinearity of the dynamic models, it is necessary to handle them in order to improve the system performance. Since a recursive technique is required, an ILQR is developed for the control problem of a cable suspended payload with quadrotors according to the LQR approach. The main advantage of utilising the ILQR controller is to eliminate the high nonlinearity of the dynamic system while tracking the desired trajectory. Moreover, provide less RMS load position errors while tracking the trajectory based on the iterative technique and reach 0.0026(m).

4. A model predictive controller is developed and performed via MATLAB simulation within two approaches, linear MPC and nonlinear MPC. They are used to deal with the constraints of the states and inputs, which are caused by the limitations of environments and the motor powers. The optimisation algorithm QP and YALMIP are used inside of the linear and nonlinear MPC controllers. The control performance has been ameliorated to demonstrate a worthy reduction of the RMSE to 0.0012(m) for the payload position.

5. A dynamic game controller has been designed using an incentive strategy to solve the Leader follower dynamic control problem, where one of the quadrotors is involved as a guide leader and the second to follow the leader while both carrying a payload by cables. While the leader keeps the load tracking the predefined path and achieving load position and stability and the leader's attitudes, the follower is kept following the leader and maintaining the payload position, reducing the errors of the state and controlling the follower's orientations. It is worth noting that both the leader and the follower consider reducing the swing angles. The advantage of using the leader-follower method is its ability to determine the best control parameters suitable for the required situation and avoid the collision. Furthermore, this controller was enhanced by improving the RMS errors of the load position along the desired path.

1.5 Thesis Structure

The remaining structure of this thesis is as follows:

Chapter 2: reviews various control techniques for single and multiple quadrotors with suspended payload. A full coverage of the state of the art linear and nonlinear control techniques used for payload transportation system are provided.

Chapter 3: describes the dynamic models of cable suspended payload with single and two quadrotors using Lagrange equation based on Euler angles approach. The nonlinear derivation equations are provided in detail, then the state space

matrices are evaluated, considering 16 feedback states and eight degrees of freedom for a single quadrotor with the slung load; whereas in the significant cable suspended payload using two quadrotors is outlined for thirteen degrees of freedom and presents 26 system states. Lastly, the nonlinear dynamic model is linearised in order to prepare for linear controller design.

Chapter 4: includes a description of two optimal LQR and ILQR control methods, where the ILQR control technique is developed based on the LQR controller. This control technique is verified to stabilise the payload position and reduce swing angles and quadrotors' orientations by utilising MATLAB simulation. Then the results are provided for both models and compared with those of the LQR control method.

Chapter 5: reviews the linear and nonlinear predictive optimal MPC control approaches with their simulation results. The optimal MPC control technique is implemented using a Matlab simulation for two quadrotors with a load suspended by cables according to the control horizon. This controller is used on both the nonlinear Euler angles presentation and the linearised model to handle the system constraints, which are involved in the control design. Furthermore, the chapter discusses the achievement of the system attitudes stability, load position accuracy and the control tracking problem for this high nonlinearity model.

Chapter 6: presents the dynamic game control approach based on the derivation of its equations followed by the stability results for leader-follower with the suspended payload. This controller is built and developed depending on the incentive principle. In other words, each leader and follower quadrotor can be able to achieve its stability, swing angles and load stability based on its knowledge of the other's information. Avoiding collisions in a strict environment is made while both the leader and the follower make a convenient team optimal decision.

Chapter 7: presents the real hummingbird quadrotors with their accessories, payload, communication Xbee and Vicon system tracker. The experimental indoor

tests are based on a real system by utilising PID to follow the optimised trajectory provided from the MATLAB simulator and using an optimised controller compared with the classical PID control.

Chapter 8: concludes the findings and summarises the results obtained in the previous chapters. Moreover, the chapter includes suggestions for future work.

Chapter 2

Literature Review

2.1 Introduction

Transporting a cable suspended payload with quadrotors has been addressed as the main area for control tasks. Flight control techniques have recently utilised four-rotor UAVs to achieve stability for system position and orientations. The behaviour quality of the UAVs has been approved by implementing several classical and modern control systems. The proposed controllers in this survey are presented to eliminate the dynamic model uncertainty, lack of stability and limitations regarding structure complexities in the controller. This survey is divided into four categories. It starts from the nonlinear control of single and multiple UAVs with suspended payload. Then it moves on to optimal control methods, including LQR, MPC and game controllers. The survey provides the information on how to apply these optimal controllers in various robotic applications.

2.2 UAV Payload Transportation

UAVs, have become common aerial robotics for researchers and they have been implemented in different applicable situations. Furthermore, research into this area has witnessed significant growth to the development of control theory applications.

The progress in this field has included simplicity of mechanical design, position correction, dynamic modelling and exerted power as real strong steps for the future of control. In [10] a robust controller was proposed using H_∞ nonlinear optimal approach to solve the attitude control problems for quadrotors. A Lyapunov function was selected and a state feedback controller was derived based on quadrotor attitude dynamics. Quadrotor attitude was represented by unit quaternion and disturbance was considered.

Position stabilisation control methods for quadrotors based on vision feedback have been proposed. A nested saturation control technique was introduced in [11] which improved smoothing behaviour of UAV and energy conservation compared with backstepping and sliding mode techniques when the authors tested it in real time to stabilise the UAV.

In [12] a simulation was presented to stabilise the take-off and landing tasks of a single quadrotor UAV using the Newton-Euler technique of backstepping PID together with PD controllers. The best simulation results came from the nonlinear backstepping based on PID control in comparison with the linear PD controller to solve the yaw angle control problem. Two controllers were used in [13] to compare the simulation and real-time stabilisation for a quadrotor using the LQR controller and the robust H_∞ controller through a trajectory tracking test in high speed. The results showed that the H_∞ controller could not track the trajectory in real time while the LQR gave a good tracking in both the simulation and real-time tests.

[14] presented a comparison of the simulation results between the PID controller and the fuzzy logic one in take-off, hovering and landing stabilisation of a quadrotor UAV for a constant angle of each rotor. The results revealed that the system became stable within a short time of performing. A PD controller was introduced to handle the dynamic changes problem when the vehivle tuch the land in [15] and achieve the robustness of the quadrotor UAV's take-off and landing in a simulation.

[16] presented a mechanical design, dynamic modelling, sensing, and control system (PID) for an autonomous micro quadrotor. The results indicated that the quadrotor had the ability to perform many actions in a challenging environment

with high level of accuracy, which gives them efficiency in applications. A path tracking strategy was implemented to achieve this task by tracking reference trajectory signals. A comparison between two nonlinear control techniques, backstepping and sliding-mode, was administered through an open-and-closed-loop simulation to reinforce control laws [17].

The specific characteristic of UAV enables difficult or/and impossible executions in applications, such as rescue missions, transportation, surveillance, and industrial and military applications. The field of aerial transportation has matured in recent years to implement such applications. The full dynamic model of the system consists of the essential mathematical model, which represents the equations of quadrotors with four input forces and six output states in addition to the dynamic model of payload and links.

This complexity in the model structure and controller makes it difficult to perform tasks. In [18], a full control model of a quadrotor UAV was implemented, and attitude, altitude and position controls were proposed to excute the autonomous take-off, hovering, landing, and collision avoidance based on integral back-stepping control. The results were demonstrated in a simulation, and sensor noises were taken into consideration.

To sum up the aforementioned studies, the advantages of quadrotor UAVs are not only improving the performance in take-off, hovering, forward movement, side movement, and landing vertically in a limited space but also carrying different types and sizes of the payload.

2.2.1 Cable-suspended Payload with Single Quadrotor

Recently, many researchers have focused on the design and improvement of UAVs in many ways for the controller to implement different tasks for individual and team cooperation. These applications include emergency missions such as industrial and military applications, where delivering equipment and flying with carrying the suspended load safely is a necessity. Therefore, these applications require

stability of implementing tasks by adapting to disturbances and changes in the dynamic model parameters. Adaptive control has been utilised to maintain the centre of gravity and reduce the suspended load swing-free trajectory during the manoeuvring task.

Flying control of the quadrotor with a suspended payload was challenging due to the large swing angles and change of quadrotor characteristics during the transportation task. Authors in [19] proposed a baseline controller for the quadrotor with the suspended load to generate a trajectory with swing free manoeuvre and an adaptive control approach. The simulation and experiment results were verified for control validity and swing free tracking based on quadratic programming. A differential flatness hybrid system was employed in [20] with experimental results in order to find nominal trajectories to improve the limited swing for the suspended load with a quadrotor. Moreover, the control design was proposed to achieve tracking the payload position and system attitudes by handling a zero cable tension case. Tracking was enabled for both the quadrotor and load attitude. In [21], however, the desirable trajectories were determined for both the suspended rigid body payload as a point payload and the quadrotor through using differential flatness control.

An adaptive controller was added to cover the change in the centre of gravity, and a batch reinforcement learning approach was implemented in [22] to overcome the problem of swing-free trajectory generation. This predefined path was demonstrated to be feasible for autonomous quadrotors with suspended payload characteristics learning through reinforcement. Similarly, a technique based on dynamic programming was presented in [23] for obtaining desired waypoints and swing-free trajectory tracking problem of a quadrotor with the cable-suspended load.

Sadr, S. et al.[24] developed a dynamic model system using the Newton-Euler formulation compared with the Lagrange method and designed a nonlinear controller for the position and attitude of a quadrotor with slung load based on an anti-swing algorithm. The aim was to control the quadrotor position and attitudes to the

desired path and reject the swing angles, and the simulation results demonstrated control validity.

An adaptive control technique was designed based on a combination between the least-square estimation and the geometric control to transport the unknown payload mass from one point to the other [25], where the geometric control relied on the online estimator to evaluate the system parameters to track a prescribed path. Reasonable results were obtained with slight placement changes of the suspended payload over the basis. Similarly, a combination of an adaptive PD and neural network controller was proposed in [26] to cancel the effect of unmodeled dynamics, while a nonlinear PD controller was presented in [27] for stabilisation and trajectory following. The effect of the payload mass uncertainty was compensated for using a retrospective cost adaptive controller. The proposed controller was tested in a real vehicle to verify its validity. A vision-based extended kalman filter control was addressed by [28] in three dimensions closed loop. The load's states were estimated due to the camera detection placed downward with respect to that of the quadrotor, and an onboard controller was set up to handle aggressive manouvers to achieve large swing angles. Promising results were obtained with the largest agile load angle 53°. In [29] a coordinate-free dynamical model was derived for a quadrotor carrying a load by a flexible cable with a spring. The geometric control was developed to solve the tracking control problem for a payload position with an elastic cable using singular perturbation to validate the proposed controller.

2.2.2 Cable-Suspended Payload with Multiple Quadrotors

Another distinct load transportation approach is comprised of a cooperation of multi-quadrotors, in which one load is suspended by more than one vehicle in different ways. Cooperative aerial transportation of multiple UAVs has been used for transporting a payload suspended by a fixed cable, which is characterised by high accuracy in the equilibrium of position and orientation over individual quadrotors. This dynamic model is more challenging due to the system complexities

with a high degree of freedom; and because of some hazardous missions, a significant fluctuation of load swing angles is ensured. This situation may lead to a change of the flight system characteristics. In spite of this, it is essential that this system dynamic can be able to overcome the difficulties and inconvenient changes. In the previous works, researchers concentrated on solving the control problem of a part of the multiple flight robots with suspended payload by cables.

The Euler-Lagrange equation was used in [30] for an arbitrary number of quadrotors transporting a rigid body. A geometric controller was used to transport a payload carried by the quadrotors with flexible cables to the required fixed point, and the control problem was the payload position and system stability to be achieved.

[31] presented the design and construction of a controller of an arbitrary number of quadrotor UAVs with a rigid body payload suspended via rigid links. This controller ensured the implementation of safe transportation of the payload by cooperative quadrotors using a coupled dynamic between payload, links, and quadrotors. This geometric controller was introduced based on coordinate-free equations and derived using the Lagrange method to solve the system stability issue during the tracking task. By utilising a popular technique, another model was able to control not only a flexible payload ring to the desired position and attitude with multi-agent systems but also suppress the deformations to zero and proof the system controllability and observability using LQR controller [32].

Tracking control methods have been implemented for multiple quadrotor UAVs connected with a point mass, which tracks visible trajectory asymptotically via massless links. A three stages geometric nonlinear control strategy have been implemented for quadrotors with suspended load path tracking. A simple PD controller was presented in [33] for tracking and controlling a team of quadrotors with a suspended point mass load. The coupling effects between the quadrotors and the suspended payload were considered based on the designed controller to reject the load fluctuations. The quadrotor dynamic model was represented in eight degrees of freedom and it was based on a differential flatness. The hybrid control system stability analysis was guaranteed for stabilisation and path tracking

tasks. Cooperative control laws were designed by [34] to control an arbitrary number of aerial towing robots manipulating and transporting a payload in three dimensions.

A nonlinear control technique based on the dynamic model has also been proposed for a swing free stabilisation and trajectory tracking of quadrotors with a suspended load. Authors in both [35] and [36] proposed a differential flatness method for trajectory planning of quadrotors with a suspended load transportation problem. In their works, both a three-dimensional rigid body and a point mass load were suspended by multiple quadrotors via flexible cables. A coordinate-free dynamic was used based on the linearisation of the system equations. In [36], both single and multiple quadrotors with slung load were considered to track a desired trajectory.

A geometric feedback controller was implemented in [37] to track a predefined trajectory of the load attitude and position. Furthermore, a geometric controller was constructed to improve the performance of controlling a cable suspended payload with multiple quadrotors following the desired trajectory. Where a coordinate-free dynamical model was developed based on equations of motions and then a geometric feedback controller was designed. The load was suspended by multiple quadrotors as a point mass, while the proposed controller was used to control the quadrotors yaw angle. Another problem to solve with more than one UAV using fault free sliding mode control (SMC) was introduced by [38], where a fault tolerant control strategy was adopted to investigate and solve the issue of fire detection and tracking faults in forest monitoring. Similarly, a cooperative formation of two quadrotors UAVs was proposed in [39] using a nonlinear underactuated controller based on partial feedback linearisation to track a trajectory. Rejecting load disturbance for lateral xz-plane was achieved in the simulation.

2.3 Linear Quadratic Regulator (LQR) and Iterative (ILQR)

A review of the literature shows that an optimal linear quadratic regulator LQR, a Sequential Linear Quadratic SLQ, a linear quadratic Gaussian LQG and an Iterative Linear Quadratic ILQR controller have been applied on the cable suspended payload with quadrotors in order to solve the problem of stabilisation and accuracy.

A linear quadratic control technique LQR was introduced for addressing the trajectory tracking problem in [40] and a zero steady-state error was obtained based on the integrated D-methodology with the anti-wind-up technique. This strategy was adopted in order to rely on the linearised model of UAV quadrotor to track a predefined 3D trimming trajectory by the LQR controller. The experimental test results of this controller demonstrated small errors. Likewise, a practical test was implemented in [41] to track a predefined trajectory using an LQR controller. This controller was implemented with and without the Model-Free Control (MFC) algorithm and tested practically. The test results showed the augmented effectiveness of the system performance.

An iterative linear quadratic regulator ILQR using feedback gain control was presented in [42]. This control gain resulted in a better solution by yielding faster convergence to the nominal trajectory according to the optimal feedback control law, which was computed via LQR modification. It is worth mentioning that non-linear dynamical systems were linearised for three applied models. The first model was a 2-link musculoskeletal arm model, where the angle of a joint was achieved to facilitate convergence to the goal (i.e. reach movement representation) and energy efficiency was obtained. The second nonlinear dynamic model was realistic muscle actuators added to the arm, and a swing-up simple inverted pendulum was introduced as a third nonlinear dynamic model. In [43] a global trajectory generation, a trajectory control, and a linearisation technique kinematic model of 3-dimensional configuration for a wheeled mobile robot system were presented. The proposed

control method was used to design accurate trajectory tracking for two courses of trajectories "S" and "8" to improve the control sequences for linear and angular velocities iteratively. This controller was the iterative linear quadratic regulator ILQR, which improved performance in simulation and practical application.

A demonstration was applied in [44] to improve the convergence of the third required iteration on nonlinear control problems. This was achieved through simulating a quadrotor with obstacles and physical differential drive robots using iterative LQR. The smoothing concept of LQR controller was tantamount to Kalman smoothing. An iterative computation technique was applied with nonlinear dynamics and non-quadratic cost in order to achieve the faster convergence and construct locally optimal feedback control. This controller was improved with the forward pass and backward pass implementations by the standard LQR Riccati equation. The Riccati equation were constructed to compute cost-to-come and cost-to-go functions. The aggregate of both functions gave the total cost function, which provided natural points for the capable linearised dynamics and cost quadratic.

Another distinct control approach of payload lifting, and transportation has been proposed in different scenarios. In [45], lifting and transporting a payload was performed for a quadrotor carrying the load with a cable. The proposed controller was a linear quadratic regulator LQR control algorithm. Two modes were presented: the starting mode through taking-off without the load influence and the switching mode with the effect of the load. The simulation results for LQR controller were compared with the PD controller results. However, in other works, different tasks were approved with a linear controller. In [46] an LQR controller was designed for a quadrotor to maintain the position and attitude equilibrium in spite of losing a single propeller, two opposite or even three propellers, while a combination control of linear quadratic regulator LQR and sliding mode control (SMC) was used for leader and follower formation maintenance [47]. It was found that the inner and outer loops for position and attitudes were improved for the trajectory tracking simulation.

In recent years, the difficulties of performing a nonlinear design have acquired increasing attention in order to improve the system stability. Among these problems are manual control derivation, complexity in design and combination of different tasks, time-consuming and difficult implementation of different specific controllers. For instance, a dynamic model was developed based on the Udwadia-Kalaba Equations for a slung load lifted by a quadrotor in [48]. Appropriate neural network and adaptive control were proposed to improve the attitude, then numerical simulations were performed. In [49], a hybrid dynamical system was modelled to navigate a quadrotor with a slung load in known obstacle environments, and two challenges were presented. The first challenge was that the full system had to guarantee obstacle avoidance and the second challenge was giving permission to manoeuvres through the adaptation between subsystems.

Furthermore, nonlinear controllers have been presented in many specific approaches to obtain and maintain better results for position and attitude. These controllers have been applied through many techniques such as the backstepping controller for a payload connected with the quadrotor centre of gravity. Kane's method was modelled for tracking the trajectory and verifying the simulation results in [50]. Therefore, a linear control strategy has been considered in order to overtake limitations in different publications. A Sequential Linear Quadratic SLQ control, for example, was employed by [51] and an iterative LQG algorithm ILQG was used in [52] for a hybrid model quadrotor and slung load to perform two approaches. The first was passing through an unfair high window for the payload pass by implementing aggressive manoeuvres. The second was to demonstrate a go-to-goal task with the failure of one and two rotors.

Moreover, some researchers have published papers to compare between a linear and nonlinear controller for quadrotors with a slung load. For example, a Nonlinear Model Predictive Control NMPC was proposed by [53] to actively track waypoints precisely and restrain large oscillations for the slung load and then compare the performance with a linear-quadratic regulator LQR controller to improve the simulation results considering aggressive manoeuvres. In [54], however, the proposed controllers were tested through simulation to verify their validity for the precise

and safe operation of a heavy slung load. Authors in [55] proposed an iterative Linear Quadratic Gaussian (ILQG) control for a stochastic nonlinear system. Such a method was applied by eliminating approximation to the optimal cost function.

2.4 Model Predictive Control (MPC)

The most important priority for researchers in terms of designing an optimal controller has been to improve the performance and guarantee the system robustness in difficult conditions based on complex non-linear dynamic models. Therefore, MPC controllers have commonly been executed to tackle the performance of the system states.

The MPC controller has been attracted by many applications due to the beneficial efforts such as the constraints included in the control design and the specific horizon where the control action is enabled. Moreover, this controller was performed in the linear and nonlinear forms based on using the linearised equations of motion via the nonlinear model. To this end, this survey below presents a review of the previous works on linear and nonlinear controllers.

2.4.1 Linear Model Predictive Control

In [56], a linear MPC scheme was designed for a multi-rotor system with a slung load. The simulation results for the MPC were compared with those of the LQR algorithm and the experiment test was verified in real flight. Similarly, in [57] both control algorithms LQR and MPC were introduced based on a linear model and performed on a quadrotor testbed (Qball-X4) in order to control against the control loss with a fault-tolerant control strategy. Furthermore, a fault-tolerant controller was designed in order to compare the performance. A linear MPC was proposed in [58] and the model was simplified with only a two-dimensional movement based on the least square identification. The authors maintained convergence in a circular

path around the desired point, and an interface was provided with the actual quadrotor in real time using MATLAB.

Cooperative UAV control in a form of multiple team formation was accomplished under a linear MPC control law [59] using the Qball-X4 quadrotor. This approach was implemented without constraints and no load was considered. The contribution of this controller was to make velocity matching and prevent collisions. The encirclement situation was applied for multi quadrotors team formation around the target based on a linear MPC to improve encirclement and collision avoidance between Qball X4 quadrotors [60]. In [61] an MPC controller was presented with PID for a quadrotor to autonomously track a predefined trajectory without constraints. This control design was based on inner and outer loops, and on longitudinal and lateral control feedback law. Each controller utilised a decomposed control signal, then a test of control performance was conducted to obtain satisfactory results. These results were based on a simple test, forward flight and hovering tests. An optimal flight control MPC for a quadrotor with a suspended load was presented in [62] using visual feedback to compute the load position accurately. The state vector and control vector constraints were applied, utilising an onboard processor, so that the load position can be calculated and connected through a wireless network. However, the results were based on the insignificant effect of the system behaviour through the tracking method. In [63], however, a visual method was presented for trajectory generation of a quadcopter in real time by solving a convex optimisation. On the other hand, in [64] an unconstrained MPC controller was proposed to track a trajectory for quadrotors based on three levels of control, and the structure of the feedback equivalent system led to a decrease in the complexity of MPC in real time.

In terms of UAV stabilisation and desired trajectory tracking, an LMPC was proposed using a hierarchical strategy to achieve the system stability and obstacle avoidance in [65]. The results showed the ability to control the state constraints and the flexibility of trajectory planning by using the LMPC and decentralised LMPC controller [66]. Likewise, a robust MPC controller was designed based on piecewise affine linear systems (PWA) to control a quadrotor's attitudes in

severe environmental conditions [67]. The models were simplified, and the wind-disturbance was considered as a challenging issue when implementing sudden manoeuvres. The control input constraint was imposed on the system.

2.4.2 Nonlinear Model Predictive Control MPC

Due to the system's high requirements to solve the control problem under challenging environments, a nonlinear MPC has been applied in several areas. An integral MPC controller was executed for a single quadrotor to improve the tracking performance in [68]. The dynamic model was presented based on piecewise affine (PWA) systems with the consideration of physical system constraints. A constrained robust MPC technique was applied over the constrained environments for a single quadrotor manoeuvring with fixed payload in [69], while an autonomous multi-copter slung load system was presented in [70] based on a constrained MPC approach and PID controller to follow a desired derived trajectory.

A MPC was presented in [71] for manoeuvres at very constrained environments. In this study, a quadrotor position control was addressed based on implementing the extended kalman filter for translation velocity estimation and switching the MPC Controller for angular rates and accelerations. In [72], a MPC was used to minimise the predicted tracking attitude errors of a quadrotor with a fixed payload. This strategy was applied for take-off, hovering, and landing tasks with a reactive safety mode.

The implementation of an MPC controller in a tilt-rotor UAV with a suspended payload was reported in [73] and [74]. Moreover, a horizon motion planning was integrated, and an obstacle avoidance ability was achieved by M. Saska et al. [75] for micro aerial vehicles (MAV) using MPC control for maintaining leader-follower formation mechanism with follower stabilisation. Similarly, the authors in [76] introduced an MPC controller for leader-follower UAVs to avoid collision in a restricted area, and the results showed zero steady-state position error. Path tracking and obstacles avoidance were tackled in [77] by using a nonlinear guidance

logic for a MAV, where the MPC controller was tested and then the predictive path and obstacles avoidance were achieved.

An automation flight control was presented in [78] and it relied on MPC implementation for UAVs indoors. The Vicon system was employed for tracking the formation system to apply flight control and record data based on MPC in order to verify the system performance. An indoor demonstration was implemented in the concept of air traffic management (ATM) and geo-facing environment [79]. Experimental results were provided, and they described the modification of a learning-based MPC technique to improve the dynamic response and guarantee robustness, then converge by applying the quadrotor on an ultra-low-voltage processor in real time.

Comparing the results of linear and nonlinear MPC control to track a trajectory using MAVs under wind disturbance, [80] shows that the difference between them was not significant. To increase online optimisation time, a high-level MPC and low-level linear control framework were introduced for helicopter autonomous flight control in [81]. The simulation results were verified using the proposed control.

2.5 Stackelberg Games

This section focuses on one of the most important control aspects. It has been presented by non-cooperative and cooperative game control. In the first aspect, a non-cooperative game theory was adopted using a new suitable algorithm. This algorithm was described in [82] to improve the game design for a multi-agent based on AI approach. The N-Tuple Bandit algorithm was developed to improve exploration and exploitation balancing using bandit approach. Indeed, better robust results were obtained from the application of this algorithm and these were compared with both Biased Mutation and Mutation Hill Climber.

Another application for a target tracking control for multi-agent robots was the dynamic Stackelberg game framework. This framework was applied to multiple

robots for target tracking control in team formation based on semi-cooperative Stackelberg game theory [83], where fuzzy logic was used to adjust the weighting parameters in the cost functions and enhanced robustness was achieved. This algorithm was used to enhance the system robustness by achieving the cost function convergence. In [84], the non-cooperative Stackelberg game approach for formation robot coordination was used to solve the tracking problem and avoid collision.

In the form of Stackelberg games, the power control for the quality optimising service (QOS) in wireless networks was solved [85], while the cognitive radio power was used to change transmission power level [86] and the transportation networks were utilised to maximize toll revenue in [87]. A wireless body area network was proposed to increase the value of healthcare service with high-security in [88] based on a non-cooperative Stackelberg security game theory. In security domain with leader player as a patrol and follower as a robber in [89], an optimal leader strategy was found.

In terms of energy management, a controller was studied based on a Stackelberg game for electric vehicle charging in [90] by achieving a beneficial trade-off between battery charging and the cost. An energy management controller was developed for hybrid electric vehicles (HEVs) in [91] to penalise fuel consumption, battery state of charge deviation and NOx emissions under a game framework. In order to control pollution caused by these emissions in each country, a simple model pollution control was introduced in [92]. This controller was put based on non-cooperative differential game theory to minimise the cost function for the linear combination of pollution and friendly cost of environmental policies. A probabilistic game scenario was implemented in simulation and experiment in [93] using a fleet of UAVs and UGVs based on a hierarchical architecture in two policies. These policies were presented by local-max and global-max pursuit-evasion with expected capture time to catch the evaders.

A coordination control approach has been presented in many publications, where multiple robots perform their tasks in a complex environment structure using the Nash equilibrium concept. Based on N-person game approach, an elementary task

planning method was presented, and a hybrid system architecture was designed in order to control the team of robots performing a non-cooperative game task [94]. An effective exploration test was presented in simulation and carried out in a complex structure of environments. However, in this transportation task, performance could not be guaranteed using a one-step-ahead plan. In [95], where the tracking target problem was proposed, achieving the required task could be approved by avoiding robot collision with a target robot or obstacles in the system configuration. This was due to the competitive situation between agents in tracking the target robot. Therefore, a switch of the method was proposed using the non-cooperative Nash and the semi-cooperative Stackelberg equilibrium. By comparing the results with those of single leaders in terms of social welfare, the researchers concluded that the equilibrium point was subjected to the multi-leaders.

In the case of multiple leaders, presented by Wi-Fi, SCSP and MSP, with multiple followers, the Stackelberg game approach was proposed for more complex situations in order to improve high social welfare [96]. Similarly, in [97], a Stakelberg security game control was applied for multiple defenders and attackers based non-cooperative approach. In addition, a Stackelberg game was introduced to swarm robot converge control in [98].

On the other hand, the cooperative game controller has been studied and tested theoretically and experimentally in various environments. Stackelberg security aspects were presented based on intelligent players in the realistic performance of the transportation and computation system [99]. A roundabout test-bed situation was introduced in [100], making use of two agents' cooperative strategy to solve the conflict case. This strategy was based on Prisoner's Dilemma approach to implement vehicle to vehicle (V2V) decision making autonomously in roundabouts. This technique was proposed between two agents to achieve better reward of the system behaviour through analysing the agents' actions and their influence on one another. The test results of (V2V) cooperative decision making showed an improvement in terms of managing and decreasing the delay time (waiting time) for autonomous agents.

The non-zero-sum game theory was applied in real time and dynamic environment in [101], and a game method based on multi-agents was proposed using dynamic-programming to optimise the searching strategy [102]. A different game theory modelling was developed between a human and a vehicle [103]. In this latter study, four paradigms were categorised based on the interaction between a driver's steering and vehicle collision avoidance control model to follow the desired path and maintain various optimisation problems. These paradigms were named as decentralised, non-cooperative Nash, non-cooperative Stackelberg and cooperative Pareto paradigms. Incentive dynamic Stackelberg games allow the leader and follower players to work cooperatively through an incentive mechanism by which the leader encourages the follower to cooperate on minimising the team cost (leader's cost). They were used in security domain [104] and in robot team navigation [105].

2.6   Summary

Most of the aforementioned studies have focused on novel methodologies for the trajectory quadrotor or payload control on standard position and attitude for single and multi-quadrotors with or without slung load.

Nonlinear controllers can stabilise the systems. But the control performance can not be optimized. Optimal controllers, such as LQR,ILQR, MPC, and game controllers, are able to stabilise the systems and optimise the performance. Although, LQR,ILQR and MPC controllers have been applied to the systems of quadrotors with suspended payload in one way or another, no explicitly applications of ILQR and Game controllersto the two quadrotors withsuspended payload have been found. No applications of the MPC controllers to two quadrotors with suspended payload under the consideration of both input and state constraints have been found. The following chapters will explore how to use these optimal controllers in single and two quadrotors with suspended payload and reveal what the control ferformance could be achieved under these optimal controllers.

Chapter 3

Dynamic Models for Single and Two Quadrotors with Suspended Payload

3.1 Introduction

In this chapter, dynamic models for single and two quadrotors with suspended payload are introduced. Two dynamic systems are suggested in this thesis to test the proposed controller and handle the system complexities. Two models are chosen: one where the payload is suspended by a cable from a single quadrotor and another where the payload is suspended by two cables from two quadrotors.

The single UAV quadrotor is connected to the suspended payload by a cable from the centre of gravity. This model is built based on Euler angles considering payload position, and quadrotor and payload's attitudes. Relying on the rope angles, the operating point can be analysed. A heavyweight load is taken into account by creating a new model system to deal with this load. In the two quadrotors model, however, the payload position, two quadrotors' orientations and four swing angles are employed as the system states, while the control inputs are presented by two quadrotors' forces and six moments.
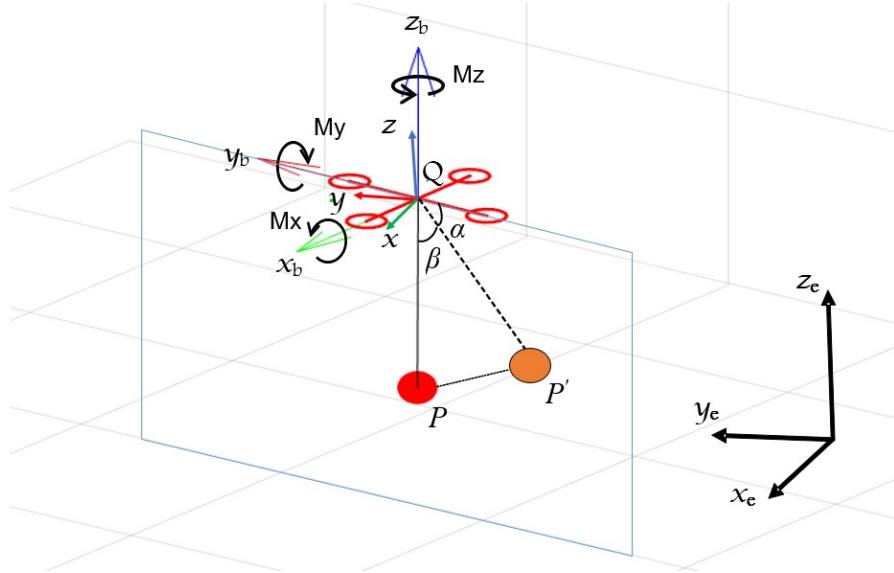
FIGURE 3.1: Single quadrotor carrying a payload

3.2 Single Quadrotor with Suspended Load

3.2.1 Nonlinear Dynamic Model Description

The point mass load suspended with a single quadrotor is described by a derivation of a dynamic model in this section. Figure 3.1 illustrates the representation of the dynamic model which consists of the quadrotor carrying a payload suspended by a cable. There are two coordinate reference frames, an inertial frame (earth fixed frame) denoted by $\mathbf{e}$ and a rigid body fixed frame denoted by $\mathbf{b}$. Their coordinate positions are denoted as $x_\mathbf{e}, y_\mathbf{e}, z_\mathbf{e}$ and $x_\mathbf{b}, y_\mathbf{b}, z_\mathbf{b}$ respectively. The payload attitude is represented in three dimensions and its position is considered with respect to the inertal frame. The system description is presented in Figure 3.1 including the inertial frame, intermediate frame, and body-fixed frame. The vertical and horizontal forces generated by each propeller and swing angles of the cable with respect to the intermediate frame are also shown, where the intermediate frame is the translation result from the inertial frame to the centre of quadrotors [106].

To simplify the problem, some reasonable hypotheses are given as follows:

1. The quadcopter is considered as a symmetrical rigid body.

2. The payload is considered as a point mass and is attached at the centre of the quadcopter.

3. The cable tension is always non-zero.

4. The air drag of the propellers is negligible.

Coordinates of different unit orthogonal vectors in appropriate frames are:

$$\boldsymbol{E}_1 = [1, 0, 0]^T$$
$$\boldsymbol{E}_2 = [0, 1, 0]^T \tag{3.1}$$
$$\boldsymbol{E}_3 = [0, 0, 1]^T$$

The following relationships are available:

$$\boldsymbol{\rho} = [-sin(\beta), -cos(\alpha)cos(\beta), sin(\alpha)cos(\beta)]^T$$
$$\boldsymbol{\xi}_P = x_P \boldsymbol{E_1} + y_P \boldsymbol{E_2} + z_P \boldsymbol{E_3} \tag{3.2}$$
$$\boldsymbol{\xi}_Q = \boldsymbol{\xi}_P + L_r \boldsymbol{\rho}$$

The rotational velocity Jacobian is

$$\boldsymbol{\Omega} = \boldsymbol{\jmath}\dot{\boldsymbol{\eta}} = \begin{bmatrix} 1 & 0 & -sin(\theta) \\ 0 & cos(\phi) & sin(\phi)cos(\theta) \\ 0 & -sin(\phi) & cos(\phi)cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{3.3}$$

3.2.2 Euler-Lagrange Equation

The cable suspended load is modelled with eight degrees of freedom, which are comprised of six for the quadrotor as a rigid body and the rest for the spherical pendulum. Choosing $\boldsymbol{q} = [x_P, y_P, z_P, \ \alpha, \beta, \phi, \theta, \psi]^T$ as the generalised coordinates will not only be convenient while controlling the trajectory of the payload but also be helpful for extending to multi-vehicle situations. Because of the change of the cable tension from slack to taut, two mathematical models are taken into

account: a quadrotor with no load and a quadrotor with cable-suspended load. A switching process is used to transfer the operation from the first to the second model depending on the cable tension. As a result, the Lagrangian $\mathfrak{L}$ is composed by subtraction of the kinetic and potential energies denoted by $\mathfrak{T}$ and $\mathfrak{U}$ as clarified in the equations below:

$$
\begin{aligned}
\mathfrak{T} &= \frac{1}{2} m_P (\dot{\boldsymbol{\xi}}_P)^T \cdot \dot{\boldsymbol{\xi}}_P + \frac{1}{2} m_Q (\dot{\boldsymbol{\xi}}_Q)^T \cdot \dot{\boldsymbol{\xi}}_Q + \frac{1}{2} (\boldsymbol{\Omega})^T \boldsymbol{I}_Q \boldsymbol{\Omega} \\
\mathfrak{U} &= m_P g \boldsymbol{\xi}_P \cdot \boldsymbol{E}_3 + m_Q g \boldsymbol{\xi}_Q \cdot \boldsymbol{E}_3 \\
\mathfrak{L} &= \mathfrak{T} - \mathfrak{U}.
\end{aligned}
\tag{3.4}
$$

Then the Euler-Lagrange equation is

$$
\frac{d}{dt} \left(\frac{\partial \mathfrak{L}}{\partial \dot{\boldsymbol{q}}} \right) - \frac{\partial \mathfrak{L}}{\partial \boldsymbol{q}} = \mathbf{Q}
\tag{3.5}
$$

where $\mathfrak{L}$ is the Lagrangian and the generalised force $\mathbf{Q}$ defined here is based on the choice of the generalised coordinates $\boldsymbol{q} \in \mathbb{R}^8$ and the external conservative force $\boldsymbol{F}$ is $[\boldsymbol{F}_Q^T, \boldsymbol{M}_Q^T]^T \in \mathbb{R}^4$.

However, $\mathbf{Q}$ cannot be calculated directly via D'Alembert's Principle as a transformation is required beforehand. The D'Alembert's Principle is a fundamental of theoretical physics based on the Newton's laws of motion, which was discovered by the French scientist named Jean le Rond d'Alembert [107]. This is the principle of virtual work obtained by applied forces on the dynamic equilibrium system along a virtual displacement. It states that the sum of the external real forces $\boldsymbol{f}$ applied on the body system minus the body system mass m times acceleration a reacting by this system as projected along the system displacement is zero $\boldsymbol{f} - ma = 0$.

In this chapter the main contribution of utilising this approach lies in the fact that it eliminates the system dynamic problem under the body equilibrium consideration using inertial forces and moments. Indeed, at the centre of gravity the

forces must apply while the moments can be able to act at any point, which simplifies calculation and eliminates forces from unselected points from the moment equations [108].

Firstly, the thrust of a quadrotor should be transformed from the body frame into the $\boldsymbol{S}_b^K$ frame,

$$\boldsymbol{F}_Q = F_z \boldsymbol{e}_3. \tag{3.6}$$

Define the unit orthogonal vectors of $\boldsymbol{S}_b$:

$$\boldsymbol{e}_\mathcal{I} = \boldsymbol{T}_{e2b} \boldsymbol{E}_\mathcal{I}, \mathcal{I} = 1, 2, 3 \tag{3.7}$$

Secondly, the torques of the quadrotor should be transformed from $\boldsymbol{S}_b$ into an appropriate frame in terms of Euler angles by velocity Jacobian matrix. As $\boldsymbol{M}_Q$ is not defined based on Euler angles, a transformation from $\boldsymbol{M}_Q$ to its generalised form in terms of Euler angles is $\boldsymbol{M}_\eta = [M_\phi, M_\theta, M_\psi]$

$$\boldsymbol{M}_\eta = \boldsymbol{J}^T \boldsymbol{M}_Q. \tag{3.8}$$

Due to the following identical equations of power flow through the joint

$$\boldsymbol{M}_\eta \cdot \dot{\boldsymbol{\eta}} = \boldsymbol{M}_\eta^T \dot{\boldsymbol{\eta}} = \left(\boldsymbol{J}^T \boldsymbol{M}_Q\right)^T \dot{\boldsymbol{\eta}} = \boldsymbol{M}_Q \cdot \boldsymbol{\Omega},$$

the generalised force **Q** is given by equation (3.9).

$$\boldsymbol{Q}_i = \boldsymbol{F} \cdot \frac{\partial \boldsymbol{\xi}_Q}{\partial \boldsymbol{q_i}} + \boldsymbol{M}_\eta \cdot \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{q_i}}, \boldsymbol{i} = 1, 2, ...4. \tag{3.9}$$

Taking the generalised forces and Eq (3.4) into Eq (3.5), the Euler-Lagrange equation based on $\boldsymbol{G} \in \mathbb{R}^{8 \times 8}$ matrix is denoted by a symmetric matrix, and the system model function f can be rewritten in

$$\boldsymbol{G}\ddot{\boldsymbol{q}} = g(\boldsymbol{F}, \boldsymbol{q}, \dot{\boldsymbol{q}}) = f(\boldsymbol{x}, \boldsymbol{u}.) \tag{3.10}$$

In the trajectory tracking control, the operating point is considered as the balance situation where the Euler angles of the quadrotor are equal to zero and the cable angles are equal to the designed constants. The state $\boldsymbol{x}$ and control $\boldsymbol{u}$ of the system are defined as below:

$$\boldsymbol{x} = \left[x_P, \dot{x}_P, y_P, \dot{y}_P, z_P, \dot{z}_P, \alpha, \dot{\alpha}, \beta, \dot{\beta}, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi} \right]^T \in \mathbb{R}^{16},$$

$$\boldsymbol{u} = [F_z, M_x, M_y, M_z]^T \in \mathbb{R}^4.$$

As for the control vector inputs, a non-linear state space equation is then given as

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}.) \tag{3.11}$$

3.2.3 Equilibrium Points and Linear Time Varying Model

In order to utilise linear control strategy, equilibrium points and the corresponding linear models must be given beforehand. Generally, the equilibrium points $(\boldsymbol{x}_{eq}, \boldsymbol{u}_{eq})$ are recognised as stationary points as they satisfy the following equation:

$$0 = f\left(\boldsymbol{x}_{eq}, \boldsymbol{u}_{eq}\right). \tag{3.12}$$

With Taylor's series expansion, equation (3.11) can then be approximated by a linear equation around $(\boldsymbol{x}_{eq}, \boldsymbol{u}_{eq})$ and the linearised model is obtained in equation (3.13):

$$\dot{\boldsymbol{x}} \approx f\left(\boldsymbol{x}_{eq}, \boldsymbol{u}_{eq}\right) + \left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_{eq}} (\boldsymbol{x} - \boldsymbol{x}_{eq}) + \left.\frac{\partial f}{\partial \boldsymbol{u}}\right|_{\boldsymbol{u}_{eq}} (\boldsymbol{u} - \boldsymbol{u}_{eq})$$

$$\Rightarrow \delta\dot{\boldsymbol{x}} = \boldsymbol{A}\delta\boldsymbol{x} + \boldsymbol{B}\delta\boldsymbol{u} \tag{3.13}$$

where, $\delta \boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_{eq}$, $\delta \boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_{eq}$, $\delta \dot{\boldsymbol{x}} = \dot{\boldsymbol{x}} = \boldsymbol{G}\ddot{\boldsymbol{q}}$.

Where $\boldsymbol{G} = \boldsymbol{M}$, the linearised model is obtained in equation (3.14):

$$\ddot{\boldsymbol{q}} = \boldsymbol{M}^{-1} \left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_0} \delta \boldsymbol{x} + \boldsymbol{M}^{-1} \left.\frac{\partial f}{\partial \boldsymbol{u}}\right|_{\boldsymbol{u}_0} \delta \boldsymbol{u}. \tag{3.14}$$

Furthermore, equation (3.13) can be transformed into a discrete form:

$$\boldsymbol{x_{k+1}} = \boldsymbol{A_k x_k} + \boldsymbol{B_k u_k} \tag{3.15}$$

and $\boldsymbol{A_k} \in \mathbb{R}^{16 \times 16}$, $\boldsymbol{B_k} \in \mathbb{R}^{16 \times 4}$

where the generalised forces, $\mathbf{Q}$, matrix $\boldsymbol{M}$ and $f(\boldsymbol{x}, \boldsymbol{u})$ are detailed in (3.17), (3.16) and (3.18), respectively:

Generalized forces:

$\mathbf{Q_1} = F_z(C(\phi)C(\psi)S(\theta) + S(\phi)S(\psi))$

$\mathbf{Q_2} = F_z(-C(\psi)S(\phi) + C(\phi)S(\theta)S(\psi))$

$\mathbf{Q_3} = F_z C(\theta)C(\phi)$

$\mathbf{Q_4} = F_z L_r C(\beta)(C(\alpha)C(\theta)C(\phi) + S(\alpha)(-C(\psi)S(\phi) + C(\phi)S(\theta)S(\psi)))$

$\mathbf{Q_5} = F_z L_r(-C(\theta)C(\phi)S(\alpha)S(\beta) + C(\alpha)S(\beta)(-C(\psi)S(\phi)$
$\quad + C(\phi)S(\theta)S(\psi)) - C(\beta)(C(\phi)C(\psi)S(\theta) + S(\phi)S(\psi)))$

$\mathbf{Q_6} = M_x$

$\mathbf{Q_7} = M_y C(\phi) - M_z S(\phi)$

$\mathbf{Q_8} = -M_x S(\theta) + C(\theta)(M_z C(\phi) + M_y S(\phi))$

$$\tag{3.16}$$

Hereafter, C stands for *cos*, S stands for *sin*.

Matrix $\boldsymbol{M}$ can be given as:

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{0}_{5\times3} \\ \boldsymbol{0}_{3\times5} & \boldsymbol{M}_{22} \end{bmatrix} \tag{3.17}$$

where $\boldsymbol{M}_{11}$ and $\boldsymbol{M}_{22}$ are defined as follows:

$$\boldsymbol{M}_{11} = \begin{bmatrix} m_P + m_Q & 0 & 0 & 0 & -L_r m_Q C(\beta) \\ 0 & m_P + m_Q & 0 & L_r m_Q C(\beta) S(\alpha) & L_r m_Q C(\alpha) S(\beta) \\ 0 & 0 & m_P + m_Q & L_r m_Q C(\alpha) C(\beta) & -L_r m_Q S(\alpha) S(\beta) \\ 0 & L_r m_Q C(\beta) S(\alpha) & L_r m_Q C(\alpha) C(\beta) & L_r^2 m_Q C(\beta)^2 & 0 \\ -L_r m_Q C(\beta) & L_r m_Q C(\alpha) S(\beta) & -L_r m_Q S(\alpha) S(\beta) & 0 & L_r^2 m_Q \end{bmatrix}$$

$$\boldsymbol{M}_{22} = \begin{bmatrix} I_x & 0 & -I_x S(\theta) \\ 0 & I_y C^2(\phi) + I_z S^2(\phi) & (I_y - I_z) C(\theta) C(\phi) S(\phi) \\ -I_x S(\theta) & (I_y - I_z) C(\theta) C(\phi) S(\phi) & I_x S^2(\theta) + C^2(\theta)(I_z C^2(\phi) + I_y S^2(\phi)) \end{bmatrix}.$$

Function $f(\boldsymbol{x}, \boldsymbol{u})$ can be defined as:

$$f_1(\boldsymbol{x}, \boldsymbol{u}) = F_z C(\phi) C(\psi) S(\theta) + F_z S(\phi) S(\psi) - L_r m_Q S(\beta) \dot{\beta}^2$$

$$f_2(\boldsymbol{x}, \boldsymbol{u}) = -F_z C(\psi) S(\phi) + F_z C(\phi) S(\theta) S(\psi)$$
$$+ L_r m_Q \left(2 S(\alpha) S(\beta) \dot{\alpha} \dot{\beta} - C(\alpha) C(\beta) \left(\dot{\alpha}^2 + \dot{\beta}^2 \right) \right)$$

$$f_3(\boldsymbol{x}, \boldsymbol{u}) = -g(m_P + m_Q) + F_z C(\theta) C(\phi)$$
$$+ L_r m_Q \left(2 C(\alpha) S(\beta) \dot{\alpha} \dot{\beta} + C(\beta) S(\alpha) \left(\dot{\alpha}^2 + \dot{\beta}^2 \right) \right)$$

$$f_4(\boldsymbol{x}, \boldsymbol{u}) = L_r C(\beta)(C(\alpha)(-g m_Q + F_z C(\theta) C(\phi))$$
$$+ F_z S(\alpha)(-C(\psi) S(\phi) + C(\phi) S(\theta) S(\psi)) + 2 L_r m_Q S(\beta) \dot{\alpha} \dot{\beta})$$

$$f_5(\boldsymbol{x}, \boldsymbol{u}) = -L_r((-g m_Q + F_z C(\theta) C(\phi)) S(\alpha) S(\beta)$$
$$+ F_z (C\alpha() S(\beta)(C(\psi) S(\phi) - C(\phi) S(\theta) S(\psi))$$
$$+ C(\beta)(C(\phi) C(\psi) S(\theta) + S(\phi) S(\psi))) + L_r m_Q C(\beta) S(\beta) \dot{\alpha}^2)$$

$$f_6(\boldsymbol{x}, \boldsymbol{u}) = M_x + (-I_y + I_z) C(\phi) S(\phi) \dot{\theta}^2 + C(\theta)(I_x + (I_y - I_z) C(2\phi)) \dot{\theta} \dot{\psi}$$
$$+ (I_y - I_z) C^2(\theta) C(\phi) S(\phi) \dot{\psi}^2$$

$$f_7(\boldsymbol{x}, \boldsymbol{u}) = M_y C(\phi) - M_z S(\phi) + (I_y - I_z) S(2\phi) \dot{\theta} \dot{\phi}$$
$$+ C(\theta) \dot{\psi} \left(-(I_x + (I_y - I_z) C(2\phi)) \dot{\phi} + S(\theta) \left(I_x - I_z C^2(\phi) - I_y S^2[\phi] \right) \dot{\psi} \right)$$

$$f_8(\boldsymbol{x}, \boldsymbol{u}) = -M_x S(\theta) + C(\theta)(M_z C(\phi) + M_y S(\phi))$$
$$+ (I_y - I_z) C(\phi) S(\theta) S(\phi) \dot{\theta}^2 + (-I_y + I_z) C(\theta)^2 S(2\phi) \dot{\phi} \dot{\psi}$$
$$+ \dot{\theta} \left(C(\theta)(I_x + (-I_y + I_z) C(2\phi)) \dot{\phi} + S(2\theta) \left(-I_x + I_z C^2(\phi) + I_y S^2(\phi) \right) \dot{\psi} \right).$$

$$(3.18)$$

The linearized model can be presented in the following matrices:

$$
A = \begin{pmatrix}
0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8909 & 0 & 0 & 0 & 10.6909 & 0 & 0 & 0 \\
0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -0.8909 & 0 & 0 & 0 & -10.6909 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -21.3818 & 0 & 0 & 0 & -21.3818 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -21.0 & 0 & 0 & 0 & -21.3818 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
$$

$$
B = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1.6667 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 434.7826 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 357.1429 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 217.3913
\end{pmatrix}
$$
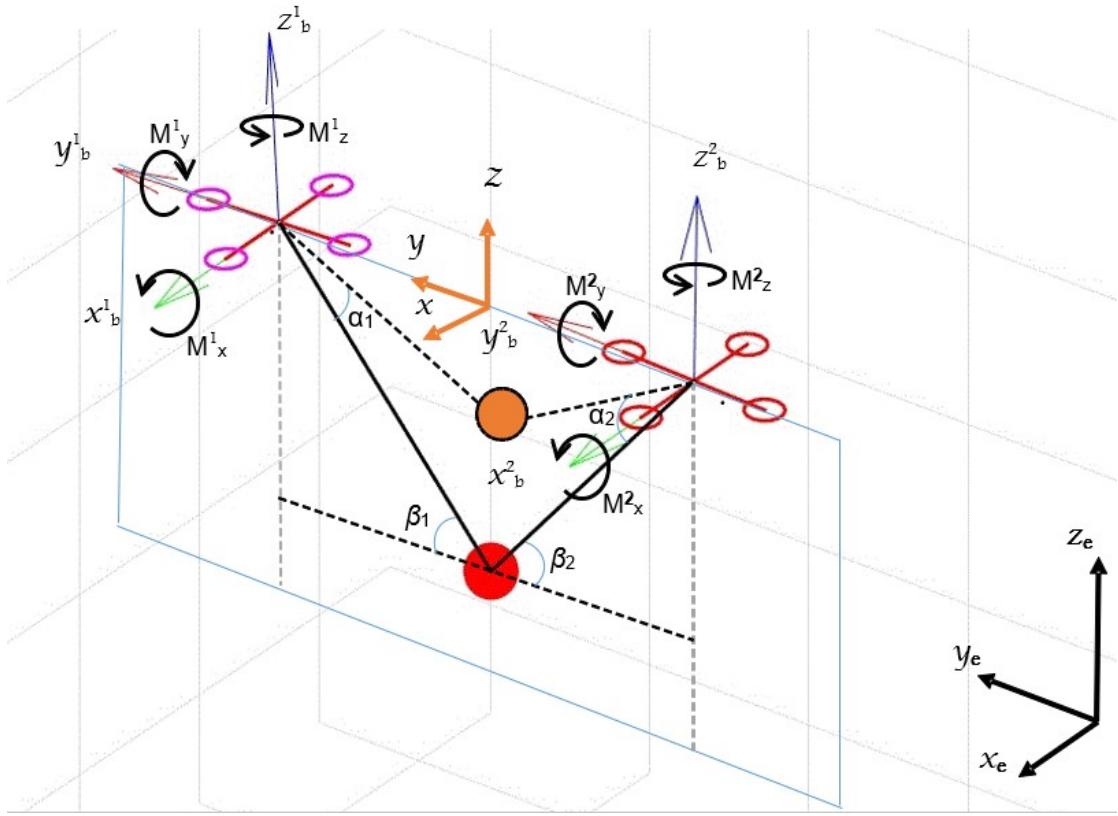
FIGURE 3.2: Two quadrotors carrying a payload

3.3 Two Quadrotors with Suspended Load

3.3.1 Model Description

Figure 3.2 displays the full system, including the inertial frame, intermediate frame, and body-fixed frame. The vertical and horizontal forces generated by each propeller and swing angles of the cable with respect to the intermediate frame are also shown, where the intermediate frame is the result of the translation from the inertial frame to the centre of quadrotors.

Coordinates of different unit orthogonal vectors in appropriate frames are given as:

$$
\begin{aligned}
\boldsymbol{E}_1 &= [1, 0, 0]^T \\
\boldsymbol{E}_2 &= [0, 1, 0]^T \\
\boldsymbol{E}_3 &= [0, 0, 1]^T
\end{aligned}
\tag{3.19}
$$

The following relationships are available:

$$\boldsymbol{\rho^K} = \left[cos(\alpha^K)cos(\beta^K), cos(\alpha^K)sin(\beta^K), sin(\beta^K)\right]^T$$

$$\boldsymbol{\xi}_P = x_P\boldsymbol{E_1} + y_P\boldsymbol{E_2} + z_P\boldsymbol{E_3} \tag{3.20}$$

$$\boldsymbol{\xi_Q^K} = \boldsymbol{\xi}_P + L_r^K\boldsymbol{\rho^K}$$

The rotational velocity Jacobian is

$$\boldsymbol{\Omega^K} = \boldsymbol{J^K}\boldsymbol{\dot{\eta}^K} = \begin{bmatrix} 1 & 0 & -sin(\theta^K) \\ 0 & cos(\phi^K) & sin(\phi^K)cos(\theta^K) \\ 0 & -sin(\phi^K) & cos(\phi^K)cos(\theta^K) \end{bmatrix} \begin{bmatrix} \dot{\phi^K} \\ \dot{\theta^K} \\ \dot{\psi^K} \end{bmatrix}$$

3.3.2 Euler-Lagrange Equation

There are 13 degrees of freedom in the quadrotor-payload system. Choosing $\boldsymbol{q} = [x_P, y_P, z_P, \ \alpha^1, \beta^1, \phi^1, \theta^1, \psi^1, \alpha^2, \beta^2, \phi^2, \theta^2, \ \psi^2]^T$ as the generalised coordinates will not only be convenient for controlling the payload trajectory but also help in extending to multi-vehicle situations. As a result, the Lagrangian $\mathfrak{L}$ is composed by subtraction of the kinetic and potential energies denoted by $\mathfrak{T}$ and $\mathfrak{U}$, as shown in the equations below:

$$\mathfrak{T} = \frac{1}{2}m_P(\boldsymbol{\dot{\xi}}_P)^T \cdot \boldsymbol{\dot{\xi}}_P + \frac{1}{2}m_Q^1(\boldsymbol{\dot{\xi}_Q^1})^T \cdot \boldsymbol{\dot{\xi}_Q^1} + \frac{1}{2}(\boldsymbol{\Omega^1})^T\boldsymbol{I_Q^1}\boldsymbol{\Omega^1}$$
$$\frac{1}{2}m_Q^2(\boldsymbol{\dot{\xi}_Q^2})^T \cdot \boldsymbol{\dot{\xi}_Q^2} + \frac{1}{2}(\boldsymbol{\Omega^2})^T\boldsymbol{I_Q^2}\boldsymbol{\Omega^2}$$
$$\mathfrak{U} = m_Pg\boldsymbol{\xi}_P \cdot \boldsymbol{E_3} + m_Q^1g\boldsymbol{\xi_Q^1} \cdot \boldsymbol{E_3} + m_Q^2g\boldsymbol{\xi_Q^2} \cdot \boldsymbol{E_3} \tag{3.21}$$
$$\mathfrak{L} = \mathfrak{T} - \mathfrak{U}.$$

Then the Euler-Lagrange equation becomes

$$\frac{d}{dt}\left(\frac{\partial\mathfrak{L}}{\partial\boldsymbol{\dot{q}}}\right) - \frac{\partial\mathfrak{L}}{\partial\boldsymbol{q}} = \mathbf{Q} \tag{3.22}$$

where $\mathfrak{L}$ is the Lagrangian, and the generalised force $\mathbf{Q}$ defined here is based on the choice of the generalised coordinates $\boldsymbol{q} \in \mathbb{R}^{13}$ and the the external conservative force $\boldsymbol{F}$ is $[\boldsymbol{F_Q^{KT}}, \boldsymbol{M_Q^{KT}}]^T \in \mathbb{R}^8$. However, the Euler-Lagrange equation cannot be directly utilised to calculate $\mathbf{Q}$ via D'Alembert's Principle as a transformation is required in advance.

Firstly, the thrust of each quadrotor should be transformed from the body frame into the $\boldsymbol{S_e}$ frame

$$\boldsymbol{F_Q^K} = F_z^K \boldsymbol{e_3^K}.$$

Define the unit orthogonal vectors of $\boldsymbol{S_b^K}$:

$$\boldsymbol{e_{\mathcal{I}}^K} = \boldsymbol{T_{e2b}^K} \boldsymbol{E_{\mathcal{I}}}, \mathcal{I} = 1, 2, 3. \tag{3.23}$$

Secondly, the torques of each quadrotor should be transformed from $\boldsymbol{S_b^K}$ into an appropriate frame in terms of Euler angles by velocity Jacobian matrix

$$\boldsymbol{M_Q^{\eta K}} = \boldsymbol{\jmath^{KT}} \boldsymbol{M_Q^K}.$$

According to the following identical equations of powerflow through the joint,

$$\boldsymbol{M_Q^{\eta}} \cdot \dot{\boldsymbol{\eta}} = \boldsymbol{M_Q^{\eta T}} \dot{\boldsymbol{\eta}} = \left(\boldsymbol{\jmath^T} \boldsymbol{M_Q} \right)^T \dot{\boldsymbol{\eta}} = \boldsymbol{M_Q} \cdot \boldsymbol{\jmath} \dot{\boldsymbol{\eta}} = \boldsymbol{M_Q} \cdot \boldsymbol{\Omega}.$$

Thus, the generalised force $\mathbf{Q}$ is presented by equation (3.24), where

$$\boldsymbol{Q_i} = \frac{\partial \left(\sum_{K=1}^2 \boldsymbol{F_Q^K} \cdot \boldsymbol{\xi_Q^K} + \boldsymbol{M_Q^{\eta K}} \cdot \boldsymbol{\eta^K} \right)}{\partial \boldsymbol{q_i}}, \boldsymbol{i} = 1, 2, ...8. \tag{3.24}$$

Applying the generalised forces and equation (3.21) to equation (3.22), the Euler-Lagrange equation becomes based on $\boldsymbol{G} \in \mathbb{R}^{13 \times 13}$ matrix denoted by a symmetric matrix, and the system model function f can then be rewritten as

$$G\ddot{q} = g(F, q, \dot{q}) = f(x, u.) \tag{3.25}$$

In the trajectory tracking control, the operating point is regarded as the balance situation where the Euler angles of both quadrotors are equal to zero and the cable angles are equal to the designed constants. The state x and the control vector input u of the system are defined as

$$x = \left[x_P, \dot{x}_P, y_P, \dot{y}_P, z_P, \dot{z}_P, \alpha^1, \dot{\alpha}^1, \beta^1, \dot{\beta}^1, \phi^1, \dot{\phi}^1, \theta^1, \right.$$
$$\left. \dot{\theta}^1, \psi^1, \dot{\psi}^1, \alpha^2, \dot{\alpha}^2, \beta^2, \dot{\beta}^2, \phi^2, \dot{\phi}^2, \theta^2, \dot{\theta}^2, \psi^2, \dot{\psi}^2 \right]^T \in \mathbb{R}^{26}$$
$$u = [F_z^1, M_x^1, M_y^1, M_z^1, F_z^2, M_x^2, M_y^2, M_z^2]^T \in \mathbb{R}^8 \tag{3.26}$$

and a non-linear state space equation is then given as

$$\dot{x} = f(x, u). \tag{3.27}$$

The nonlinear discrete dynamic model is

$$x_{k+1} = f(x_k, u_k). \tag{3.28}$$

3.3.3 Equilibrium Points and Linear Time Varying Model

To make use of the linear control strategy, equilibrium points and the corresponding linear models must be provided in advance. In general, the equilibrium points (x_{eq}, u_{eq}) are recognized as stationary points since they satisfy the following equation:

$$0 = f\left(x_{keq}, u_{keq}\right). \tag{3.29}$$

By utilising Taylor's series expansion, it is possible for equation (3.27) to be approximated by a linear equation around $(\boldsymbol{x_{keq}}, \boldsymbol{u_{keq}})$ and the linearised model is obtained in equation (3.30):

$$\ddot{\boldsymbol{q}} = f\left(\boldsymbol{x_{keq}}, \boldsymbol{u_{keq}}\right) + \left.\frac{\partial f}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{x_{keq}}} (\boldsymbol{x_k} - \boldsymbol{x_{keq}}) + \left.\frac{\partial f}{\partial \boldsymbol{u}_k}\right|_{\boldsymbol{u_{keq}}} (\boldsymbol{u_k} - \boldsymbol{u_{keq}}) \tag{3.30}$$

where, $\delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_{eq}$, $\delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_{eq}$, $\delta\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}} = \boldsymbol{G}\ddot{\boldsymbol{q}}$.

Equation (3.31) gives the linearised model below:

$$\boldsymbol{G}\ddot{\boldsymbol{q}} = \boldsymbol{M}^{-1}\left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_0} \delta\boldsymbol{x} + \boldsymbol{M}^{-1}\left.\frac{\partial f}{\partial \boldsymbol{u}}\right|_{\boldsymbol{u}_0} \delta\boldsymbol{u} \tag{3.31}$$

Furthermore, equation (3.31) can be transformed into the following discrete form:

$$\boldsymbol{x_{k+1}} = \boldsymbol{A}_k\boldsymbol{x_k} + \boldsymbol{B}_k\boldsymbol{u_k} \tag{3.32}$$

and $\boldsymbol{A_k} \in \mathbb{R}^{26\times26}$, $\boldsymbol{B_k} \in \mathbb{R}^{26\times8}$.

In this system, however, there are an infinite number of equilibrium points as well as the linear models, which mainly depend on the team formation heading angle α_F.
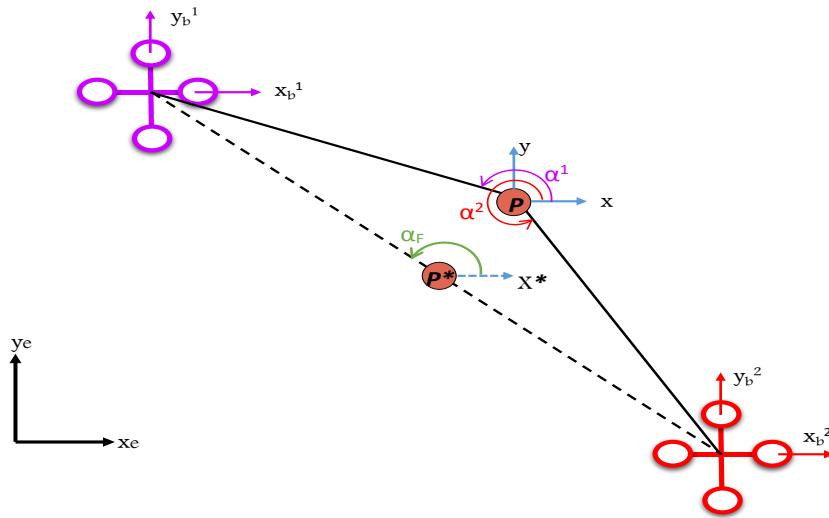
FIGURE 3.3: (Top View) Two quadrotors (Q^1 and Q^2) carrying a cable-suspended payload. P: unbalanced position of payload, P^*: stationary position of payload.

As can be seen in Figure 3.3, if the system remains stationary, where the quadrotors and the payload must stay in the same vertical plane (Q^1, Q^2 and P^*), then the formation heading angle can be denoted by α_F. Thus, the following relationships can by yielded:

$$\alpha_1 = \alpha_F$$
$$\alpha_2 = \alpha_F + \pi. \tag{3.33}$$

The other swing angles β^1 and β^2 are considered to keep staying close to a fixed constant operational point. Thus, equation (3.31) can be restated in the following linear-time-varying form:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_k\left(\alpha_F\right)\boldsymbol{x}_k + \boldsymbol{B}_k\left(\alpha_F\right)\boldsymbol{u}_k \tag{3.34}$$

where the generalised forces **Q** are given below in equation 3.35:

$$\mathbf{Q_1} = Fz^1 C(\phi^1) C(\psi_1) S(\theta^1) + Fz^2 C(\phi^2) C(\psi^2) S(\theta^2) + Fz^1 S(\phi^1) S(\psi^1) + Fz^2 S(\phi^2) S(\psi^2)$$

$$\mathbf{Q_2} = -Fz^1 C(\psi^1) S(\phi^1) - Fz^2 C(\psi^2) S(\phi^2) + Fz^1 C(\phi^1) S(\theta^1) S(\psi^1) + Fz^2 C(\phi^2) S(\theta^2) S(\psi^2)$$

$$\mathbf{Q_3} = Fz^1 C(\theta^1) C(\phi^1) + Fz^2 C(\theta^2) C(\phi^2)$$

$$\mathbf{Q_4} = -Fz^1 Lr(\beta^1)(C(\alpha^1 - \psi^1) S(\phi^1) + C(\phi^1) S(\theta^1) S(\alpha^1 - \psi^1))$$

$$\mathbf{Q_5} = Fz^1 Lr(C(\beta^1) C(\theta^1) C(\phi^1) + S(\beta^1)(-C(\phi^1) C(\alpha^1 - \psi^1) S(\theta^1) + S(\phi^1) S(\alpha^1 - \psi^1)))$$

$$\mathbf{Q_6} = M_x^1$$

$$\mathbf{Q_7} = M_y^1 C(\phi^1) - M_z^1 S(\phi^1)$$

$$\mathbf{Q_8} = -M_x^1 S(\theta^1) + C(\theta^1)(M_z^1 C(\phi^1) + M_y^1 S(\phi^1))$$

$$\mathbf{Q_9} = -Fz^2 LrC(\beta^2)(C(\alpha^2 - \psi^2) S(\phi^2) + C(\phi^2) S(\theta^2) S(\alpha^2 - \psi^2))$$

$$\mathbf{Q_{10}} = Fz^2 Lr(C(\beta^2) C(\theta^2) C(\phi^2) + S(\beta^2)(-C(\phi^2) C(\alpha^2 - \psi^2) S(\theta^2) + S(\phi^2) S(\alpha^2 - \psi^2)))$$

$$\mathbf{Q_{11}} = Mx^2$$

$$\mathbf{Q_{12}} = M_y^2 C(\phi^2) - M_z^2 S(\phi^2)$$

$$\mathbf{Q_{13}} = -M_x^2 S(\theta^2) + C(\theta^2)(M_z^2 1 C(\phi^2) + M_y^2 S(\phi^2))$$

$$(3.35)$$

and the **M** matrix can be defined as:

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} & \mathbf{0}_{5\times3} & \mathbf{M}_{14} & \mathbf{0}_{5\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times2} & \mathbf{M}_{23} & \mathbf{0}_{5\times2} & \mathbf{0}_{5\times3} \\ \mathbf{M}_{31} & \mathbf{0}_{5\times3} & \mathbf{0}_{5\times3} & \mathbf{M}_{34} & \mathbf{M}_{35} \end{bmatrix} \tag{3.36}$$

where, $\mathbf{M}_{11}$, $\mathbf{M}_{12}$, $\mathbf{M}_{14}$, $\mathbf{M}_{23}$, $\mathbf{M}_{31}$, $\mathbf{M}_{34}$ and $\mathbf{M}_{35}$ are defined as follows:

$$\mathbf{M}_{11} = \begin{bmatrix} m_P + m_{Q1} + m_{Q2} & 0 & 0 \\ 0 & m_P + m_{Q1} + m_{Q2} & 0 \\ 0 & 0 & m_P + m_{Q1} + m_{Q2} \\ -Lrm_{Q1} C(\beta^1) S(\alpha^1) & Lrm_{Q1} C(\alpha^1) C(\beta^1) & 0 \\ -Lrm_{Q1} C(\alpha^1) S(\beta^1) & -Lrm_{Q1} S(\alpha^1) S(\beta^1) & Lrm_{Q1} C(\beta^1) \end{bmatrix}$$

$$
\boldsymbol{M}_{12} = \begin{bmatrix}
-Lrm_{Q1}C(\beta^1)S(\alpha^1) & -Lrm_{Q1}C(\alpha^1)S(\beta^1) \\
Lrm_{Q1}C(\alpha^1)C(\beta^1) & -Lrm_{Q1}S(\alpha^1)S(\beta^1) \\
0 & Lrm_{Q1}C(\beta^1) \\
Lr^2 m_{Q1}C(\beta^1)^2 & 0 \\
0 & L_r^2 m_{Q1}
\end{bmatrix}
$$

$$
\boldsymbol{M}_{14} = \begin{bmatrix}
-Lrm_{Q2}C(\beta^2)S(\alpha^2) & -Lrm_{Q2}C(\alpha^2)S(\beta^2) \\
Lrm_{Q2}C(\alpha^2)C(\beta^2) & -Lrm_{Q2}S(\alpha^2)S(\beta^2) \\
0 & Lrm_{Q2}C(\beta^2)
\end{bmatrix}
$$

$$
\boldsymbol{M}_{23} = \begin{bmatrix}
I_x^1 & 0 & -I_x^1 S[\theta^1] \\
0 & I_y^1 C(\phi^1)^2 + I_z^1 S(\phi^1)^2 & (I_y^1 - I_z^1)C(\theta^1)C(\phi^1)S(\phi^1) \\
-I_x^1 S(\theta^1) & (I_y^1 - I_z^1)C(\theta^1)C(\phi^1)S(\phi^1) & I_x^1 S(\theta^1)^2 + C(\theta^1)^2(I_z^1 C(\phi^1)^2 + I_y^1 S(\phi^1)^2)
\end{bmatrix}
$$

$$
\boldsymbol{M}_{31} = \begin{bmatrix}
-Lrm_{Q2}C(\beta^2)S(\alpha^2) & Lrm_{Q2}C(\alpha^2)C(\beta^2) & 0 \\
-Lrm_{Q2}C(\alpha^2)S(\beta^2) & -Lrm_{Q2}S(\alpha^2)S(\beta^2) & Lrm_{Q2}C(\beta^2) \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix}
$$

$$
\boldsymbol{M}_{34} = \begin{bmatrix}
Lr^2 m_{Q2}C(\beta^2)^2 & 0 \\
0 & Lr^2 m_{Q2} \\
0 & 0 \\
0 & 0 \\
0 & 0
\end{bmatrix}
$$

$$
\boldsymbol{M}_{35} = \begin{bmatrix}
I_x^2 & 0 & -I_x^2 S(\theta^2) \\
0 & I_y^2 C(\phi^2)^2 + I_z^2 S(\phi^2)^2 & (I_y^2 - I_z^2)C(\theta^2)C(\phi^2)S(\phi^2) \\
-I_x^2 S(\theta^2) & (I_y^2 - I_z^2)C(\theta^2)C(\phi^2)S(\phi^2) & I_x^2 S(\theta^2)^2 + C(\theta^2)^2(I_z^2 C(\phi^2)^2 + I_y^2 S(\phi^2)^2)
\end{bmatrix}
$$

and the function $f(\boldsymbol{x}, \boldsymbol{u})$ is presented in:

$$f_1(\boldsymbol{x}, \boldsymbol{u}) = F_z^1 C(\phi^1) C(\psi^1) S(\theta^1) + F_z^2 C(\phi^2) C(\psi^2) S(\theta^2) + F_z^1 S(\phi^1) S(\psi^1) +$$

$$F_z^2 S(\phi^2) S(\psi^2) + Lr(-2m_{Q2} S(\alpha^1) S(\beta^1)(\dot{\alpha}^1)(\dot{\beta}^1) + m_{Q1} C(\alpha^1) C(\beta^1)$$

$$((\dot{\alpha}^1)^2 + (\dot{\beta}^1)^2) - 2m_{Q2} S(\alpha^2) S(\beta^2)(\dot{\alpha}^2)(\dot{\beta}^2) + m_{Q2} C(\alpha^2) C(\beta^2)((\dot{\alpha}^2)^2 + (\dot{\beta}^2)^2))$$

$$f_2(\boldsymbol{x}, \boldsymbol{u}) = -F_z^1 C(\psi^1) S(\phi^1) - F_z^2 C(\psi^2) S(\phi^2) + F_z^1 C(\phi^1) S(\theta^1) S(\psi^1) +$$

$$F_z^2 C(\phi^2) S(\theta^2) S(\psi^2) + Lr(2m_{Q1} C(\alpha^1) S(\beta^1)(\dot{\alpha}^1)(\dot{\beta}^1) + m_{Q1} C(\beta^1)$$

$$S(\alpha^1)((\dot{\alpha}^1)^2 + (\dot{\beta}^1)^2) + 2m_{Q2} C(\alpha^2) S(\beta^2)(\dot{\alpha}^2)(\dot{\beta}^2) +$$

$$m_{Q2} C(\beta^2) S(\alpha^2)((\dot{\alpha}^2)^2 + (\dot{\beta}^2)^2))$$

$$f_3(\boldsymbol{x}, \boldsymbol{u}) = -g(m_P + m_{Q1} + m_{Q2}) + F_z^1 C(\theta^1) C(\phi^1) + F_z^2 C(\theta^2) C(\phi^2) + Lrm_{Q1} S(\beta^1)$$

$$(\dot{\beta}^1)^2 + Lrm_{Q2} S(\beta^2)(\dot{\beta}^2)^2$$

$$f_4(\boldsymbol{x}, \boldsymbol{u}) = LrC(\beta^1)(-F_z^1(C(\alpha^1 - \psi^1) S(\phi^1) + C(\phi^1) S(\theta^1) S(\alpha^1 - \psi^1)) +$$

$$2Lrm_{Q1} S(\beta^1)(\dot{\alpha}^1)(\dot{\beta}^1))$$

$$f_5(\boldsymbol{x}, \boldsymbol{u}) = -Lr(C(\beta^1)(gm_{Q1} - F_z^1 C(\theta^1) C(\phi^1)) + F_z^1 S(\beta^1)(C(\phi^1) C(\alpha^1 - \psi^1))$$

$$S(\theta^1) - S(\phi^1) S(\alpha^1 - \psi^1)) + Lrm_{Q1} C(\beta^1) S(\beta^1)(\dot{\alpha}^1)^2)$$

$$f_6(\boldsymbol{x}, \boldsymbol{u}) = M_x^1 + (-I_y^1 + I_z^1) C(\phi^1) S(\phi^1)(\dot{\theta}^1)^2 + C(\theta^1)(I_x^1 + (I_y^1 - I_z^1) C(2\phi^1))$$

$$(\dot{\theta}^1)(\dot{\psi}^1) + (I_y^1 - I_z^1) C(\theta^1)^2 C(\phi^1) S(\phi^1)(\dot{\psi}^1)^2$$

$$f_7(\boldsymbol{x}, \boldsymbol{u}) = M_y^1 C(\phi^1) - M_z^1 S(\phi^1) + (I_y^1 - I_z^1) S(2\phi^1)(\dot{\theta}^1)(\dot{\phi}^1) + C(\theta^1)$$

$$(\dot{\psi}^1)(-(I_x^1 + (I_y^1 - I_z^1) C(2\phi^1))(\dot{\phi}^1) + S(\theta^1)(I_x^1 - I_z^1 C(\phi^1)^2 - I_y^1 S(\phi^1)^2)(\dot{\psi}^1))$$

$$f_8(\boldsymbol{x}, \boldsymbol{u}) = -M_x^1 S(\theta^1) + C(\theta^1)(M_z^1 C(\phi^1) + M_y^1 S(\phi^1)) + (I_y^1 - I_z^1) C(\phi^1) S(\theta^1)$$

$$S(\phi^1)(\dot{\theta}^1)^2 + (-I_y^1 + I_z^1) C(\theta^1)^2 S(2\phi^1)(\dot{\phi}^1)(\dot{\psi}^1) + (\dot{\theta}^1)(C(\theta^1)$$

$$(I_x^1 + (-I_y^1 + I_z^1) C(2\phi^1))(\dot{\phi}^1) + S(2\theta^1)(-I_x^1 + I_z^1 C(\phi^1)^2 + I_y^1 S(\phi^1)^2)(\dot{\psi}^1))$$

$$f_9(\boldsymbol{x}, \boldsymbol{u}) = LrC(\beta^2)(-F_z^2(C(\alpha^2 - \psi^2) S(\phi^2) + C(\phi^2) S(\theta^2) S(\alpha^2 - \psi^2)) + 2Lrm_{Q2} S(\beta^2)(\dot{\alpha}^2)(\dot{\beta}^2))$$

$$f_{10}(\boldsymbol{x}, \boldsymbol{u}) = -Lr(C(\beta^2)(gm_{Q2} - F_z^2 C(\theta)^2) C(\phi^2)) + F_z^2 S(\beta^2)(C(\phi^2) C(\alpha^2 - \psi^2)$$

$$S(\theta^2) - S(\phi^2) S(\alpha^2 - \psi^2)) + Lrm_{Q2} C(\beta^2) S(\beta^2)(\alpha^2)^2)$$

$$f_{11}(\boldsymbol{x}, \boldsymbol{u}) = M_x^2 + (-I_y^2 + I_z^2) C[\phi^2] S(\phi^2)](\dot{\theta}^2)^2 + C(\theta^2)(I_x^2 + (I_y^2 - I_z^2) C(2\phi^2))$$

$$(\dot{\theta}^2)(\dot{\psi}^2) + (I_y^2 - I_z^2) C(\theta^2)^2 C(\phi^2) S(\phi^2)(\dot{\psi}^2)^2$$

$$f_{12}(\boldsymbol{x}, \boldsymbol{u}) = M_y^2 C(\phi^2) - M_z^2 S(\phi^2) + (I_y^2 - I_z^2) S(2\phi^2)(\dot{\theta}^2)(\dot{\phi}^2) + C(\theta^2)$$

$$(\dot{\psi}^2)(-(I_x^2 + (I_y^2 - I_z^2) C(2\phi^2)(\dot{\phi}^2) + S(\theta^2)(I_x^2 - I_z^2 C(\phi^2)^2 - I_y^2 S(\phi^2)^2)(\dot{\psi}^2))$$

$$f_{13}(\boldsymbol{x}, \boldsymbol{u}) = -M_x^2 S(\theta^2) + C(\theta^2)(M_z^2 C(\phi^2) + M_y^2 S(\phi^2) + (I_y^2 - I_z^2) C(\phi^2) S(\theta^2)$$

$$S(\phi^2)(\dot{\theta}^2)^2 + (-I_y^2 + I_z^2) C(\theta^2)^2 S(2\phi^2)(\dot{\phi}^2)(\dot{\psi}^2) + (\dot{\theta}^2)(C(\theta^2)$$

$$(I_x^2 + (-I_y^2 + I_z^2) C(2\phi^2)(\dot{\phi}^2) + S(2\theta^2)(-I_x^2 + I_z^2 C(\phi^2)^2 + I_y^2 S(\phi^2)^2)(\dot{\psi}^2))$$

$$(3.37)$$

Linearized model matrices are presented as follows:

$$
A = \begin{pmatrix}
0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.3066 & 0 & 0 & 0 & 3.5933 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.3066 & 0 & 0 & 0 & 3.5933 & 0 \\
0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 4.900 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4.900 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.3066 & 0 & 0 & 0 & 3.5933 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.3066 & 0 & 0 & 0 & -3.5933 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -9.4495 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.5199 & 0 & 0 & 0 & 6.9296 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4.3677 & 0 & 0 & 0 & -9.4495 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5.0817 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 6.9296 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -9.4495 & 0 & 0 & 0 & 16.5718 & 0 & 0 & 2.5199 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4.3677 & 0 & 0 & 0 & 9.4495 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
$$

$$
B = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.7602 & 0 & 0 & 0 & -0.7602 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.7602 & 0 & 0 & 0 & -0.7602 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1.0752 & 0 & 0 & 0 & -1.0752 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 434.7826 & 0 & 33.4448 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 357.1429 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 219.9489 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.0752 & 0 & 0 & 0 & 1.0752 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 434.7826 & 0 & -33.4448 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 357.1429 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 219.9489
\end{pmatrix}
$$

3.4 Summary

In this chapter, a single quadrotor carrying a payload by a cable is modelled first. The payload position, the quadrotor attitudes and the swing angles are chosen as the states. A transformation matrix is involved to convert the system coordination from the body frame to the initial frame in order to represent the generalised forces. The non-linear dynamic model is derived first, then the equilibrium point is found, and the system is linearised.

Two quadrotors carrying a payload by two cables are modelled next. The modelling procedure is the same as the first one. Only the dimensions of states and inputs are increased.

The main contribution to the modelling process is the selection of the system states. If quadrotor positions are chosen as the states, instead of payload position, the dimension of the state vector becomes larger and the dynamic models become more complex. Due to the model complexities, the Wolfram Mathematica software is used to produce the mathematic model in order to guarantee its correctness. The linearised models will be used in the following chapters. The non-linear model will be used in Chapter Six, where a non-linear MPC controller will be developed.

Chapter 4

Linear Optimal Controllers

4.1 Introduction

In the previous chapter, the dynamic models were established. As the models had high non-linearity, it was difficult to develop non-linear controllers from them, and the performance could not be optimised under most existing non-linear control techniques. This chapter aims to develop linear optimal controllers for the linearised dynamic systems.

The first controller is defined by a linear quadratic tracking controller LQR. This controller is applied to both models in order to be tested in comparison with a classic PD controller. Furtheremore, an iterative LQR controller is next developed to improve the performance and handle the high nonlinearity of the models.

The LQR controller is able to optimise the control performance. However, the changes in the operating point when non-linearity is high, and the changes induced by the slung load for linearised systems should not be ignored. The iterative LQR is able to reduce the changes in the operation point via multiple iterations. In this chapter, the ILQR optimal controllers with quadratic approximation are developed via dynamic programming approach for the transportation task. The design of the ILQR controller is based on the LQR method. The control objective is to control the point mass payload to follow a desired trajectory in position and attitude.

The test of system performance is conducted in various aggressive trajectories to demonstrate the complexities and challenges of the nonlinear system.

The ILQR controller is derived in the next section. When the iteration loops once, the ILQR controller becomes an LQR controller. In the simulation section, the performance of both the LQR and ILQR controllers is tested.

4.2 LQR controller

Linear Quadratic Regulator LQR is a linear state feedback optimal controller. It is presented based on dynamic model linearization of the quadrotor with cable-suspended load system in order to achieve minimum cost of the desired parameters. The error minimisation of the dynamic model is enforced by the convenient parameters of weight matrices [13] using cost objective function J of the form

$$J = \int_{t_o}^{t_f} \frac{1}{2}((\boldsymbol{x} - \boldsymbol{x}^*)^T Q(\boldsymbol{x} - \boldsymbol{x}^*) + \boldsymbol{u}^T R \boldsymbol{u}) dt \tag{4.1}$$

Where the initial and final time of the control horizon are t_o and t_f, matrices $Q \geq 0$ and $R > 0$ are the cost of the state $\boldsymbol{x}$ and control input $\boldsymbol{u}$ gain of the linear system represented in state space as follows

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}, \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \tag{4.2}$$

The goal is to minimize the cost function J via a calculated control input

$$\boldsymbol{u}^\star = -K\boldsymbol{x} = -R^{-1}\boldsymbol{B}^T P \boldsymbol{x} \tag{4.3}$$

Where P can be calculated from the continuous Ricatti equation

$$\dot{P}(t) + P(t)\boldsymbol{A} + \boldsymbol{A}^T P(t) - P(t)\boldsymbol{B}R^{-1}\boldsymbol{B}^T P(t) + Q = 0 \tag{4.4}$$

Consequently, the state feedback optimal control gain K can be calculated using the following formula

$$K = lqr(\boldsymbol{A}, \boldsymbol{B}, Q, R) \tag{4.5}$$

The LQR controller is designed by choosing positive parameters for Q and R matrices to determined the desired thrust and orientations. This controller is presented to estimate state feedback tuning parameter, which is similar to individually tuning as in PD controller parameters[109].

4.3 Iterative LQR Controller

The objective of this thesis is to develop an ILQR optimal controller, which iteratively linearises the non-linear dynamic model and cost function around the nominal optimal result. Then it implements the LQR technique in order to calculate the optimal feedback control. Initially, a nominal control sequence and the corresponding state sequence represented by $\boldsymbol{x_k}$ and $\boldsymbol{u_k}$, respectively, are produced. The nominal state is acquired from applying $\boldsymbol{u_k}$ to the open loop dynamical model iteratively. Through each iteration, the improved sequence $\boldsymbol{u_k}$ is obtained by linearising the nonlinear dynamics of the system around the nominal control $\boldsymbol{u_k}$ and state $\boldsymbol{x_k}$. Then by keeping iteratively refined, the modified LQR problem is solved and the convergence is achieved due to the control $\delta\boldsymbol{u_k}$ and state $\delta\boldsymbol{x_k}$ deviations from the nominal [42] and [43]. The discrete-time non-linear dynamical model is

$$\boldsymbol{x_{k+1}} = f(\boldsymbol{x_k}, \boldsymbol{u_k}). \tag{4.6}$$

The quadratic form cost function is

$$J = \frac{1}{2}(\boldsymbol{x_N} - \boldsymbol{x_N^*})^T Q_f (\boldsymbol{x_N} - \boldsymbol{x_N^*})$$
$$+ \frac{1}{2}\sum_{k=0}^{N-1}((\boldsymbol{x_k} - \boldsymbol{x_k^*})^T Q (\boldsymbol{x_k} - \boldsymbol{x_k^*}) + \boldsymbol{u_k^T} R \boldsymbol{u_k}) \tag{4.7}$$

where the final and target states are denoted by $\boldsymbol{x_N}$ and $\boldsymbol{x_N^*}$ with $\boldsymbol{N}$ steps respectively. The final state cost weighting matrix Q_f is a semidefinite symmetric positive matrix and R is a positive definite control cost matrix.

The linearised system is

$$\delta\boldsymbol{x_{k+1}} = \boldsymbol{A_k}\delta\boldsymbol{x_k} + \boldsymbol{B_k}\delta\boldsymbol{u_k} \tag{4.8}$$

where the matrices $\boldsymbol{A_k}$ and $\boldsymbol{B_k}$ are denoted by the Jacobians. These are evaluated along $\boldsymbol{x_k}$ and $\boldsymbol{u_k}$ with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$ respectively. Solve the iterative LQR problem by computing the second order Taylor of the cost J based on the linearised model (4.8).

$$J = \frac{1}{2}(\boldsymbol{x_N} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*})^T Q_f (\boldsymbol{x_N} + \delta\boldsymbol{x_N} - \boldsymbol{x_k^*})$$
$$+ \frac{1}{2}\sum_{k=0}^{N-1}(\boldsymbol{x_k} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*})^T Q (\boldsymbol{x_k} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*}) \tag{4.9}$$
$$+ (\boldsymbol{u_k} + \delta\boldsymbol{u_k})^T R (\boldsymbol{u_k} + \delta\boldsymbol{u_k}).$$

Based on a constraint added to the cost equation (4.9), the value function is formed as

$$V = \frac{1}{2}(\boldsymbol{x_N} + \delta\boldsymbol{x_N} - \boldsymbol{x_N^*})^T Q_f (\boldsymbol{x_N} + \delta\boldsymbol{x_N} - \boldsymbol{x_k^*})$$
$$+ \frac{1}{2}\sum_{k=0}^{N-1}(\boldsymbol{x_k} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*})^T Q (\boldsymbol{x_k} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*})$$
$$+ (\boldsymbol{u_k} + \delta\boldsymbol{u_k})^T R (\boldsymbol{u_k} + \delta\boldsymbol{u_k})$$
$$+ \delta\lambda_{k+1}^T (\boldsymbol{A_k}\delta\boldsymbol{x_k} + \boldsymbol{B_k}\delta\boldsymbol{u_k} - \delta\boldsymbol{x_{k+1}})$$

where the Lagrange multiplier is denoted by $\delta\lambda_{k+1}^T$. The Hamiltonian function is a first step for the implementation of the optimal control $\delta\boldsymbol{u_k}$ represented as

$$
\begin{aligned}
\boldsymbol{H_k} =& (\boldsymbol{x_k} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*})^T Q(\boldsymbol{x_k} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*}) \\
&+ (\boldsymbol{u_k} + \delta\boldsymbol{u_k})^T R(\boldsymbol{u_k} + \delta\boldsymbol{u_k}) \\
&+ \delta\lambda_{k+1}^T (\boldsymbol{A_k}\delta\boldsymbol{x_k} + \boldsymbol{B_k}\delta\boldsymbol{u_k}).
\end{aligned}
$$

The required derivatives of the Hamiltonian function according to the minimum approval of the value equation are:

$$
\frac{\partial\boldsymbol{H_k}}{\partial(\delta\boldsymbol{x_k})} = \delta\lambda_k, \frac{\partial\boldsymbol{H_k}}{\partial(\delta\boldsymbol{u_k})} = 0, \frac{\partial\boldsymbol{H_k}}{\partial(\delta\boldsymbol{x_N})} = \delta\lambda_{\boldsymbol{N}}.
$$

The result costate equation is

$$
\delta\lambda_{\boldsymbol{k}} = \boldsymbol{A_k^T}\delta\lambda_{\boldsymbol{k+1}} + Q(\boldsymbol{x_k} + \delta\boldsymbol{x_k} - \boldsymbol{x_k^*}). \tag{4.10}
$$

The stationary condition of the Hamiltonian function is

$$
R(\boldsymbol{u_k} + \delta\boldsymbol{u_k}) + \boldsymbol{B_k^T}\delta\lambda_{\boldsymbol{k+1}} = 0. \tag{4.11}
$$

The boundary condition is

$$
\delta\lambda_N = Q_f(\boldsymbol{x_N} + \delta\boldsymbol{x_N} - \boldsymbol{x_N^*}). \tag{4.12}
$$

From the boundary equation, we assume that

$$
\delta\lambda_{\boldsymbol{k}} = S_{\boldsymbol{k}}\delta\boldsymbol{x_k} + \upsilon_{\boldsymbol{k}} \tag{4.13}
$$

where the boundary conditions are

$$S_{\boldsymbol{N}} = Q_f, \upsilon_{\boldsymbol{N}} = Q_f(\boldsymbol{x_N} - \boldsymbol{x_N^*}). \tag{4.14}$$

Based on the boundary equation (4.11) and using (4.13), the optimal control error equation is presented as

$$\delta\boldsymbol{u_k} = -R^{-1}\boldsymbol{B_k^T}\delta\lambda_{\boldsymbol{k+1}} - \boldsymbol{u_k}. \tag{4.15}$$

By solving the equations (4.8), (4.11) and (4.13), The thrust and torque control error equations of the system are considered in the following:

$$\delta\boldsymbol{u_k} = -K\delta\boldsymbol{x_k} - K_\upsilon \upsilon_{\boldsymbol{k+1}} - K_u\boldsymbol{u_k}. \tag{4.16}$$

Consequently,

$$K = (\boldsymbol{B_k^T}S_{\boldsymbol{k+1}}\boldsymbol{B_k} + R)^{-1}\boldsymbol{B_k^T}S_{\boldsymbol{k+1}}\boldsymbol{A_k} \tag{4.17}$$

$$K_\upsilon = (\boldsymbol{B_k^T}S_{\boldsymbol{k+1}}\boldsymbol{B_k} + R)^{-1}\boldsymbol{B_k^T} \tag{4.18}$$

$$K_u = (\boldsymbol{B_k^T}S_{\boldsymbol{k+1}}\boldsymbol{B_k} + R)^{-1}R. \tag{4.19}$$

The backward recursion equations used to solve the entire sequences $S_{\boldsymbol{k}}$ and $\upsilon_{\boldsymbol{k}}$ are

$$S_{\boldsymbol{k}} = \boldsymbol{A_k^T}S_{\boldsymbol{k+1}}(\boldsymbol{A_k} - \boldsymbol{B_k}K) + Q \tag{4.20}$$

$$\upsilon_{\boldsymbol{k}} = (\boldsymbol{A_k} - \boldsymbol{B_k}K)^T\upsilon_{\boldsymbol{k+1}} - K^TR\boldsymbol{u_k} + Q(\boldsymbol{x_k} - \boldsymbol{x_k^*}) \tag{4.21}$$

Symbol	Definition	Value	Units
I_x	Roll Inertia	4.4×10^-3	$kg.m^2$
I_y	Pitch Inertia	4.4×10^-3	$kg.m^2$
I_z	Yaw Inertia	8.8×10^-3	$kg.m^2$
m_Q	Mass	0.5	kg
m_L	Mass	0.2	kg
g	Gravity	9.81	m/s^2
l	Arm Length	0.17	m
L	Cable Length	1	m
I_r	Rotor Inertia	4.4×10^-5	$kg.m^2$

TABLE 4.1: Quadrotor Parameters

where the gains K and K_u rely on the Riccati equation while the gain K_v is reliant on auxiliary sequence (4.21).

The improved nominal control can be found in

$$\boldsymbol{u_k^*} = \boldsymbol{u_k} + \delta \boldsymbol{u_k}. \tag{4.22}$$

4.4 LQR and PD Simulation Results

A MATLAB simulator of a quadrotor with a cable-suspended load was implemented to test the stability of the proposed controller. Table 4.1 shows the parameters used in this simulation [109].

The ILQR controller with one iteration was implemented as an LQR controller. In order to show how the LQR performs in a tracking task, the proposed LQR controller was tested and the results were compared with a PD controller. The error minimisation of the tracking control was enforced by the parameters of weight matrices Q and R [13]. The weight matrices for the system based on the generalised coordinates sequence and the control sequence are $\boldsymbol{q} = [x_P, \dot{x}_P, y_P, \dot{y}_P, z_P, \dot{z}_P, \alpha, \dot{\alpha}, \beta, \dot{\beta}, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$ and $\boldsymbol{u} = [F_z, M_x, M_y, M_z]^T$ in the form of diagonal matrices as

$$Q = Q_f = diag([0.039, 5, 0.039, 5, 10, 50, 1.44, 0.00001,$$

$$0.65, 0.035, 0.65, 0.035, 1, 1, 1, 1])$$

$$R = diag([10, 10, 1, 1]).$$

Applying these cost matrices, the following state feedback controller parameters are obtained

$$K = \begin{pmatrix} 0.716 & 0.0624 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.716 & 0.0624 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3.162 & 7.29 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.806 & 0.205 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.806 & 0.205 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.2 & 0.145 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0.1043 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0.1043 \end{pmatrix}$$

$$(4.23)$$

The first simulation test of the LQR controller is to track a 3D square trajectory shown in red color in Figure 4.4 with a payload height between $0.2m$ and $1.2m$. The rotation is $\phi^* = \theta^* = \psi^* = 0^o$ and the desired swing angles are $\alpha^* = 90^o$ and $\beta^* = 0^o$.

The blue line shown in all the figures is clarified to describe the system behaviour when employing the LQR tracking controller. The tracking performance is illustrated in Figures 4.1-4.6. Figure 4.1 shows the payload positions performance, while Figures 4.2 and 4.3 illustrate the swing angles and quadrotor attitude performance respectively. These results describe a stable performance with a small steady-state error for payload position and quadrotor and load attitudes. The vehicle and load angles are stabilised between $90° \leq \alpha \leq 91.5°$, $-4.5° \leq \beta \leq 4.5°$, $-10° \leq \phi \leq 10°$, $-4.5° \leq \theta \leq 4.5°$ and $-0.3° \leq \psi \leq 0.3°$.

RMSE	$x_P(m)$	$y_P(m)$	$z_P(m)$
LQR	0.026	0.0238	0.0123
PD	0.093	0.106	0.086

TABLE 4.2: Payload RMSE values for 3-doors trajectory under LQR and PD Controllers

Figures 4.4, 4.5 and 4.6 show the load path to pass through the three doors placed in different locations. These results are shown in blue in Figures 4.3 and 4.4 and in green in Figure 4.5.

The load position RMSE values for the 3-doors path using single quadrotor was demonstrated in Table 4.2
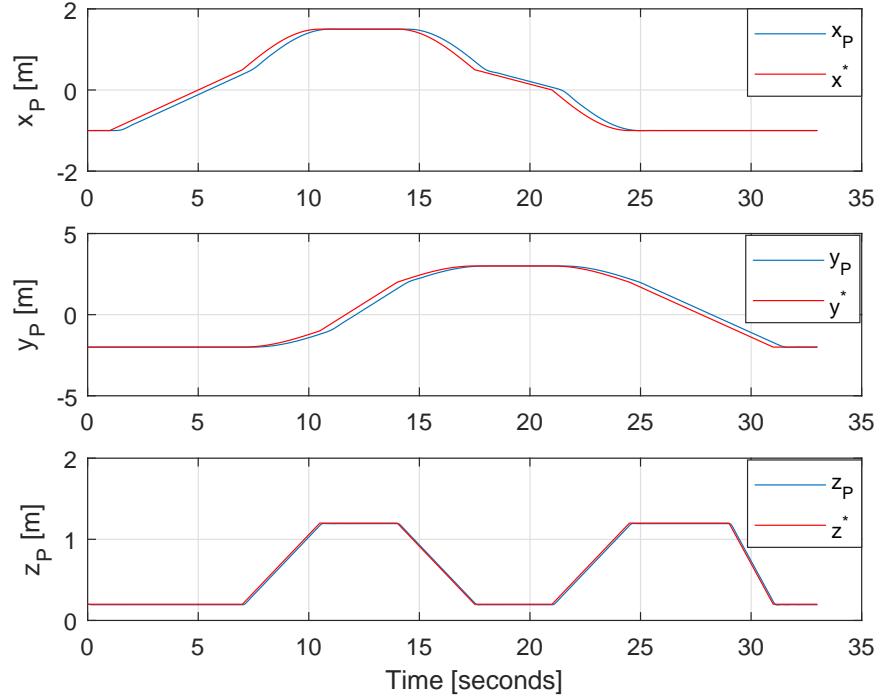
FIGURE 4.1: Payload positions, LQR Controller
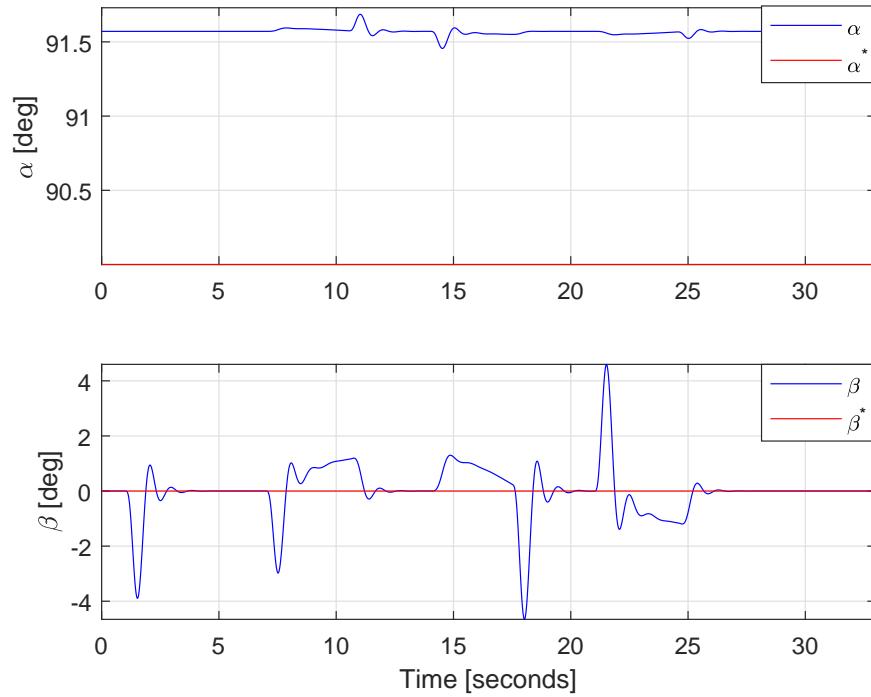
FIGURE 4.2: Rope angles, LQR Controller
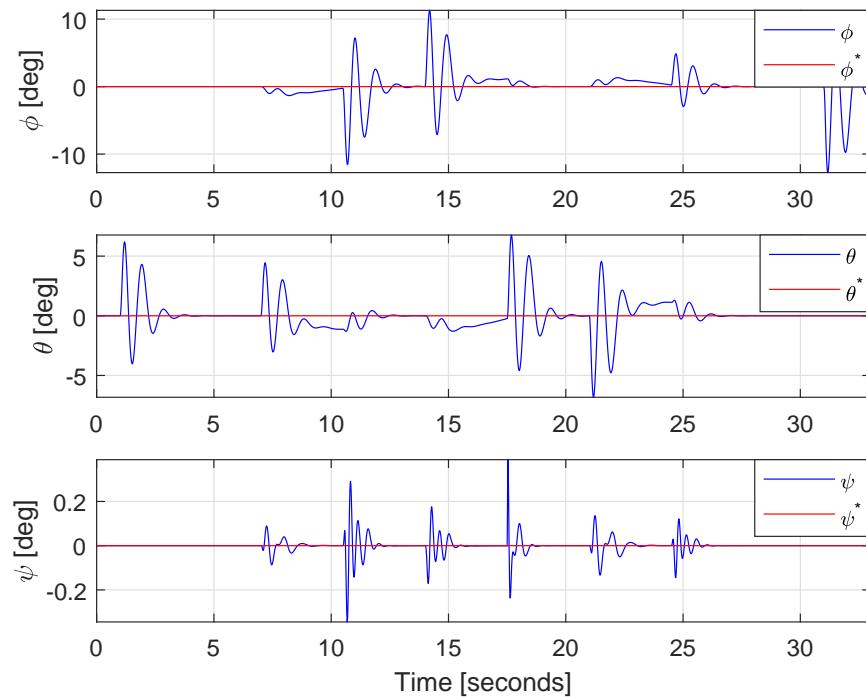
FIGURE 4.3: Quadrotor angles, LQR Controller
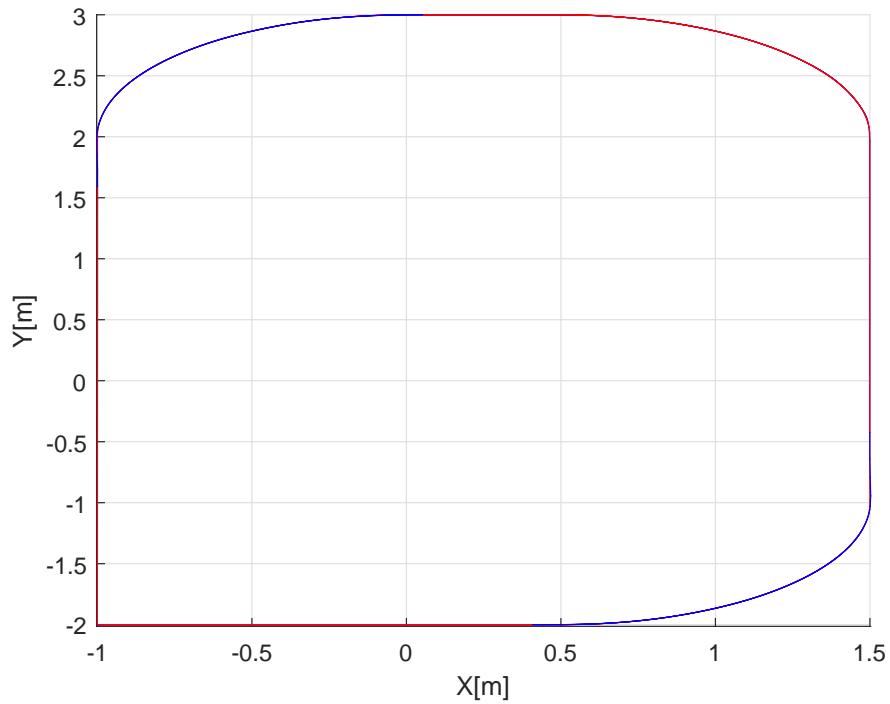
FIGURE 4.4: Payload position in two dimensions, LQR Controller
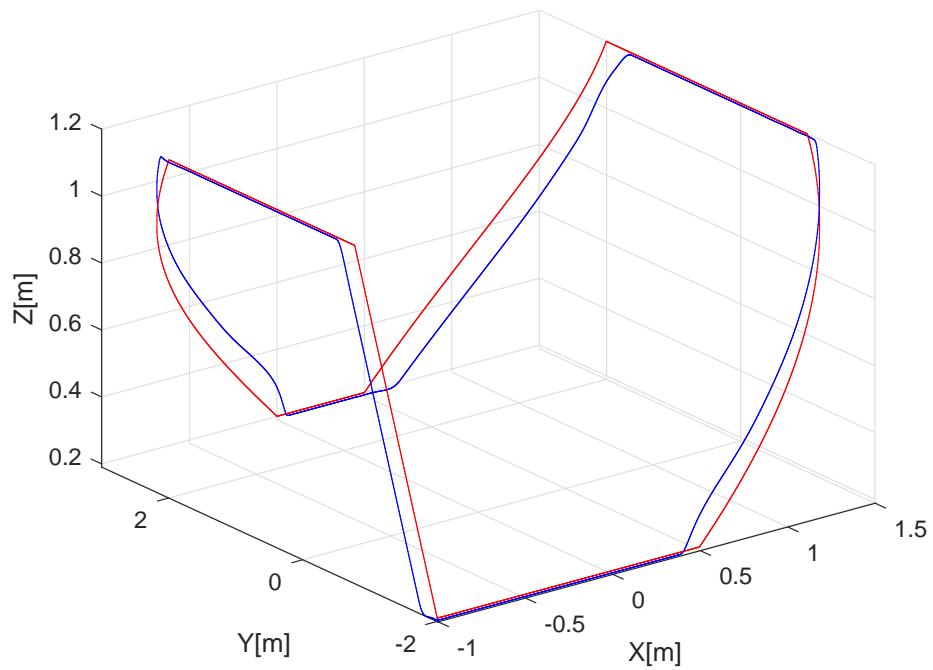
FIGURE 4.5: Payload position in three dimensions, LQR Controller
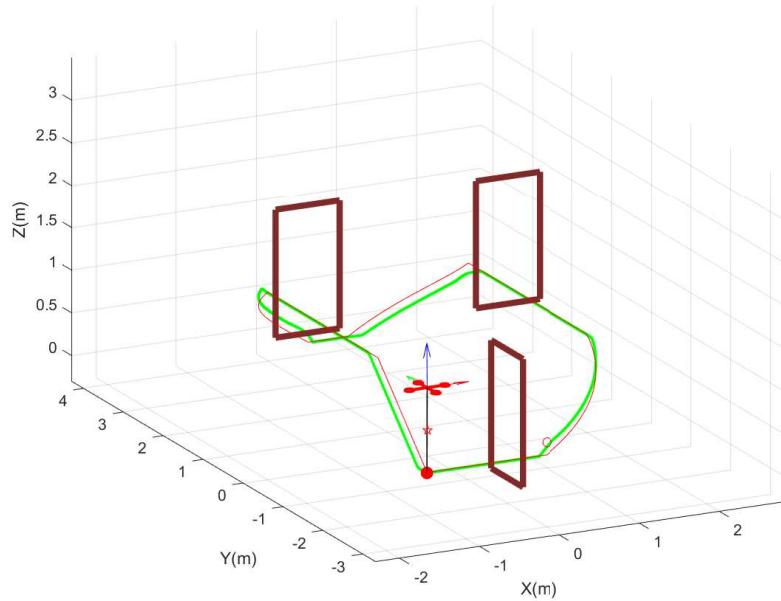
FIGURE 4.6: Payload position animation in three dimensions, LQR Controller

A simulation of a PD controller was tested in a transporting task for the comparison purpose. In the tracking task with the same desired trajectory, the results with the PD controller are illustrated in Figures 4.7 - 4.12. Figure 4.7 shows the payload positions performance using the PD controller. The quadrotor attitude angles and swing angles performance are illustrated in Figures 4.8 and 4.9. These angles are stabilised between $89.6° \leq \alpha \leq 90°$, $-2° \leq \beta \leq 2°$, $-2° \leq \phi \leq 2°$, $-1° \leq \theta \leq 1°$ and $-0.02° \leq \psi \leq 0.04°$. The load position plots in Figures 4.10 and 4.11 and the 3D trajectory in Figure 4.12 show that the payload in the tracking task cannot pass the three doors properly when the PD controller is used. It is obvious that the performance of the LQR controller is better than that of the PD controller in terms of steady-state error.
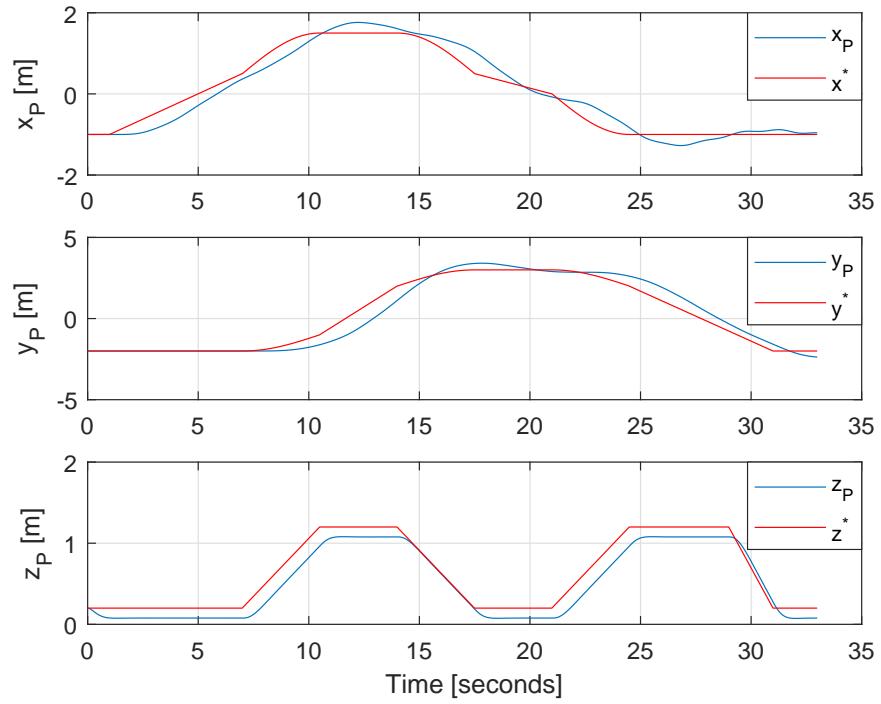
FIGURE 4.7: Payload positions, PD Controller
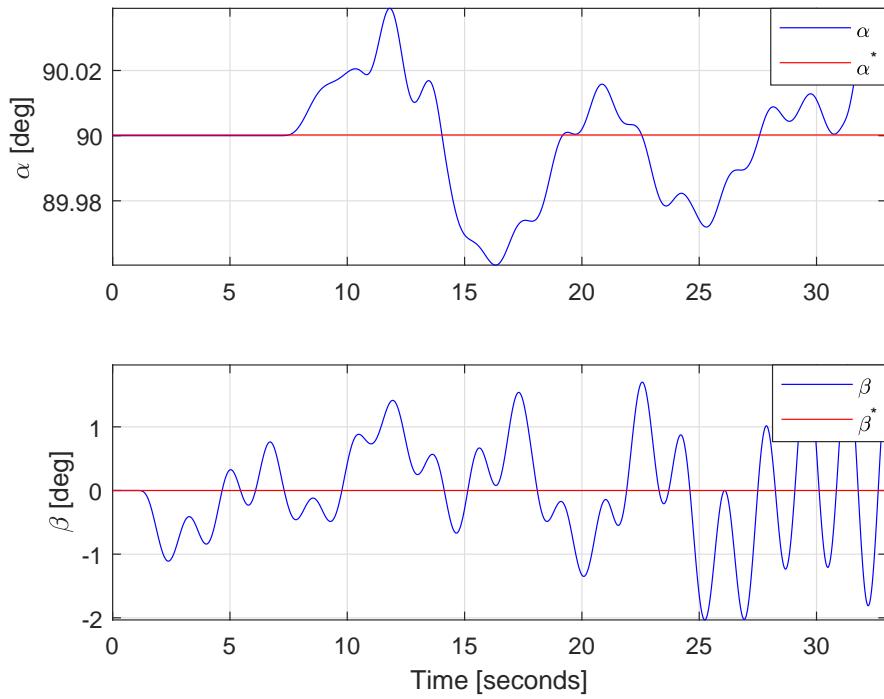
FIGURE 4.8: Rope angles, PD Controller
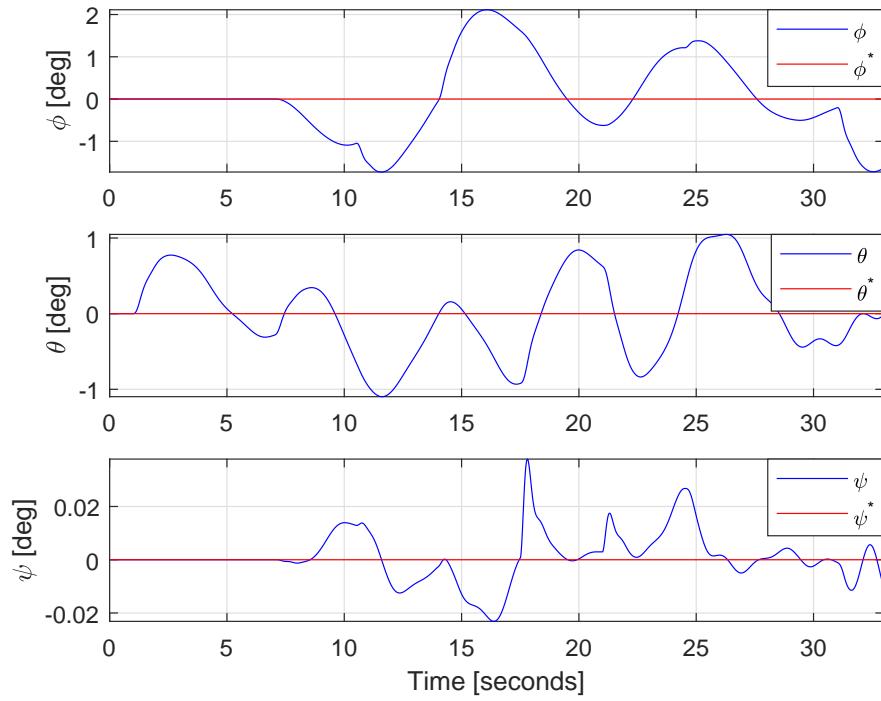
FIGURE 4.9: Quadrotor angles, PD Controller
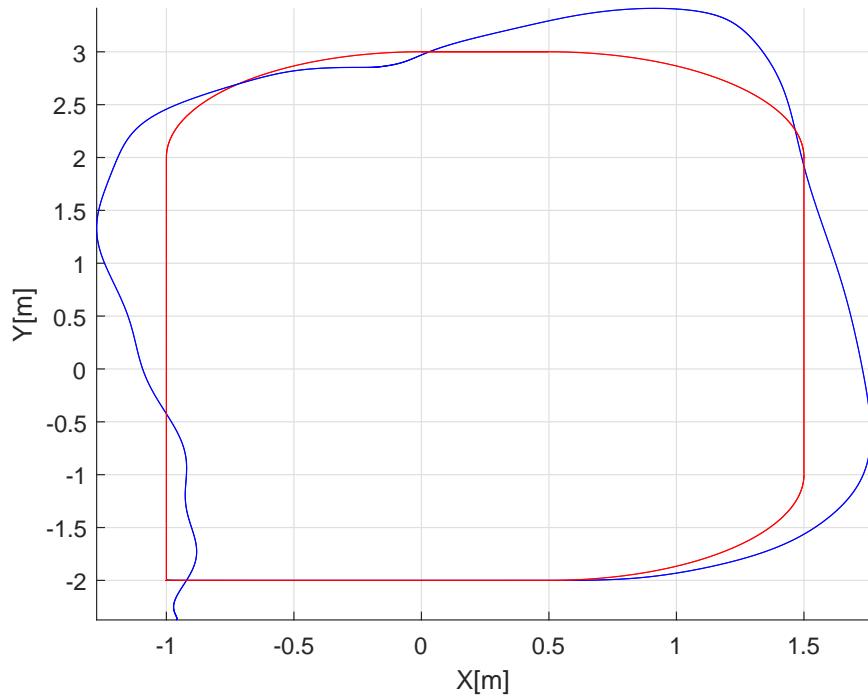
FIGURE 4.10: Payload position in two dimensions, PD Controller
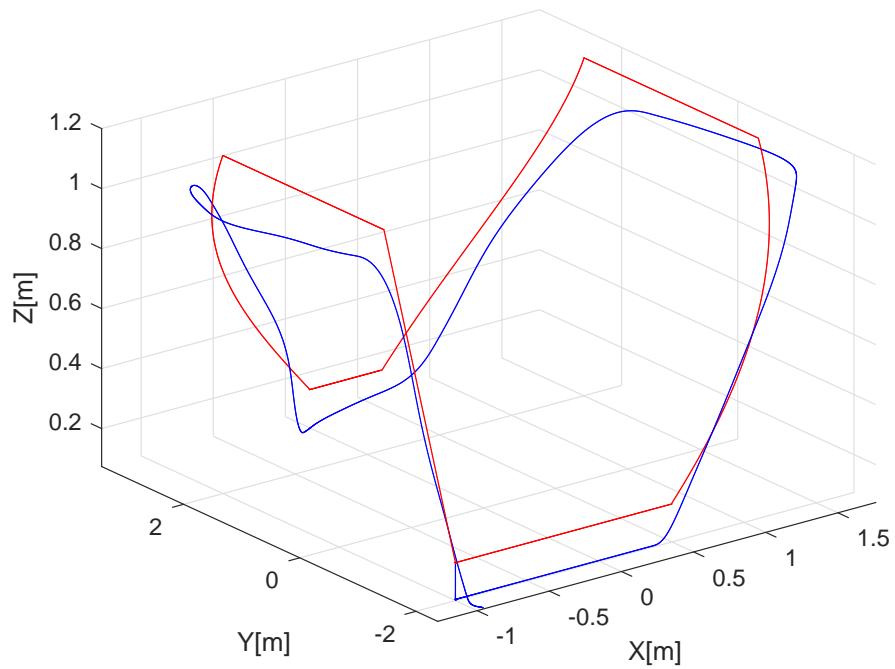
FIGURE 4.11: Payload position in three dimensions, PD Controller
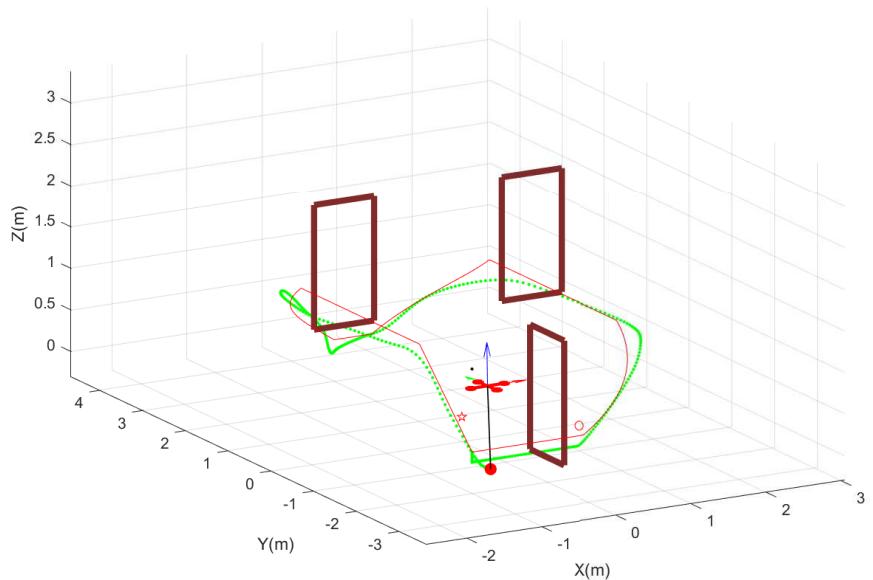
FIGURE 4.12: Payload position animation in three dimensions, PD Controller

4.5 ILQR Simulation Results for Single Quadrotor

In this section, the simulation of the ILQR controller is tested. It is compared with the LQR controller to see how nonlinearity is being handled. The first tracking trajectory is presented by an eight shape with $x^* = sin(2\pi/t)$, $y^* = sin(2\pi/t)$, $z^* = 0.1 * t$, where t is started from 0 with a sampling time of 0.02 sec to 40 sec. The tracking results are compared with the results of the LQR controller.

The second scenario is to track more aggressive trajectories in order to make the stability more challenging during the transporting task. A spiral trajectory is used $x^* = sin(2\pi/t), y^* = cos(2\pi/t), z^* = 0.1 * t.$

The performance of the first desired trajectory for the quadrotor and load are illustrated in Figures 4.13-4.16, where the payload position performance comparison between the LQR and the ILQR controllers is shown in Figure 4.13. It can be clearly seen that the load position using the ILQR controller in the fourth iteration, represented by a blue colour line, is closer to follow the desired path than the LQR controller represented in a black line, which means that the ILQR overcomes the system's high nonlinearity very effectively. The quadrotor attitude simulation results are displayed in Figure 4.14 and the swing angles results are illustrated in Figure 4.15. These orientations show fast steady results compared with those of the LQR controller. The desired trajectory in three dimensions is clarified in Figure 4.16.

In general, the load position for the eight desired path in two and three dimensions is clarified in Figure 4.16, where the first iteration is performed by the LQR tracking controller while the fourth iteration belongs to the ILQR controller.
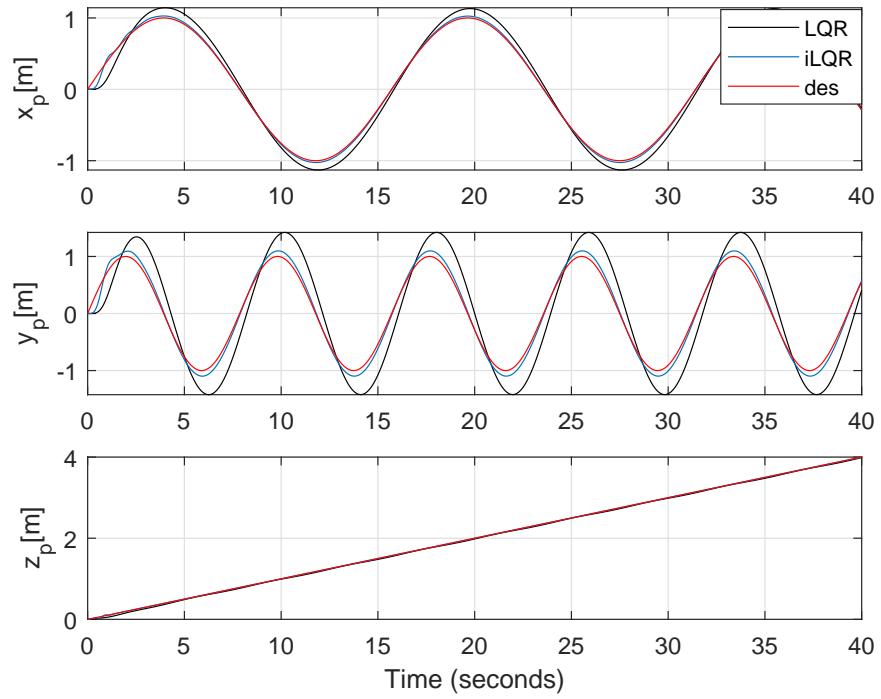
FIGURE 4.13: Payload position using the LQR and ILQR controllers
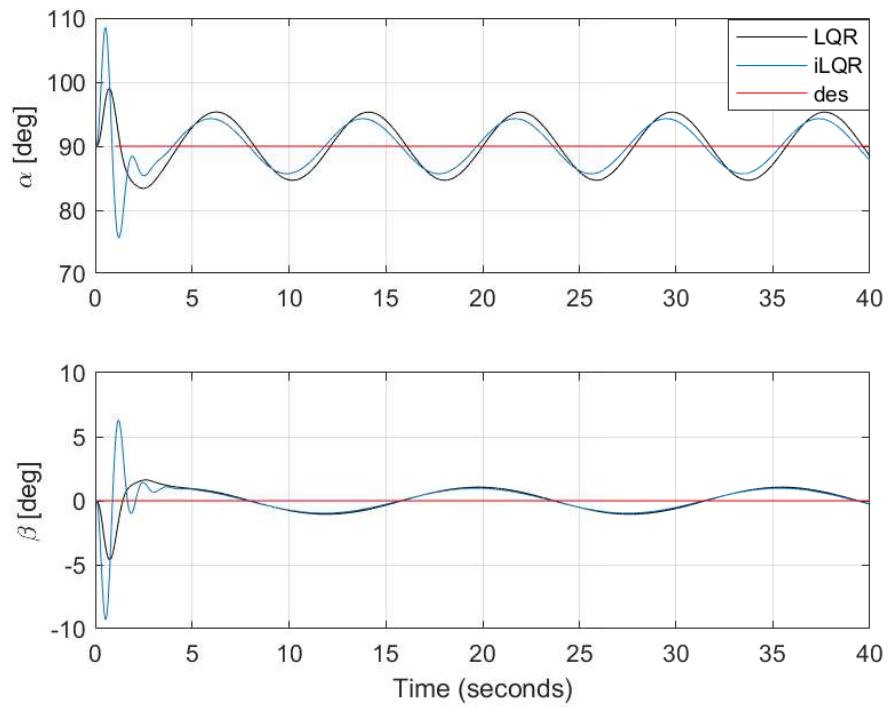
FIGURE 4.14: Payload angles using LQR and ILQR controllers
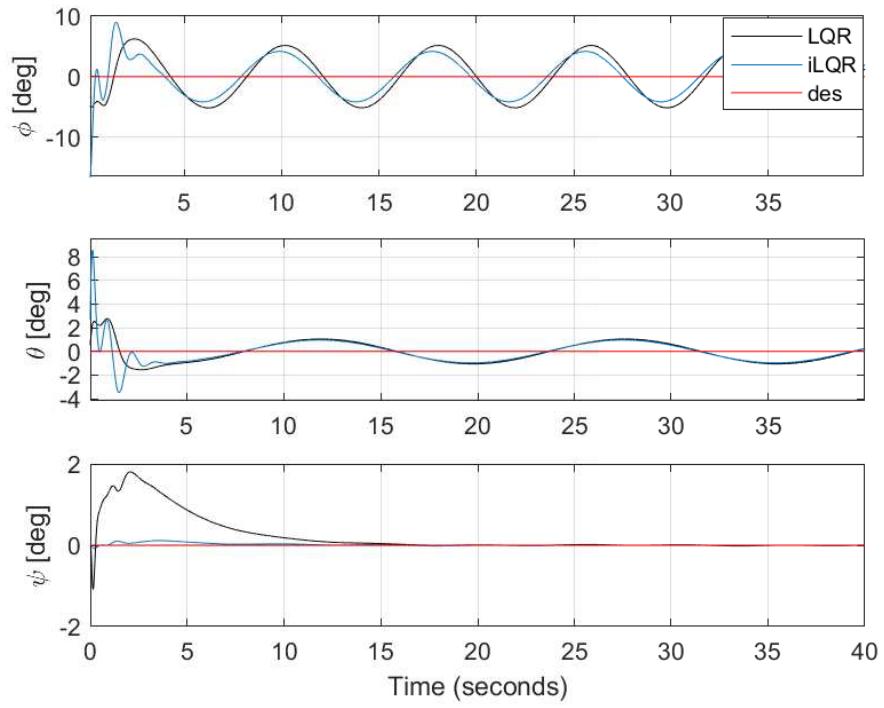
FIGURE 4.15: Quadrotor angles using the LQR and ILQR controllers
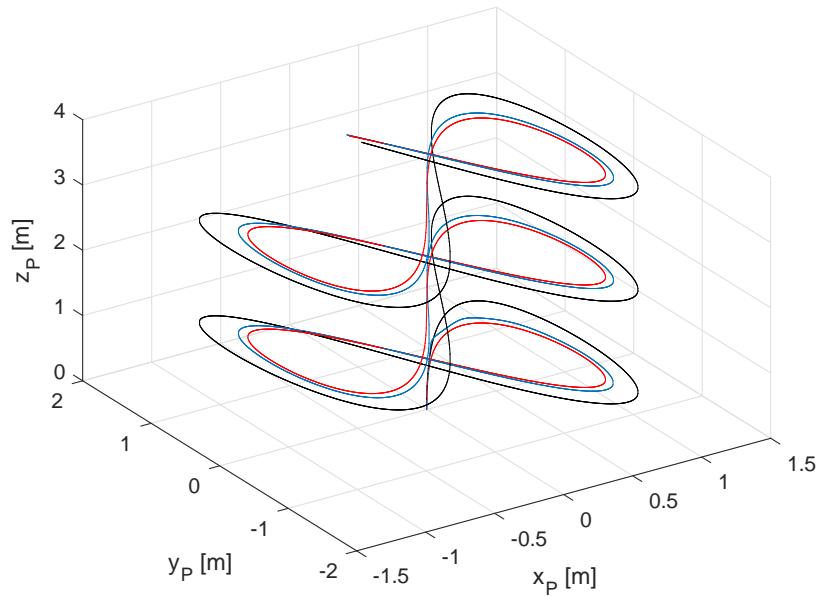
FIGURE 4.16: Load position in three dimensions using the LQR and ILQR controllers

RMSE	Eight			Spiral		
	$x_P(m)$	$y_P(m)$	$z_P(m)$	$x_P(m)$	$y_P(m)$	$z_P(m)$
ILQR	0.0026	0.0049	0.005	0.013	0.021	0.004
LQR	0.0126	0.057	0.059	0.083	0.054	0.046

TABLE 4.3: Payload position RMSE values for the two trajectories under ILQR and LQR Controllers

Similarly, the second simulation result is presented by Figure 4.17 for load trajectory, where the system stability is achieved in the first trajectory test. Both trajectory results show the improvement in performance with small steady-state errors. The same conclusion can be drawn from the result, that is, the ILQR controller outperforms the LQR controller.
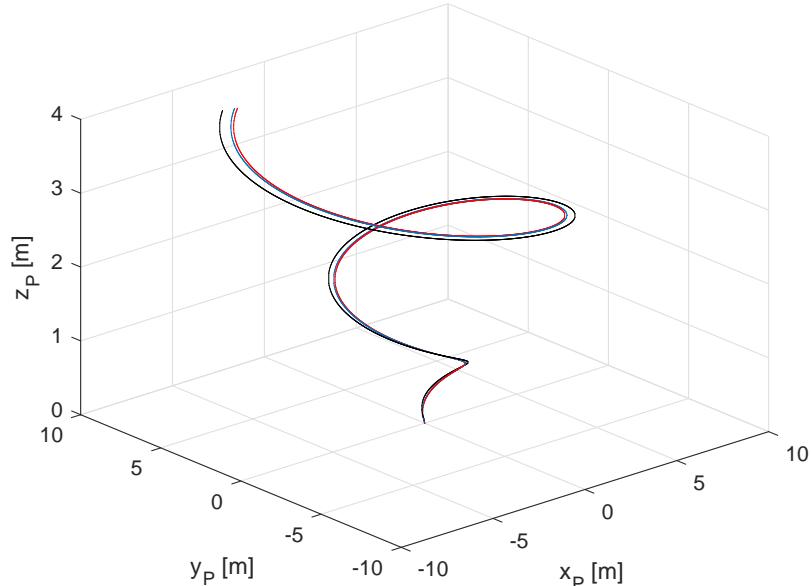
FIGURE 4.17: Quadrotor trajectory using the ILQR controller

The load position RMSE values for the two paths are demonstrated in Table 4.3

4.6 ILQR Simulation Results for Two Quadrotors

In this section, the ILQR controller is tested with two quadrotors for the transportation task to track the desired eight shape trajectory using the MATLAB simulator. The equilibrium point for the system is $\alpha^1 = 90$, $\alpha^2 = 90$, $\beta^1 = 45$ and $\beta^2 = -45$. The desired eight shape trajectory is presented by $x^* = sin(2\pi/t)$, $y^* = sin(2\pi/t)$, $z^* = 0.1 * t$. The ILQR simulation results are compared with those of the LQR controller, where the sampling time is 0.02 sec and the running time is 40 sec. Furthermore, the weight matrices Q, R and Q_f for the system are chosen based on the generalised coordinates sequence $\boldsymbol{q} = [x_P, \dot{x}_P, y_P, \dot{y}_P, z_P, \dot{z}_P,$ $\alpha^1, \dot{\alpha}^1, \beta^1, \dot{\beta}^1, \phi^1, \dot{\phi}^1, \theta^1, \dot{\theta}^1, \psi^1, \dot{\psi}^1 \ \alpha^2, \dot{\alpha}^2, \beta^2, \dot{\beta}^2, \phi^2, \dot{\phi}^2, \theta^2, \dot{\theta}^2, \ \psi^2, \dot{\psi}^2]^T$ and control weight matrix based on the control sequence $\boldsymbol{u} = [F_z^1, M_x^1, M_y^1, M_z^1, F_z^2, M_x^2, M_y^2, M_z^2]^T$ in the form of diagonal matrices as

$$Q = diag([1000, 100, 1000, 100, 1000, 100, 100, 0, 100, 0, 1000,$$
$$0, 1000, 0, 1000, 0, 100, 0, 100, 0, 1000, 0, 1000, 0, 1000, 0])$$

$$Q_f = diag([1000, 10, 1000, 10, 1000, 10, 100, 0, 100, 0, 1000,$$
$$0, 1000, 0, 1000, 10, 100, 0, 100, 0, 1000, 0, 1000, 0, 10, 0])$$

$$R = diag([0.0001, 0.001, 0.0001, 0.001, 0.0001, 0.001, 0.0001, 0.001]).$$

The performance of transporting the cable suspended payload with two quadrotors is illustrated in Figures 4.18-4.23. The payload position in Figure 4.18 shows a more stable performance and converges more quickly to the desired trajectory using the ILQR controller than the LQR controller. The first and second quadrotor attitudes are shown in Figures 4.19 and 4.20. Although all the Euler angles are

RMSE	$x_P(m)$	$y_P(m)$	$z_P(m)$
ILQR	0.0027	0.0041	0.0055
LQR	0.035	0.040	0.017

TABLE 4.4: Payload RMSE values for spiral trajectory under ILQR and LQR Controllers

not very close to the desired ones, the yaw angles from both quadrotors are more stable when using the ILQR controller than the LQR controller. The payload angles with respect to the first and second quadrotors are shown in Figures 4.21 and 4.22, respectively. Again, these angles, which are controlled by the ILQR controller, outperform the ones controlled by the ILQR controller as they have fewer oscillations.

The 3D animation trajectory using the ILQR controller is shown in Figure 4.23. It clearly shows that the payload is able to track the desired red trajectory. To compare the ILQR controller with the LQR one, the 3D trajectories are projected onto a 2D space in Figures 4.24 and 4.25. It is obvious that the trajectory controlled by the ILQR controller is better than that controlled by the LQR controller.

The load position RMSE values for the spiral path using two quadrotors was demonstrated in Table 4.4.

In summary, the ILQR controller performance is more stable and faster and produces smaller steady-state errors than the LQR controller. However, it requires more computational time due to the iteration.
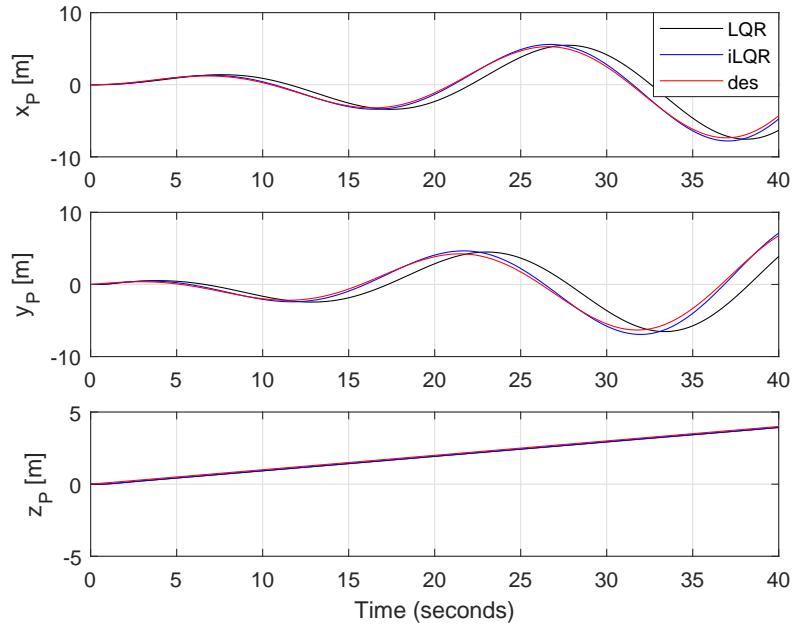
FIGURE 4.18: Payload position using the LQR and ILQR controllers
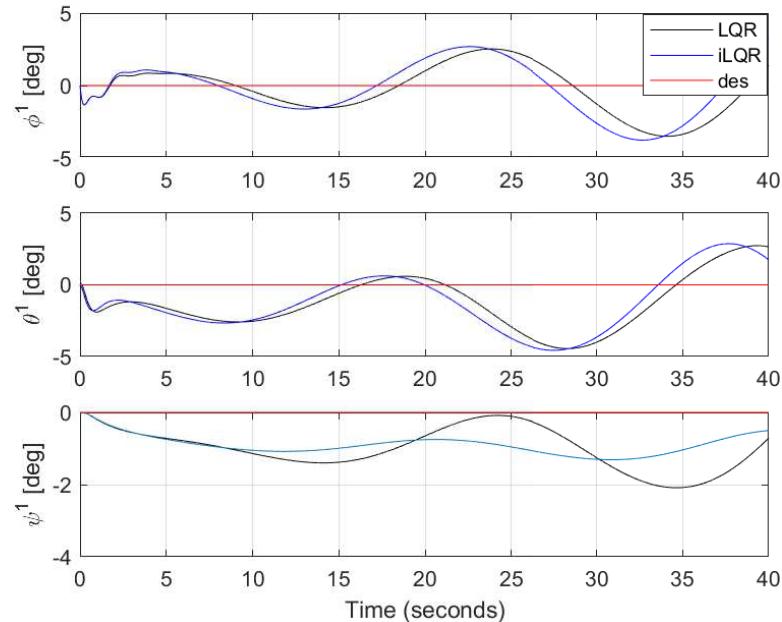
FIGURE 4.19: The first quadrotor angles using the IQR and ILQR controllers
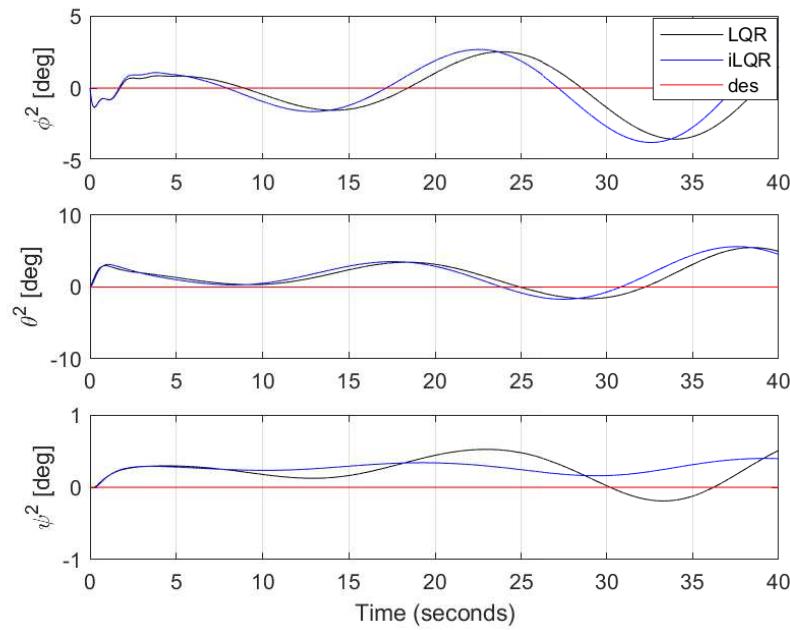
FIGURE 4.20: The second quadrotor angles using the LQR and ILQR Controllers
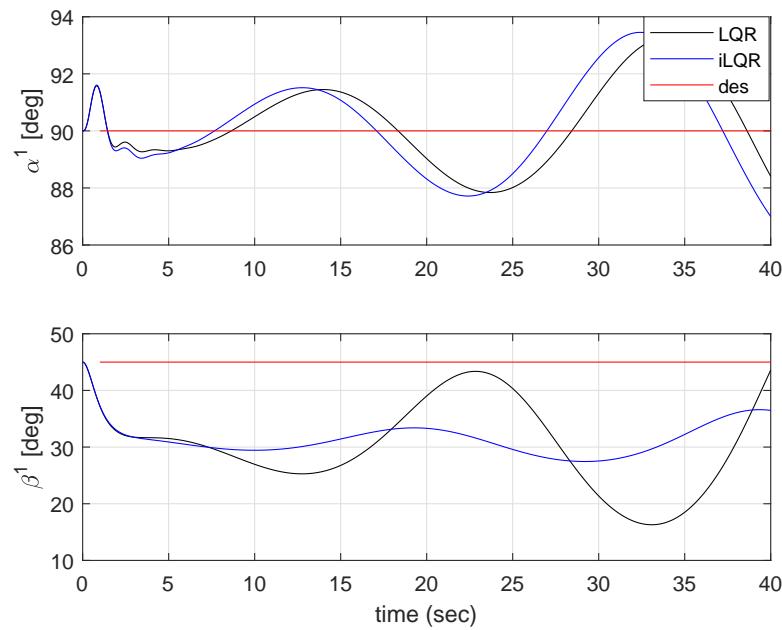
FIGURE 4.21: The angels of the first quadrotor-load rope using the LQR and ILQR controllers
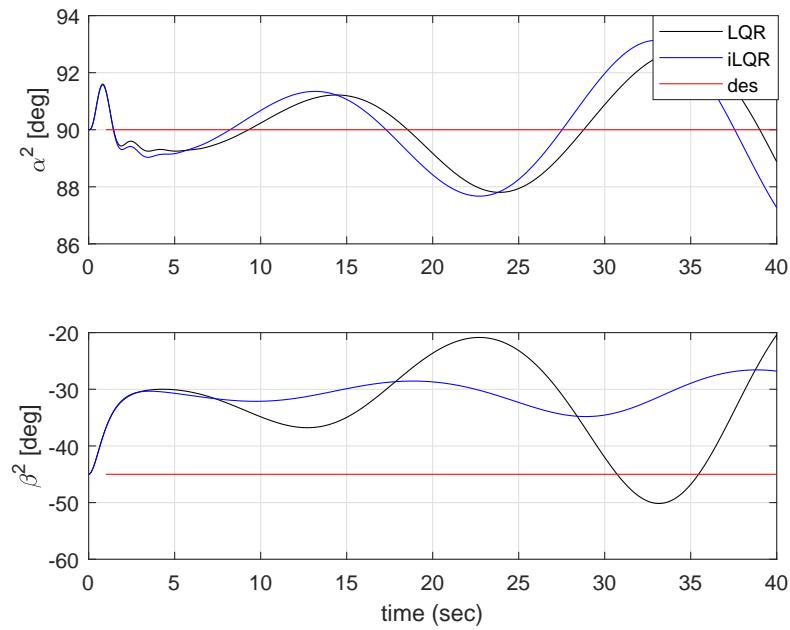
FIGURE 4.22: The angels of the second quadrotor-load rope using the LQR and ILQR Controllers
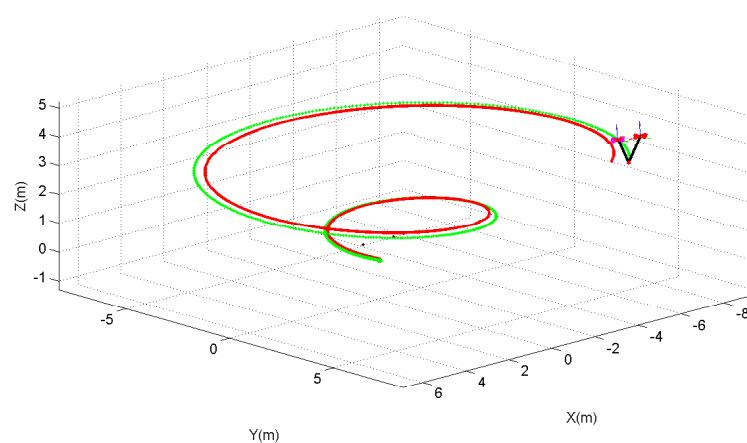
FIGURE 4.23: 3D load position using the ILQR Controller
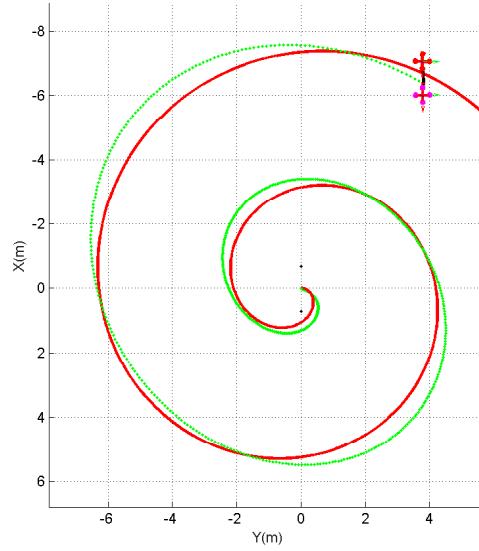
FIGURE 4.24: 2D load position using the LQR Controller
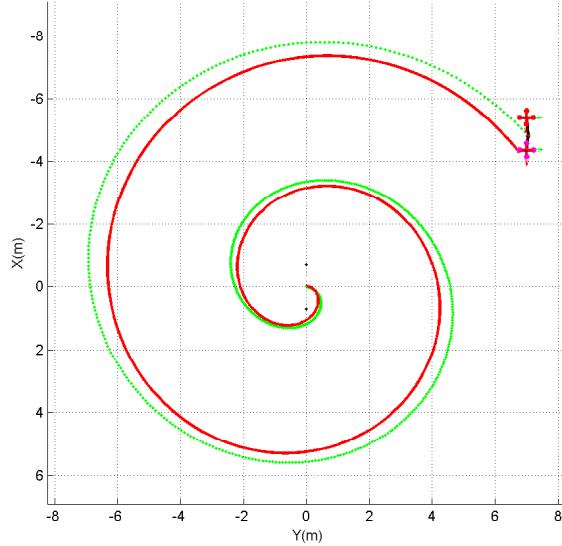
FIGURE 4.25: 2D load position using the ILQR Controller

4.7   Summary

This chapter presented the LQR and ILQR controllers in an attempt to stabilise the quadrotor with a cable-suspended load in transporting tasks. The simulation

results show that the LQR controller efficiently minimises the steady state error and time consumption to reach stability. However, in order to handle the high non-linearity, the ILQR controller is applied to the system and the results are verified.

The results show that the ILQR controller is stable and outperforms the LQR controller. This indicates that the iteration of the ILQR controller is able to compensate for the payload impact on the underlying system dynamics and the changes in the operating point induced by the slung load. The next chapter will consider various constraints imposed on the system, and MPC controllers will be developed.

Chapter 5

Constrained Model Predictive Controllers (MPC)

5.1 Introduction

In the previous chapter, it was shown that the classical PD controller produces a weak performance in terms of accuracy and system stability achievement when compared with the LQR controller. It was also revealed that the ILQR controller, when applied to the linearised model, achieves a better performance than that of the LQR controller due to the former's capability of handling high non-linearity via multiple iterations.

However, the main drawback of the LQR and ILQR controllers is their incapability of handling physical limitations of the system states and the control inputs. It is well known that handling the required state and control constraints and achieving the system stability are the main priority needs for tracking a trajectory by the high non-linear dynamic models. Therefore, we propose and develop a constrained Model Predictive Control (MPC). In this chapter, this suggested controller is the most common controller used to overcome the drawback.

The MPC controller has emerged as the most popular and valuable control technique since the 1980s, known as Receding Horizon controller as well. It is a worth

describing the method based on the principle of prediction to a finite horizon at each sampling time starting from the current state. This idea is called receding horizon. Subjected to the state and control constraints of the system dynamics, the optimisation problem is solved to determine the optimal control input. Then, the first optimised control input is applied to the system [110], [111].

In this chapter, two controllers, linear model predictive control LMPC and non-linear model predictive control NMPC, are designed to improve the performance with respect to the constraints. Firstly, an LMPC controller is used for the transportation task with two quadrotors. It is compared with the LQR controller in terms of stability. Secondly, an NMPC controller is designed with the consideration of handling high non-linearity in the dynamic model within the constraints. The constraints to be considered include input saturation, swing angle limitation, and payload position constraint.

5.2 Constrained Linear Model Predictive Control

In this chapter, an optimal tracking controller is considered for the suspended payload with two quadrotors by two cables. The linear model predictive control LMPC approach is designed by relying on the linearisation of the dynamic model. Then suitable weight matrices and horizons parameters are selected. The discrete-time dynamical model description for the system with two quadrotors carrying a cable-suspended payload is shown in equation (5.1).

$$x_{k+1} = A_k x_k + B_k u_k \tag{5.1}$$

where, $A_k \in \mathbb{R}^{26 \times 26}$, $B_k \in \mathbb{R}^{26 \times 4}$.

The finite horizon optimal controller solves the following constrained optimisation problem at each time instant k to implement the MPC algorithm. The cost

function is presented by:

$$J = (\boldsymbol{x_N} - \boldsymbol{x_N^*})^T \boldsymbol{Q_f}(\boldsymbol{x_N} - \boldsymbol{x_N^*})$$
$$+ \sum_{t=k}^{N+k-1} [(\boldsymbol{x_t} - \boldsymbol{x_t^*})^T \boldsymbol{Q}(\boldsymbol{x_t} - \boldsymbol{x_t^*}) + \boldsymbol{u_t^T} \boldsymbol{R} \boldsymbol{u_t}] \tag{5.2}$$

where the terminal state and its desired state are denoted by $\boldsymbol{x_N}$ and $\boldsymbol{x_N^*}$, respectively, and the reference state is denoted by $\boldsymbol{x_k^*}$. The prediction horizon is denoted by $\boldsymbol{N}$. $\boldsymbol{Q_f}$ and $\boldsymbol{Q}$ are positive semidefinite matrices and $\boldsymbol{R}$ is a positive definite matrix.

The constrained optimisation problem for linear MPC at each time instant k is

$$\min_{\boldsymbol{u_k}} J$$

Subject to

$$\boldsymbol{x_{k+1}} = \boldsymbol{A_k x_k} + \boldsymbol{B_k u_k}$$

$$\boldsymbol{x_k} \in \mathbb{X}, \text{the state constraints}$$

$$\boldsymbol{u_k} \in \mathbb{U}, \text{the control constraints}$$

where the state and input constraints are presented as

$$\mathbb{X} = \{\boldsymbol{x_{kmin}} \leq \boldsymbol{x_k} \leq \boldsymbol{x_{kmax}}\}$$

$$\mathbb{U} = \{\boldsymbol{u_{kmin}} \leq \boldsymbol{u_k} \leq \boldsymbol{u_{kmax}}\}$$

The linear optimisation algorithm is implemented to obtain a minimum finite horizon cost of the desired trajectory states and control inputs while the constraints are satisfied. The optimisation problem is solved for the LMPC controller. The YALMIP solver is a high-level numerical solution of the optimisation problem, which is focused on effective modelling with high-level algorithms in [112] and [113].

The main principle of the YALMIP solver relies on an external low-level numerical solution solver. A quadratic programming (QP) algorithm is used to solve the low-level optimisation problem at each time instant $\boldsymbol{k}$, which can produce an input sequence $\{\boldsymbol{u_{k|k}}, \boldsymbol{u_{k+1|k}}, \boldsymbol{u_{k+N-1|k}}\}$. Then the true input at $\boldsymbol{k}$ is $\boldsymbol{u_k} = \boldsymbol{u_{k|k}}$. This solver is applied to solve the tracking optimisation problem of the high dimensional model presented by the suspended payload with two quadrotors by cables.

5.3 Constrained Non-linear Model Predictive Control

A Non-linear Model Predictive Control NMPC approach is proposed in this chapter to handle the high non-linearity based on prediction method at each sampling time subjected to the state and control constraints. For each sampling $\boldsymbol{k}$, the optimal control sequence can be found by solving the optimisation problem including constraints in order to perform the NMPC method. This controller is implemented using high-level sophisticated algorithms. An Advanced Process Monitor (APM) is a high-level mathematical optimization software and coupled with a nonlinear programming or a quadratic programming (QP) algorithm can be used to solve the above constrained optimisation problem at each time instant $\boldsymbol{k}$.

The constrained optimisation problem for non-linear MPC at each time instant k is

$$\min_{\boldsymbol{u_k}} J$$

subject to

$$\boldsymbol{x_{k+1}} = f(\boldsymbol{x_k}, \boldsymbol{u_k})$$

$$\boldsymbol{x_k} \in \mathbb{X}, \text{the state constraints}$$

$$\boldsymbol{u_k} \in \mathbb{U}, \text{the control constraints .}$$

5.4 MPC Simulation

In this section, a linear and a non-linear MPC controllers are implemented to achieve the system optimal performance. In the first part, the proposed LMPC control performance is tested and compared with that of the unconstrained LQR controller in terms of load position path tracking and attitude stabilisation utilising a numerical MATLAB simulator.

In the second part, an NMPC controller is performed and compared with that of the LMPC controller to test the system performance. To show how nonlinearity is handled in the NMPC controller, the changes of the model parameters are considered and same external disturbances are applied. The design parameters based on the suspended load with two quadrotors by a cable used in the simulator are listed in Table 4.1.

5.4.1 LMPC Simulation Results

The simulations are conducted in MATLAB with the state and control input constraints. These restrictions are considered so that they can minimise errors in terms of payload position, swing angles and quadrotor's orientations while trajectory tracking. Two spiral and one eight shape trajectories are introduced in order to provide a strong proof of improving the performance of the system.

The operational point for the suspended load with two quadrotors by cables for four angles of two ropes are $\alpha^1 = 90^o, \beta^1 = 45^o, \alpha^2 = 90^o, \beta^2 = -45^o$. The desired payload spiral trajectory is defined by $x^* = cos(2\pi/t), y^* = sin(2\pi/t), z^* = 0.1t$, where t is started from 0 with a sampling time of $0.02s$, and the tracking time is up to $30s$. The MPC prediction horizon is selected as 900 steps and the control horizon as 120 steps. The weight matrices of the cost function are selected based on the generalised coordinates sequence $\boldsymbol{q} = [x_P, \dot{x}_P, y_P, \dot{y}_P, z_P, \dot{z}_P,$

$\alpha^1, \dot{\alpha}^1, \beta^1, \dot{\beta}^1, \phi^1, \dot{\phi}^1, \theta^1, \dot{\theta}^1, \psi^1, \dot{\psi}^1 \ \alpha^2, \dot{\alpha}^2, \beta^2, \dot{\beta}^2, \phi^2, \dot{\phi}^2, \theta^2, \dot{\theta}^2, \ \psi^2, \dot{\psi}^2\Big]^T$ for two quadrotors with suspended load by cables and the control sequence $\boldsymbol{u} = [F_z^1, M_x^1, M_y^1, M_z^1, F_z^2, M_x^2, M_y^2, M_z^2]^T$ in the form of diagonal matrices as

$$Q = Q_f = diag([100, 1, 100, 1, 100, 1, 1, 1, 1, 1, 1, 1, 1, 1,$$
$$1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1])$$

$$R = diag([0.01, 0.1, 0.01, 0.1, 0.01, 0.1,$$
$$0.01, 0.1]).$$

With regard to the constraints on control inputs, the maximum saturation limit is considered to reflect the limitation of the practical rotor's power. The constraints are presented by the maximum angular velocity for all the rotors in both quadrotors $\boldsymbol{F}_i^k \leq 700 rad/s$.

With regard to the constraints on state vectors, the maximum and minimum limits are considered to reflect the limitation of the environment. This includes the payload position limit in x direction, and the payload swing angle limit to avoid the possible collisions. The desired load trajectory is shown in Figure 5.1 and represented by the red trajectory starting from $(1, 0, 0)$. The constraints includes $x_k \leq -0.5$ and $88° \leq \alpha^1 = \alpha^2 \leq 92°$, $40° \leq \beta^1 = \beta^2 \leq 50°$.

The desired trajectory in Figure 5.1 is followed by the actual blue path. It starts from $(0, 0, 0)$, then moves towards the desired initial point $(1, 0, 0)$ and closely tracks the desired spiral path. It can be clearly seen that the payload position points along the followed trajectory show accurate and stable behaviour while the state and control constraints are applied.
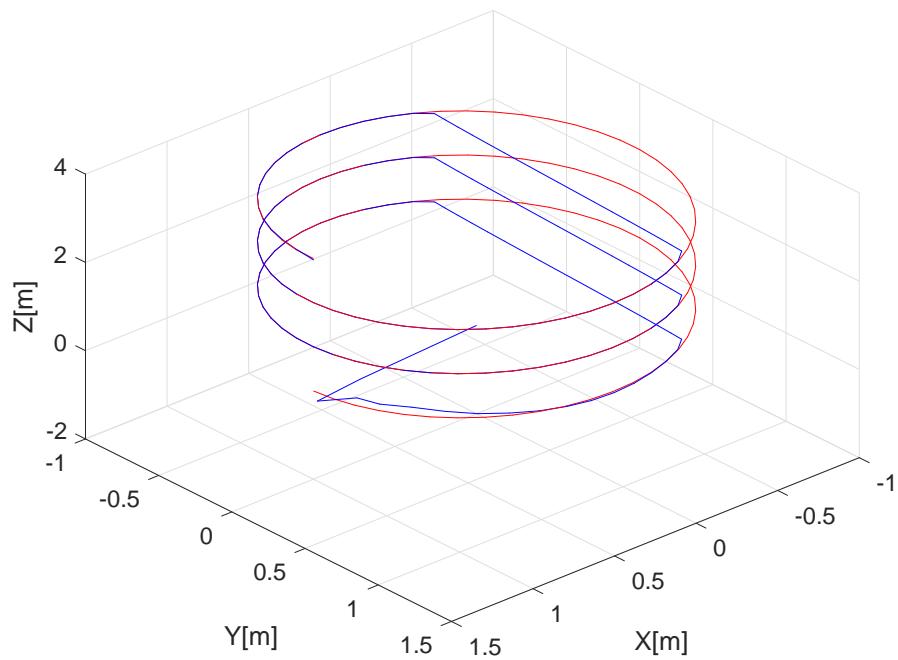
FIGURE 5.1: 3D load position using the LMPC controller

The LQR trajectory is shown in Figure 5.2 represented by the blue trajectory. It starts from $(0, 0, 0)$, then moves towards the desired one, but fails to move close to it due to the limits applied to angular velocities. Thus, the actual blue trajectory shows a steady state error and weak stability.
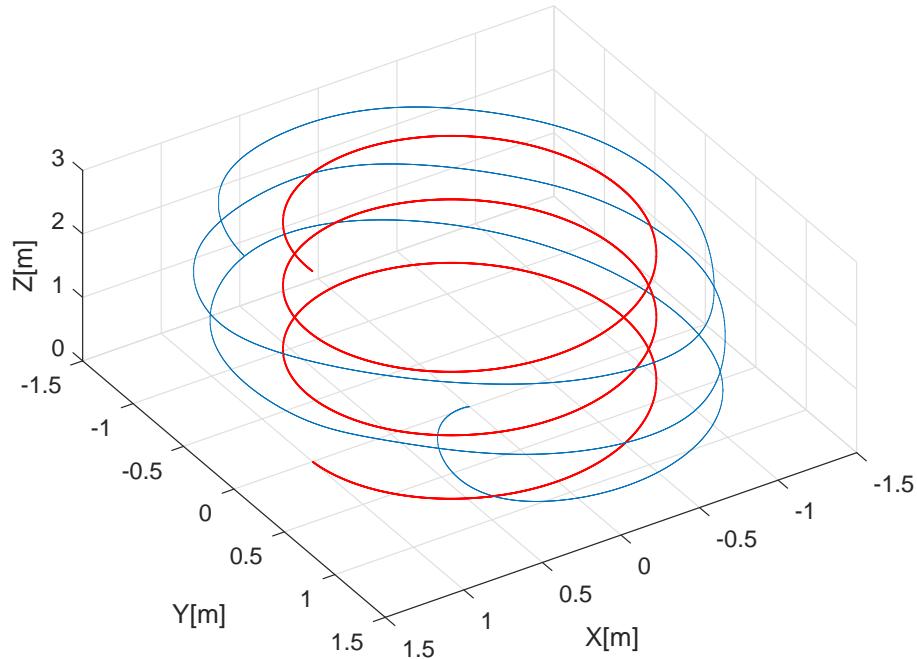
FIGURE 5.2: 3D load position using the LQR controller with the control limitation

Four angular velocities of the first quadrotor using the LMPC controller are shown in Figure 5.3. It is clearly indicated that all of them are capped at $700 rad/s$, i.e. the constraints on control inputs are utilised to keep against the power consumed by the motors. Four angular velocities of the second quadrotor have a similar performance and are ignored here.
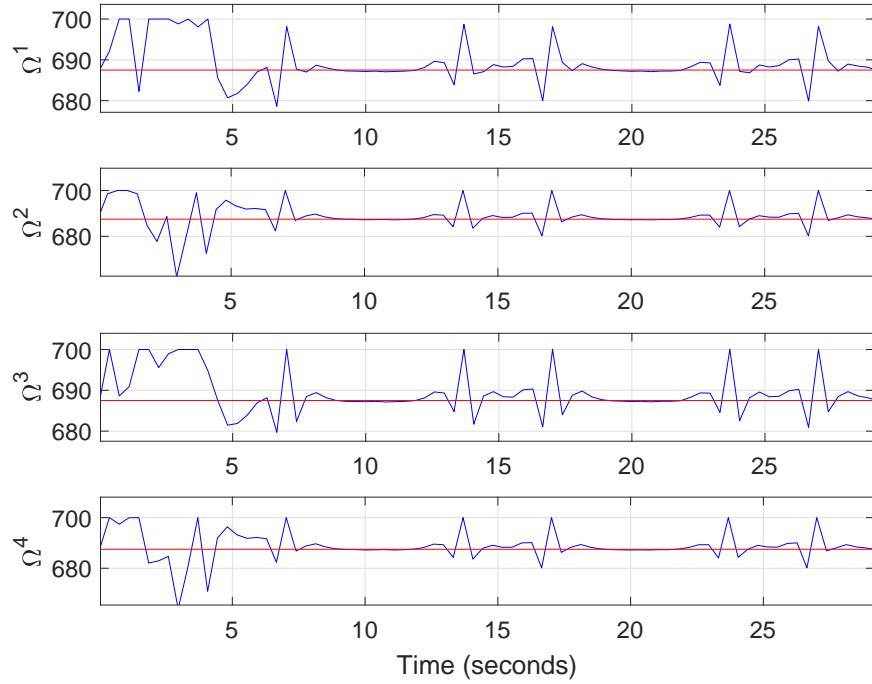
FIGURE 5.3: Angular velocities of the first quadrotor using the LMPC controller

The rope angles with the first quadrotor using the LMPC controller and the LQR controller are shown in Figures 5.4 and 5.5, respectively. It can be seen that the constraints on the angles are utilised by the LMPC controller, but not by the LQR controller. In terms of the swing angles constraints, the limitation range of the α^1 was two degrees for both minimum and maximum values. The reference value of this angle is equal to 90°, thus the upper and lower limits are presented from 92° to 88° within four degrees. Regarding the LQR controller, both swing angles are unable to be constrained.
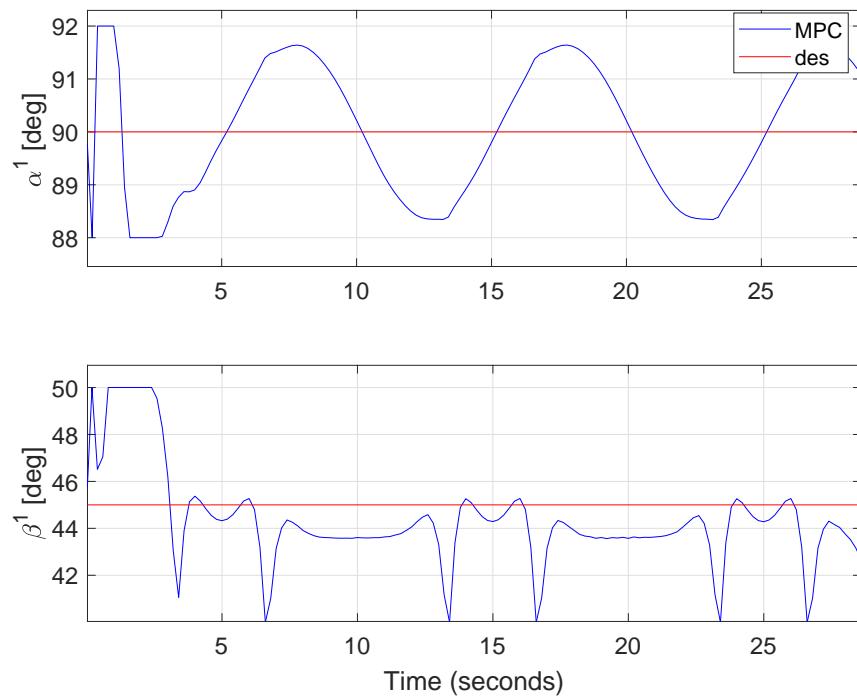
FIGURE 5.4: Rope angles with the first quadrotor using the MPC controller
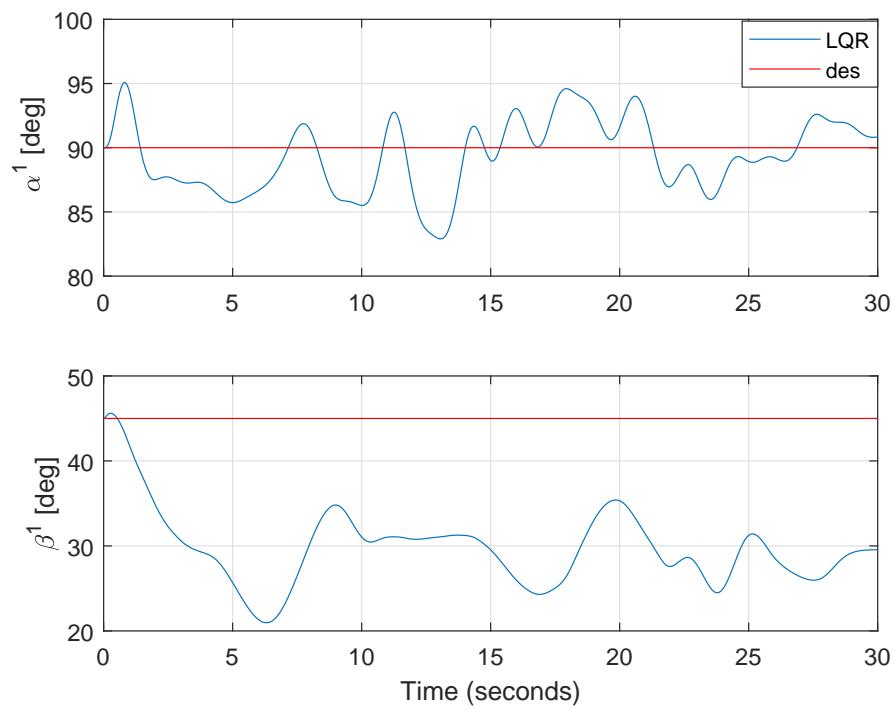
FIGURE 5.5: Rope angles with the first quadrotor using the LQR controller

The comparisons between the LMPC and LQR controllers on the payload position and the Euler angles of the first quadrotor are shown in Figures 5.6, 5.7, 5.8, and 5.9. Regarding the payload position, Figures 5.6 and 5.7 show that the LMPC controller performs better than the LQR controller in terms of handling the payload position constraints, reducing the steady state errors and stabilising the system states. Figures 5.8 and 5.9 show that the LMPC controller has a smoother pose compared with that of the LQR controller, which proves again the ability of the LMPC controller to process the system accuracy and stability behaviour while the constraints are applied.
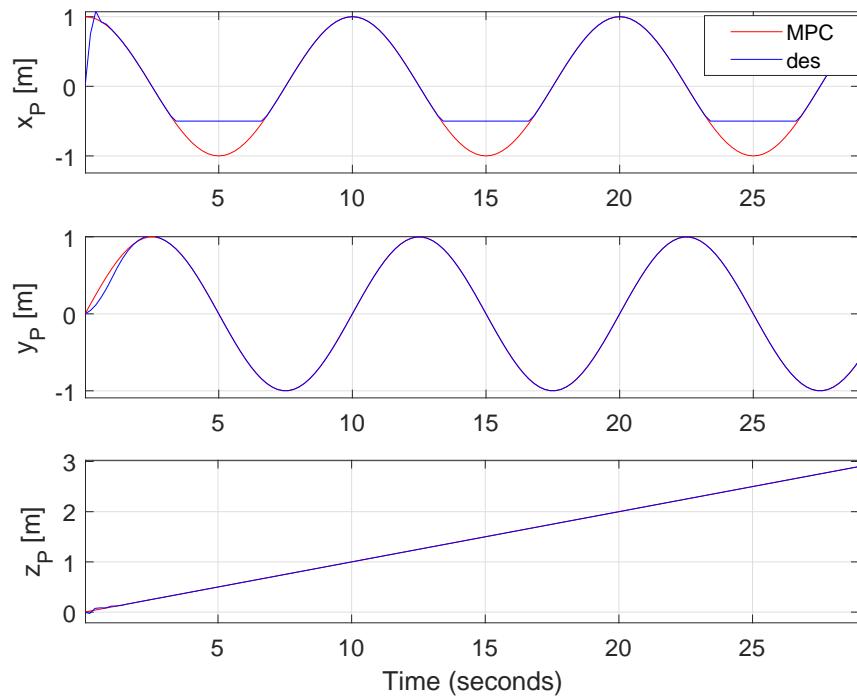
FIGURE 5.6: Payload position using the LMPC controller
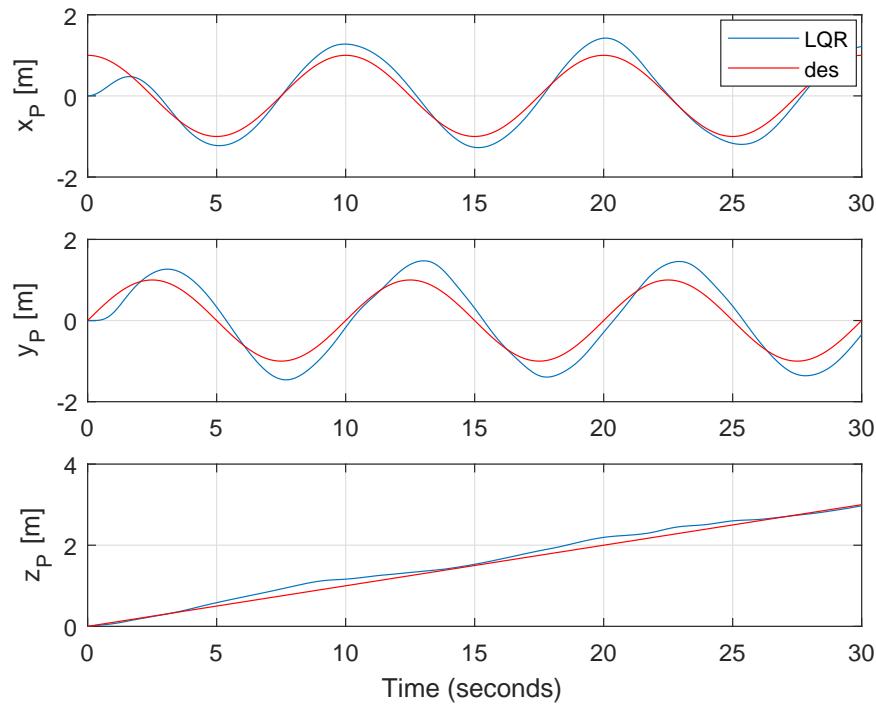
FIGURE 5.7: Payload position using the LQR controller
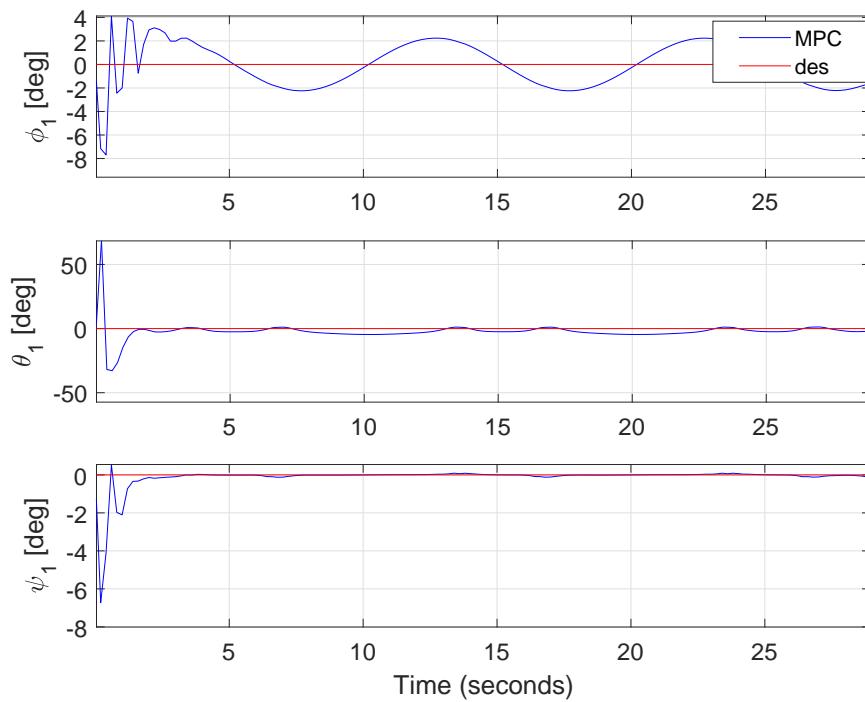
FIGURE 5.8: Eular angles of the first quadrotor using the LMPC controller
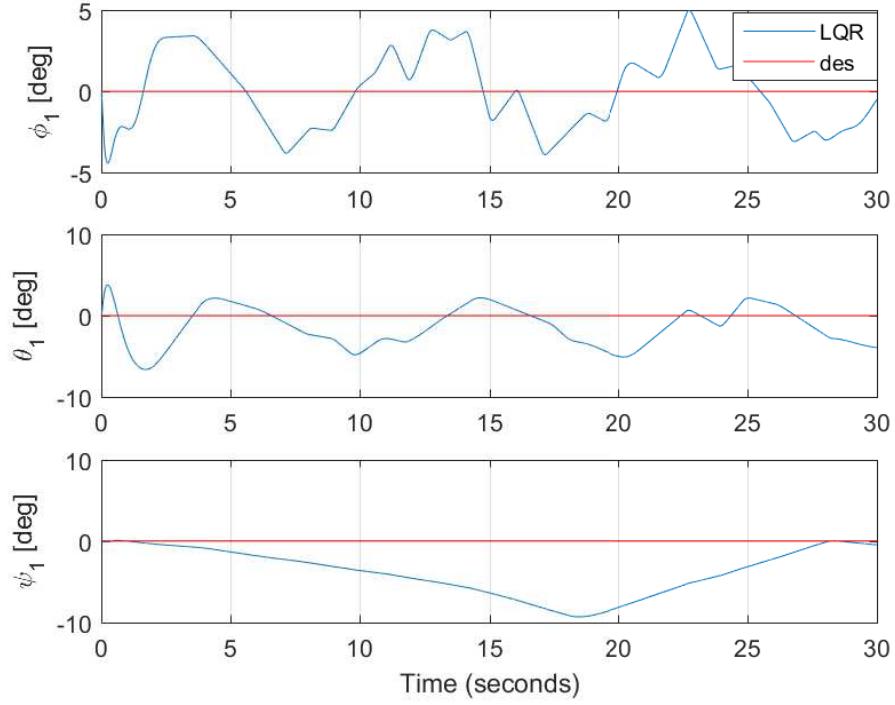
FIGURE 5.9: Eular angles of the first quadrotor using the LQR controller

The desired eight-shape trajectory is shown in red in Figures 5.10 and 5.11, where the desired initial position is $(0,0,0)$. It can be seen that the LQR tracking trajectory tried to move closer to the desired one in Figure 5.11, but failed due to the constraint imposed on the thrust. Furthermore, we cannot impose the constraints on the state variables.

Tracking the eight shape trajectory utilising the LMPC controller is shown in Figure 5.10, where the input thrust and state constraints, including $x_k \leq -0.4$ and $-1.0 \leq y_k \leq 1.0$, are imposed. It has a very slight overshoot from the desired trajectory at the initial position. Then the tracking performance works very well. The x direction constraint is clearly observed and considered. In general, a much better performance has been demonstrated by the LMPC controller when compared with the LQR controller.
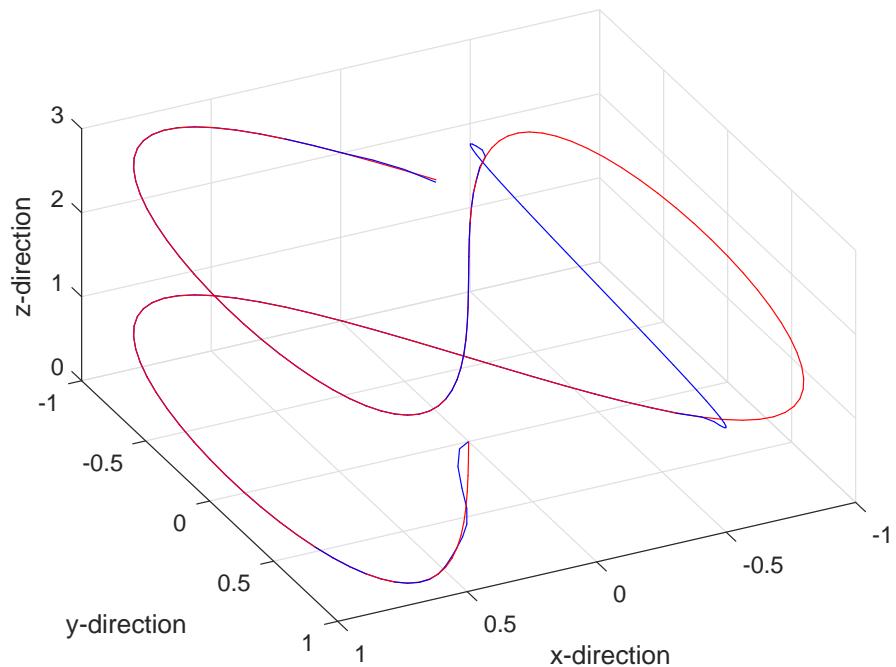
FIGURE 5.10: 3D load position using the LMPC controller
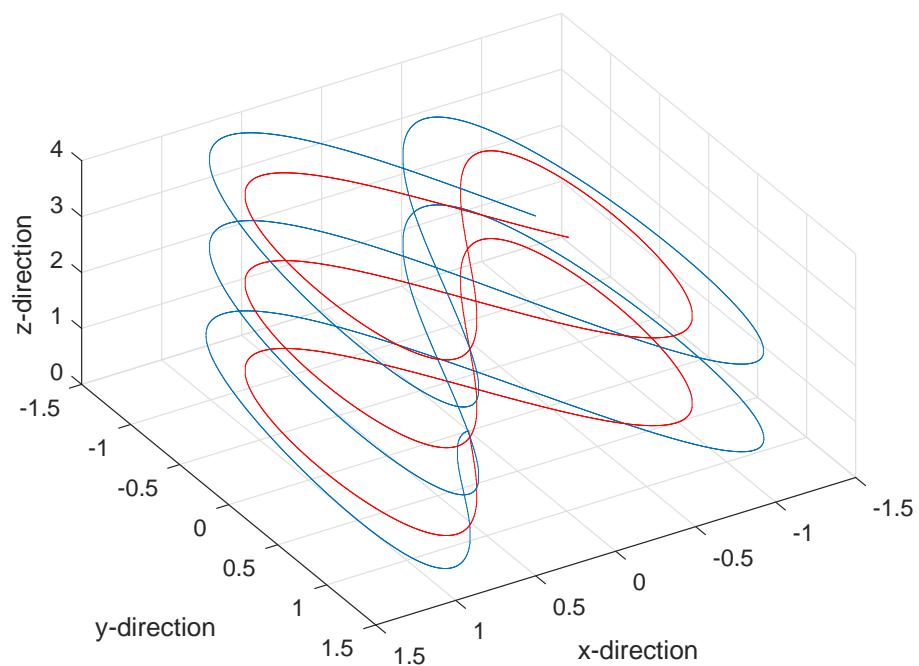
FIGURE 5.11: 3D load position using the LQR controller

	Eight			Spiral		
RMSE	$x_P(m)$	$y_P(m)$	$z_P(m)$	$x_P(m)$	$y_P(m)$	$z_P(m)$
LMPC	0.0032	0.0025	0.0012	0.0061	0.0043	0.0037
LQR	0.083	0.055	0.009	0.067	0.044	0.048

TABLE 5.1: Payload position RMSE values for the two trajectories under LMPC and LQR Controllers

The load position RMSE values for the two paths are demonstrated in Table 5.1

5.4.2   NMPC Simulation Results

The tracking task of the suspended load with two quadrotors is performed in MATLAB simulation. In this task, the system performance is tested in terms of handling constraints, accuracy, stability and robustness during the reasonable time. This test is implemented to compare between the LMPC controller and the NMPC controller by demonstrating the effect of system constraints applied to control inputs and state vectors and with input disturbance.

The reference spiral path is considered and defined by $x^* = 0.2t\cos(0.3t), y^* = 0.2t\sin(0.3t), z^* = 0.1t$, where the tracking simulation time t is up to 30s. The prediction horizon of both LMPC and NMPC controllers is chosen as 10, and the cost function weight matrices are selected as

$$Q = Q_f = diag([100, 1, 100, 1, 100, 1, 1, 1, 1, 1, 1, 1, 1, 1,$$
$$1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1])$$

$$R = diag([0.01, 0.1, 0.01, 0.1, 0.01, 0.1,$$
$$0.01, 0.1]).$$

With regard to the constraints applied on the system states, the maximum and minimum saturation limits are performed by the motors. This also includes the

payload position upper and lower limits on x and y directions, and the swing angle limits of the payload $\alpha^1, \beta^1, \alpha^2$ and β^2.

The desired payload trajectories of both NMPC and LMPC controllers are the same and are shown in Figures 5.12 and 5.13 in red. They start away from the original point $(0, 0, 0)$. Then, a disturbance is added to the system thrust 1.5 to test the system robustness from time $20s$.

The payload position constraints include $-1.0 \leq x_k \leq 1.0$, $-1.4 \leq y_k \leq 1.7$, and the angle constraints are $70° \leq \alpha^1, \alpha^2 \leq 100°$, $20° \leq \beta^1 \leq 70°$ and $-70° \leq \beta^2 \leq -20°$. In addition, the two quadrotors' attitudes are constrained by $-20° \leq \phi^1 \leq 20°$, $-20° \leq \theta^1 \leq 20°$, $-20° \leq \psi^1 \leq 20°$, $-20° \leq \phi^2 \leq 20°$, $-20° \leq \theta^2 \leq 20°$ and $-20° \leq \psi^2 \leq 20°$. Both LMPC and NMPC controllers are subjected to the same situation with regards to the constraints, disturbances, starting points, parameters and desired reference trajectory.

The blue trajectory is produced by the NMPC controller and shown in Figure 5.12, while the green one is made by the LMPC controller and displayed in Figure 5.13. They are both started from the selected initial point $(-1.0, 1.5, 0)$. While the NMPC controller moves closer to the desired starting point $(0, 0, 0)$ and begins to track the desired red path, the LMPC controller fails not only to converge accurately to the desired red trajectory but also to lose tracking control due to the disturbance applied to its model.
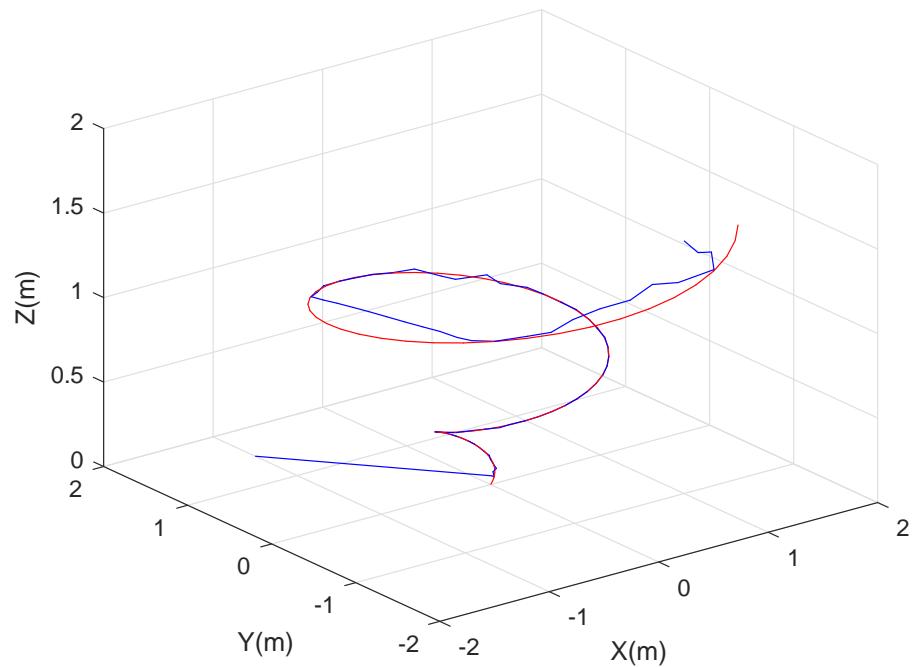
FIGURE 5.12: Load position using the NMPC controller
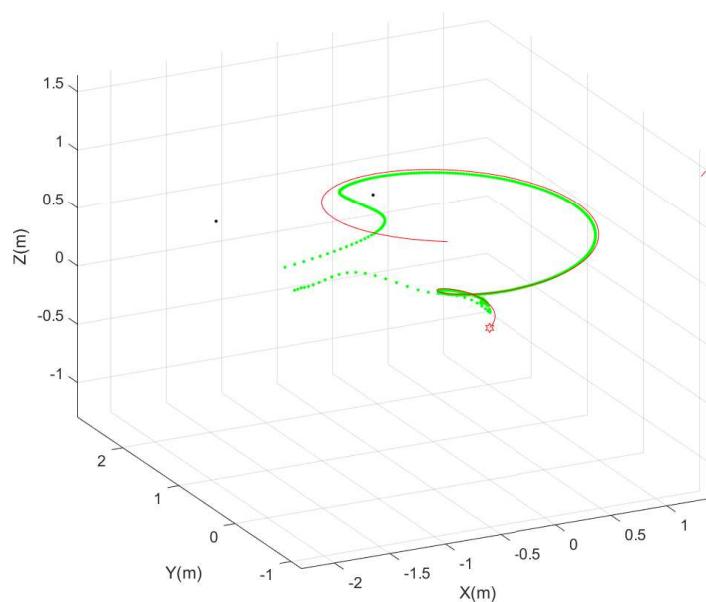
FIGURE 5.13: Load position using the LMPC controller

Two rope angles utilising the NMPC and LMPC controllers are shown in Figures 5.14 and 5.15 for the first quadrotor and in Figures 5.16 and 5.17 for the second quadrotor. It is clearly indicated that the constraints on the angles are considered by both LMPC and NMPC controllers. It can be seen that the rope angles produced by the NMPC controller have less and smaller fluctuations than those of the LMPC controller. This result indicates that the non-linearity is handled well by the NMPC controller. The large changes from $t = 20s$ are caused by the disturbance applied to the system. Again, the NMPC controller recovers better than the LMPC controller.
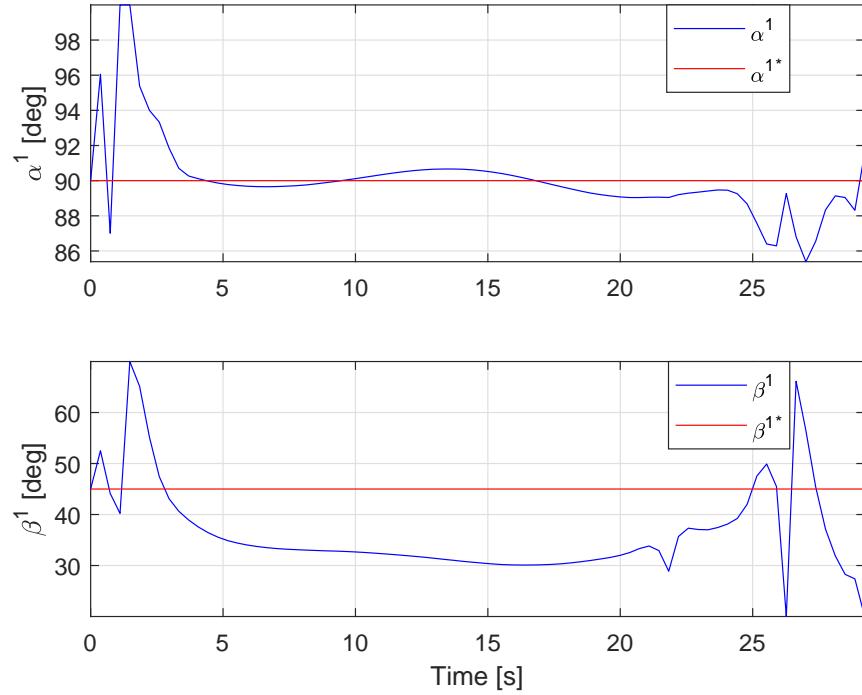
FIGURE 5.14: Rope angles of the first quadrotor using the NMPC controller
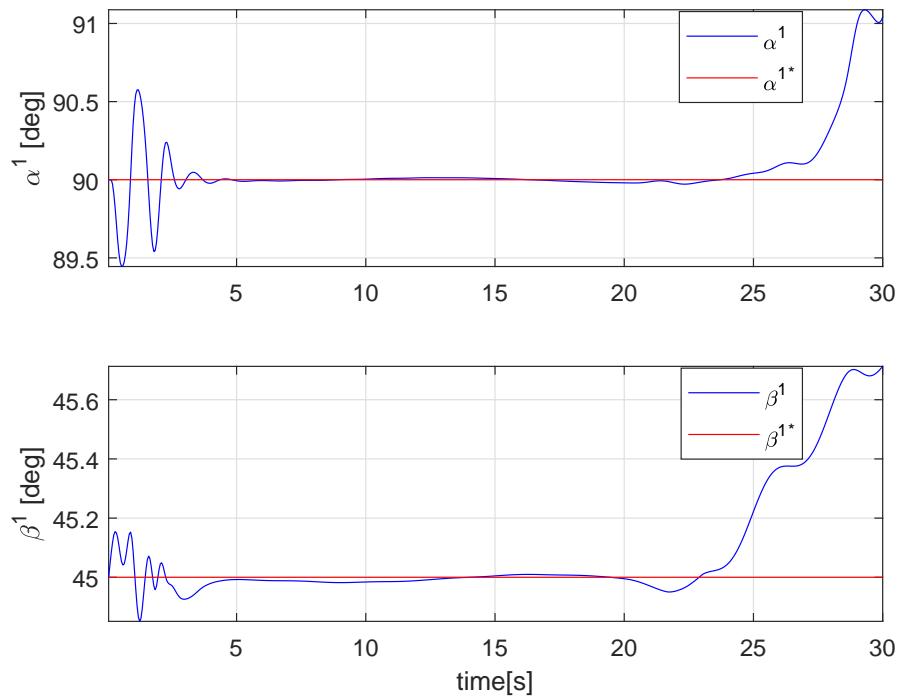
FIGURE 5.15: Rope angles of the first quadrotor using the LMPC controller
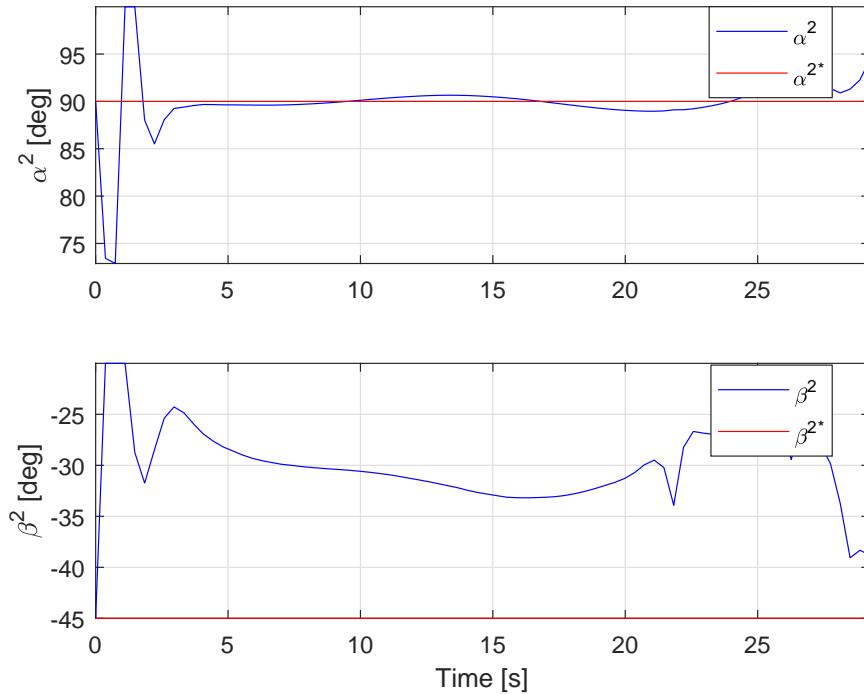
FIGURE 5.16: Rope angles of the first quadrotor using the NMPC controller
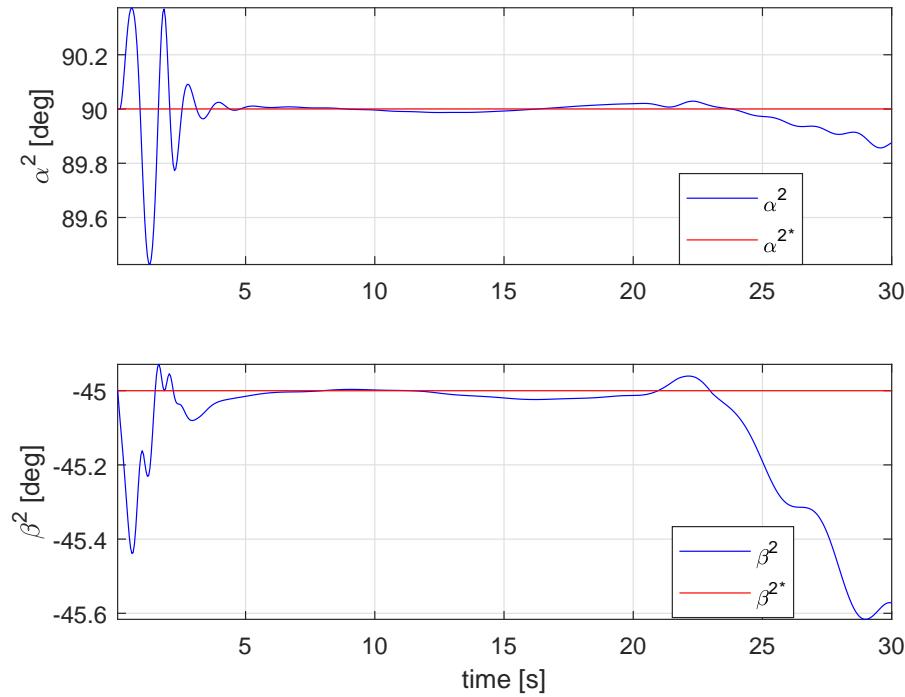
FIGURE 5.17: Rope angles of the first quadrotor using the LMPC controller

The comparisons of the NMPC controller with the LMPC controller on the payload position and the Euler angles of the first quadrotor are shown in Figures 5.18 and 5.19. It can be seen that the NMPC controller has a better response to maintain the accuracy and robustness against the disturbances than the previous LMPC controller.
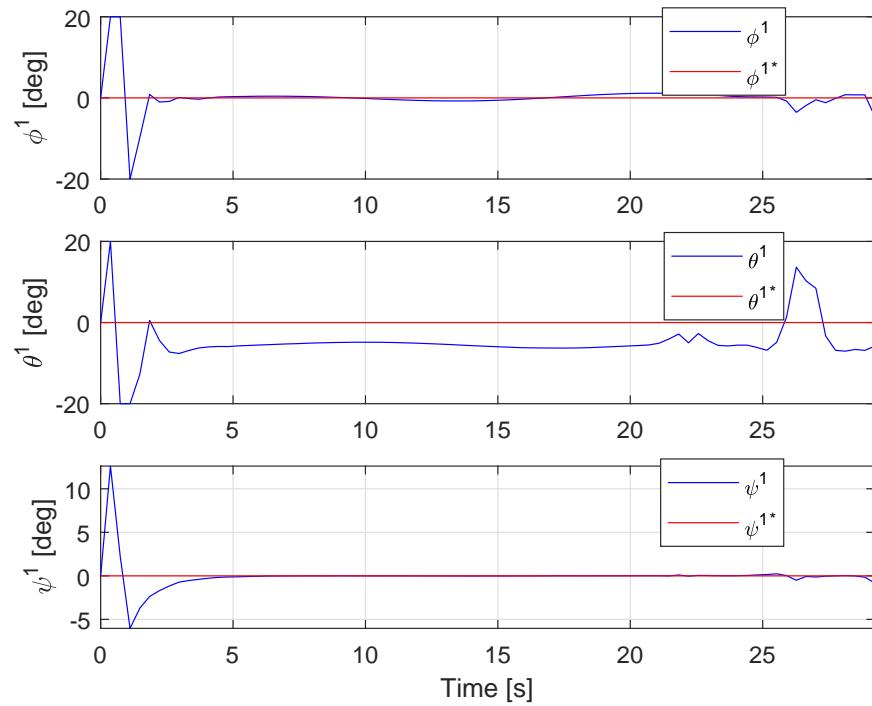
FIGURE 5.18: Eular angles of the first quadrotor using the NMPC controller
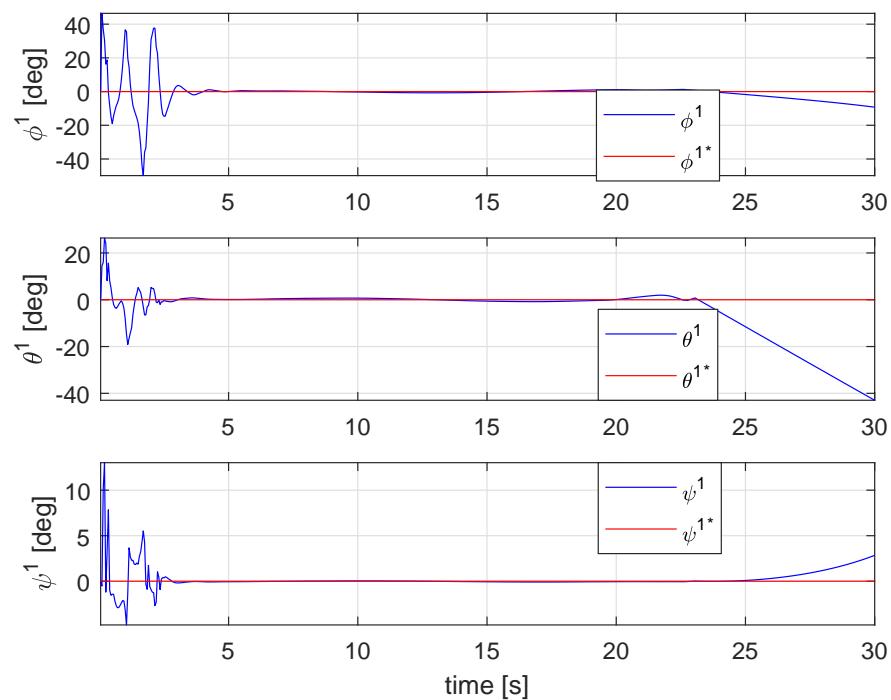
FIGURE 5.19: Eular angles of the first quadrotor using the LMPC controller

Similarly, the system performance is clarified by using the eight shape trajectory under the same environment in Figures 5.20 and 5.21. In Figure 5.20 the NMPC controller fails to complete the tracking tasks, because of the applied disturbances while the constraints are employed. In Figure 5.21, the LMPC controller has a better performance in handling the disturbances with the control and state constraints.
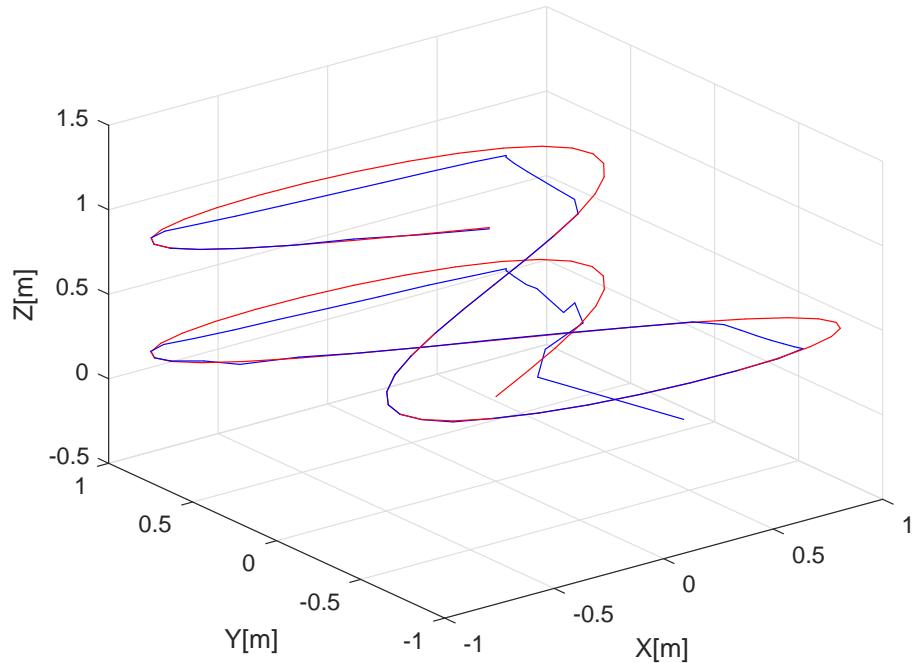
FIGURE 5.20: 3D payload position using the NMPC controller

	Eight			Spiral		
RMSE	$x_P(m)$	$y_P(m)$	$z_P(m)$	$x_P(m)$	$y_P(m)$	$z_P(m)$
NMPC	0.00125	0.00172	0.00241	0.0052	0.0032	0.00152
LMPC	0.0024	0.0073	0.0027	0.0054	0.0042	0.00331

TABLE 5.2: Payload position RMSE values for the two trajectories under NMPC and LMPC Controllers
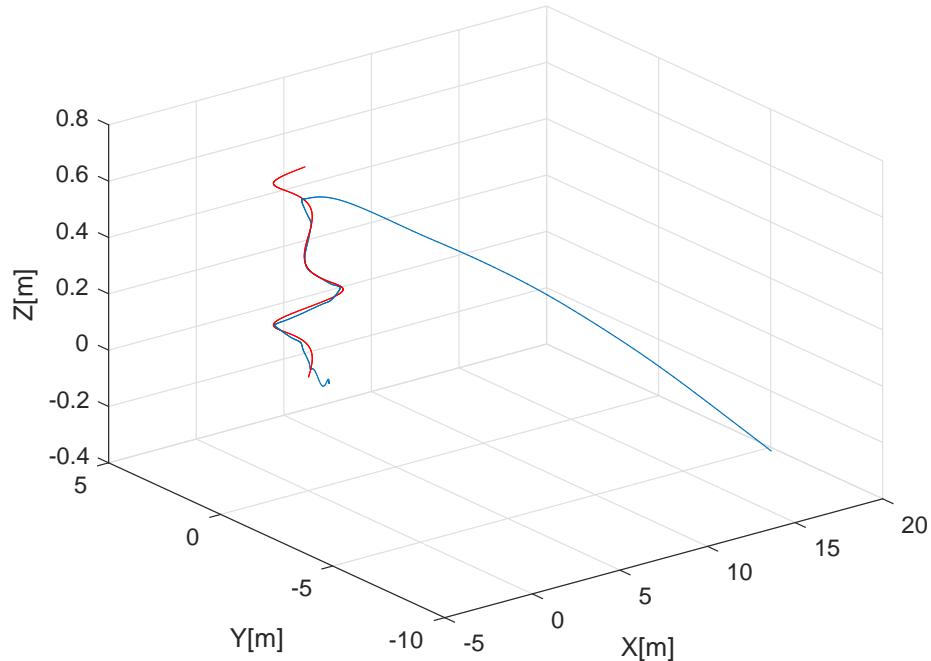
FIGURE 5.21: 3D payload position using the LMPC controller

The load position RMSE values for the two paths are demonstrated in Table 5.2

5.5 Summary

In this chapter, two constrained MPC controllers were applied for tracking control of cable suspended payload with two quadrotors: the LMPC controller and the NMPC controller. The applied constraints on the control input and state vectors were considered.

The simulation results show that the LMPC controller is better in controlling the system than the LQR controller under the consideration of constraints. The constraints are an important factor in handling the environmental limitations and quadrotors' power consumptions. Therefore, applying a controller while considering constraints is beneficial to the practical system. The simulation results also show that the NMPC controller outperforms the LMPC controller when external disturbance is applied.

In general, a much better performance is demonstrated by the NMPC controller compared to the LMPC controller, which is due to the ability of the NMPC controller in handling external disturbances under the constraints. The LMPC controller is struggling to handle the disturbances because of the system linearisation. So far the system is controlled by using a single cost function. In the next chapter, multiple cost functions are considered in order to gain more flexibility in controlling the two quadrotors with payload.

Chapter 6

Leader-Follower Dynamic Game Controller

6.1 Introduction

Dynamic game theory is a mathematical approach to design a playable system based on analysing interaction behaviour between intelligent agents without human involvement. The agents are working in non-cooperative and cooperative manners and consistently making a decision based on information knowledge between players and task management to achieve the target. Stackelberg is presented to solve problems in many fields to obtain an optimal performance based on non-cooperative and cooperative approaches in competitive environments.

A cooperative Stackelberg game theory is proposed for leader and follower quadrotors with suspended payload using an optimal strategy based on the form of linear quadratic game approach in this chapter. Two quadrotors cooperate as a team making their decisions independently, where the first announcement is given by the leader quadrotor as a punishment or a prize in a way that this decision has sufficient information about the right final desired target and follower control. The follower quadrotor receives this announcement from the leader as an incentive or threat to create its own projected decision.

Separate control inputs are obtained based on the leader and follower quadrotors cooperatively in order to determine the best performance in terms of rejection steady-state errors and system stability achievement. The main advantage of utilising Stackelberg game approach is to achieve the task while avoiding motion deadlock, solving conflict situations between the team agents and escaping inefficient behaviour [104] and [105]. The Stackelberg game controller is designed to track desired trajectories where the leader tries to follow the reference path and in charge of stabilising it's orientations. While the follower is in charge of following the leader quadrotor and stabilising the suspended load as well as the leader's attitudes utilising the incentive cooperation strategy.

6.2 Game Methodology and Implementation

Dual control scenarios are presented to perform the Stackelberg game method for two quadrotors carrying payload by cables to track the desired trajectory and achieve the requirements of the desired environment. In this chapter, the leader-follower collaboration strategy based on game theory method is examined. In this method, an efficient formulation is offered by Stackelberg game approach to find an equilibrium point of the involved leader-follower own decisions with the suspended load. The contributions of this work are presented by designing a new mathematical model based on the proposed game method through solving the discrete time of the dynamic model and its cost function. Furthermore, this work attempts to avoid the deadlock state avoidance obtained by wrong decisions from the leader-follower quadrotors, through maintaining a suitable team formation for a specific environment [105]. The incentive concept is designed through relying on the leader quadrotor's first action announcement, whereas the follower quadrotor responds to the leader in order to achieve system stability. The discrete-time dynamical model description for the system with two quadrotors carrying a cable-suspended payload is shown in (6.1)

$$\boldsymbol{x_{k+1}} = \boldsymbol{A_k x_k} + \boldsymbol{B_k u_k}. \tag{6.1}$$

6.3   Linear Dynamic Game Control

A system configuration of the leader and follower and suspended load model can be considered in the following state equation:

$$\boldsymbol{x_{k+1}} = \boldsymbol{A_k x_k} + \sum_{i=1}^{2} [\boldsymbol{B_{ik} u_{ik}}] \tag{6.2}$$

where $\boldsymbol{k}$ is the current instant, $\boldsymbol{x_k}$ is the system state vector $(m \times 1)$ at step $\boldsymbol{k}$ time and $\boldsymbol{u_{ik}^T}$ for $i = 1, 2$ are $(p \times 1)$ decision control vectors generated by the leader and follower quadrotors respectively. Assuming both control vectors have the same dimension, each quadrotor can choose its own strategy from an admissible set of strategies denoted by Γ_i in order to achieve a minimum cost function J_i.

$$J_i = 1/2 \sum_{\boldsymbol{k}=0}^{N-1} \boldsymbol{x_k^T} Q_{i\boldsymbol{k}} \boldsymbol{x_k} + \boldsymbol{u_{ik}^T} R_{ii\boldsymbol{k}} \boldsymbol{u_{ik}} \\ + \boldsymbol{u_{jk}^T} R_{ij\boldsymbol{k}} \boldsymbol{u_{ik}}. \tag{6.3}$$

Subscripts $i, j = 1, 2$, where $(i \neq j)$ represents quadrotor desecion, $R_{ii\boldsymbol{k}}$ and $R_{ij\boldsymbol{k}}$ are symmetric $(p \times p)$ and positive definite matrices, $Q_{i\boldsymbol{k}}$ is $(m \times m)$ symmetric positive semi-definite matrix at $\boldsymbol{k}$ sampling time.

The system state equation described by (6.2) and (6.3) becomes

$$\boldsymbol{x_{k+1}} = \boldsymbol{A_k x_k} + \boldsymbol{B_1 u_{1k}} + \boldsymbol{B_2 u_{2k}} \tag{6.4}$$

where $\boldsymbol{A_k}$ is a $(m \times m)$ system transition matrix and $\boldsymbol{B_{ik}}$ $(i, j = 1, 2)$ are $(m \times p)$ control matrices for the leader quadrotor and the follower quadrotor. In order to achieve the team optimum subject to the leader quadrotor, an incentive approach

is studied using certain policies, where the team optimal control of the leader and follower quadrotors are denoted by $\boldsymbol{u}_1^t$ and $\boldsymbol{u}_2^t$ respectively; and $\boldsymbol{x}^t$ is the state trajectory.

6.3.1 Team Optimal Solution

After assuming that the first quadrotor is a leader, the incentive strategy can be started. The main idea of motivation is to induce the second player (follower quadrotor) to choose a strategy where the team optimum leadership can be achieved. The first step in improving team optimal solution for the leader is calculated on a derivation basis. In order to achieve team optimisation, the leader and follower quadrotors are responsible for optimising the objective function of the leader quadrotor J_1.

$$
\begin{aligned}
J_1 = 1/2 \sum_{\boldsymbol{k}=0}^{N-1} & [\boldsymbol{x_k} - \boldsymbol{x_k^r}]^T Q_{1\boldsymbol{k}}[\boldsymbol{x_k} - \boldsymbol{x_k^r}] \\
& + \boldsymbol{u}_{1\ \boldsymbol{k}}^T R_{11\boldsymbol{k}} \boldsymbol{u}_{1\boldsymbol{k}} + \boldsymbol{u}_{2\ \boldsymbol{k}}^T R_{12\boldsymbol{k}} \boldsymbol{u}_{2\boldsymbol{k}}.
\end{aligned}
\tag{6.5}
$$

The system Hamiltonian is presented as

$$
\begin{aligned}
\mathcal{H}_1 =& 1/2[\boldsymbol{x_k} - \boldsymbol{x_k^r}]^T Q_{1\boldsymbol{k}}[\boldsymbol{x_k} - \boldsymbol{x_k^r}] + \\
& 1/2\boldsymbol{u}_{1\ \boldsymbol{k}}^T R_{11\boldsymbol{k}} \boldsymbol{u}_{1\boldsymbol{k}} + 1/2\boldsymbol{u}_{2\ \boldsymbol{k}}^T R_{12\boldsymbol{k}} \boldsymbol{u}_{2\boldsymbol{k}} + \\
& \lambda_{1\boldsymbol{k+1}}^T \times [\boldsymbol{A_k} \boldsymbol{x_k} + \boldsymbol{B}_{1\boldsymbol{k}} \boldsymbol{u}_{1\boldsymbol{k}} + \boldsymbol{B}_{2\boldsymbol{k}} \boldsymbol{u}_{2\boldsymbol{k}}].
\end{aligned}
\tag{6.6}
$$

by using the minimum principle

$$
\lambda_{1\boldsymbol{k}} = \frac{\partial \mathcal{H}_1}{\partial x_{\boldsymbol{k}}} = Q_{1\boldsymbol{k}}[\boldsymbol{x_k} - \boldsymbol{x_k^r}] + \boldsymbol{A}_{\boldsymbol{k}}^T \lambda_{1\boldsymbol{k+1}}
\tag{6.7}
$$

and

$$
0 = \frac{\partial \mathcal{H}_1}{\partial \boldsymbol{u}_{1\boldsymbol{k}}}
\tag{6.8}
$$

and

$$0 = \frac{\partial \mathcal{H}_2}{\partial \boldsymbol{u_{2k}}} \tag{6.9}$$

where the boundary condition is

$$\lambda_{1N} = Q_{1k}[\boldsymbol{x_N} - \boldsymbol{x_N^r}]. \tag{6.10}$$

The first stage is to obtain the team optimal control of the leader and follower $\boldsymbol{u_1^t}$ and $\boldsymbol{u_2^t}$ based on standard optimal control theory. Using equations (6.8) and (6.9), the following optimal control expressions are obtained:

$$\boldsymbol{u_{1k}} = -R_{11k}^{-1}\boldsymbol{B_{1k}^T}\lambda_{1k+1} \tag{6.11}$$

$$\boldsymbol{u_{2k}} = -R_{12k}^{-1}\boldsymbol{B_{2k}^T}\lambda_{1k+1}. \tag{6.12}$$

The optimal team trajectory is

$$\begin{aligned} \boldsymbol{x_{k+1}} =& \boldsymbol{Ax_k} - \boldsymbol{B_{1k}}R_{11k}^{-1}\boldsymbol{B_{1k}^T}\lambda_{1k+1} \\ &- \boldsymbol{B_{2k+1}}R_{12k}^{-1}\boldsymbol{B_{2k}^T}\lambda_{1k+1}. \end{aligned} \tag{6.13}$$

From the boundary (6.10), it is assumed that for all $k \leq N$

$$\lambda_{1k} = \mathcal{S}_k\boldsymbol{x_k} - \mathcal{V}_k. \tag{6.14}$$

The expressions $\mathcal{S}_k$ and $\mathcal{V}_k$ are the matrix and vector with proper dimensions, respectively. Substitute λ_{1k+1} in (6.13) with (6.14), then the following equation is obtained:

$$\boldsymbol{x_{k+1}} = \mathcal{K}_k^{-1}\boldsymbol{A}\boldsymbol{x_k} + \mathcal{K}_k^{-1}[\boldsymbol{B_{1k}}R_{11k}^{-1}\boldsymbol{B_{1k}^T}]\mathcal{V}_{k+1}$$
$$+ \mathcal{K}_k^{-1}[\boldsymbol{B_{2k+1}}R_{12k}^{-1}\boldsymbol{B_{2k}^T}]\mathcal{V}_{k+1} \tag{6.15}$$

where $\mathcal{K}_k^{-1}$ is defined as

$$\mathcal{K}_k^{-1} = I + \boldsymbol{B_{1k}}R_{11k}^{-1}\boldsymbol{B_{1k}^T}\mathcal{S}_{k+1} + B_{2k}R_{12k}^{-1}B_{2k}^T\mathcal{S}_{k+1} \tag{6.16}$$

and I is an identity matrix. From equation (6.7), by substituting both λ_{1k} and λ_{1k+1}, we have:

$$\mathcal{S}_k\boldsymbol{x_k} - \mathcal{V}_k = Q_{1k}[\boldsymbol{x_k} - \boldsymbol{x_k^r}] + \boldsymbol{A_k^T}\mathcal{S}_{k+1}\boldsymbol{x_{k+1}} - \boldsymbol{A_k^T}\mathcal{V}_{k+1}. \tag{6.17}$$

Substituting $\boldsymbol{x_{k+1}}$ into the above equation and it must hold for all $\boldsymbol{x_k}$, then $\mathcal{S}_k$ and $\mathcal{V}_k$ are obtained:

$$\mathcal{S}_k = Q_{1k} + \boldsymbol{A_k^T}\mathcal{S}_{k+1}\mathcal{K}_k^{-1}\boldsymbol{A_k} \tag{6.18}$$

and

$$\mathcal{V}_k = Q_{1k}\boldsymbol{x_k^r} - \boldsymbol{A_k^T}\mathcal{S}_{k+1}\mathcal{K}_k^{-1}\boldsymbol{B_{1k}}R_{11k}^{-1}\boldsymbol{B_{1k}^T}$$
$$\times \mathcal{V}_{k+1} - \boldsymbol{A_k^T}\mathcal{S}_{k+1}\mathcal{K}_k^{-1}\boldsymbol{B_{2k}}R_{12k}^{-1}\boldsymbol{B_{2k}^T} \tag{6.19}$$
$$\times \mathcal{V}_{k+1} + \boldsymbol{A_k^T}\mathcal{V}_{k+1}$$

with the boundary conditions:

$$\mathcal{S}_N = Q_{1N}$$
$$\mathcal{V}_N = Q_{1N}\boldsymbol{x_N^r}. \tag{6.20}$$

Given $\mathcal{S}_k$ and $\mathcal{V}_k$, the λ_{1k} can be found. Then the team optimal strategies $\boldsymbol{u_{1k}^t}$ and $\boldsymbol{u_{2k}^t}$ are represented as follows:

$$\boldsymbol{u_{1k}^t} = -\mathcal{F}_{11k}\boldsymbol{x_k^t} - \mathcal{F}_{12k}\mathcal{V}_{k+1} \tag{6.21}$$

Also the follower team optimal control $\boldsymbol{u_2^t}$ is presented

$$\boldsymbol{u_{2k}^t} = -\mathcal{F}_{21k}\boldsymbol{x_k^t} - \mathcal{F}_{22k}\mathcal{V}_{k+1} \tag{6.22}$$

where

$$\begin{aligned}
\mathcal{F}_{11k} &= R_{11k}^{-1}\boldsymbol{B}_{1k}^T\mathcal{S}_{k+1}\mathcal{K}_{k}^{-1}\boldsymbol{A_k} \\
\mathcal{F}_{21k} &= R_{12k}^{-1}\boldsymbol{B}_{1k}^T\mathcal{S}_{k+1}\mathcal{K}_{k}^{-1}\boldsymbol{A_k}
\end{aligned} \tag{6.23}$$

$$\begin{aligned}
\mathcal{F}_{12k} &= R_{11k}^{-1}\boldsymbol{B}_{1k}^T\mathcal{S}_{k+1}\mathcal{K}_{k}^{-1}\boldsymbol{B}_{1k} \\
&\quad \times R_{11k}^{-1}\boldsymbol{B}_{1k}^T + R_{11k}^{-1}\boldsymbol{B}_{1k}^T\mathcal{S}_{k+1}\mathcal{K}_{k}^{-1}\boldsymbol{B}_{2k} \\
&\quad \times R_{12k}^{-1}\boldsymbol{B}_{2k}^T - R_{11k}^{-1}\boldsymbol{B}_{1k}^T \\
\mathcal{F}_{22k} &= R_{12k}^{-1}\boldsymbol{B}_{2k}^T\mathcal{S}_{k+1}\mathcal{K}_{k}^{-1}\boldsymbol{B}_{1k} \\
&\quad \times R_{11k}^{-1}\boldsymbol{B}_{1k}^T + R_{12k}^{-1}\boldsymbol{B}_{2k}^T\mathcal{S}_{k+1}\mathcal{K}_{k}^{-1}\boldsymbol{B}_{2k} \\
&\quad \times R_{12k}^{-1}\boldsymbol{B}_{2k}^T - R_{12k}^{-1}\boldsymbol{B}_{2k}^T.
\end{aligned} \tag{6.24}$$

6.3.2   Follower Incentive Response

In the previous stage the team optimum is obtained. In the next stage the follower can be induced by the following incentive strategy:

$$\boldsymbol{u_{1k}} = \boldsymbol{u_{1k}^t} + P_{\boldsymbol{k}}^{\boldsymbol{T}}[\boldsymbol{x_k} - \boldsymbol{x_k^t}] + U_{\boldsymbol{k}}^T. \tag{6.25}$$

To achieve the incentive feedback, the incentive matrices $P_{\boldsymbol{k}}^T$ and $U_{\boldsymbol{k}}^T$ have to be chosen. The follower cost function is

$$\begin{aligned}
J_2 = 1/2 \sum_{\boldsymbol{k}=0}^{N-1} &[\boldsymbol{x_k} - \boldsymbol{x_k^r}]^T Q_{2k}[\boldsymbol{x_k} - \boldsymbol{x_k^r}] \\
&+ \boldsymbol{u_{1\,k}^T}R_{21k}\boldsymbol{u_{1k}} + \boldsymbol{u_{2\,k}^T}R_{22k}\boldsymbol{u_{2k}}.
\end{aligned} \tag{6.26}$$

From the Hamiltonian

$$\mathcal{H}_2 = 1/2[\boldsymbol{x_k} - \boldsymbol{x_k^r}]^T Q_{2k}[\boldsymbol{x_k} - \boldsymbol{x_k^r}] +$$
$$1/2\boldsymbol{u_{1\,k}^T} R_{21k}\boldsymbol{u_{1k}} + 1/2\boldsymbol{u_{2\,k}^T} R_{22k}\boldsymbol{u_{2k}} + \tag{6.27}$$
$$\lambda_{2k+1}^T \times [\boldsymbol{A_k x_k} + \boldsymbol{B_{1k} u_{1k}} + \boldsymbol{B_{2k} u_{2k}}]$$

and by using the minimum principle

$$\lambda_{2k}^\star = \frac{\partial \mathcal{H}_2}{\partial \boldsymbol{x_k}}$$
$$= Q_{2k}[\boldsymbol{x_k^\star} - \boldsymbol{x_k^r}] - P_k R_{21k}[\mathcal{F}_{11k} + P_k^T]\boldsymbol{x_k^t}$$
$$+ [\boldsymbol{A_k} + \boldsymbol{B_{1k}} P_k^T]^T \lambda_{2k+1}^\star - P_k R_{21k} \mathcal{F}_{12k} \mathcal{V}_{k+1} \tag{6.28}$$
$$+ P_k R_{21k} U_k^T,$$

The boundary condition becomes

$$\lambda_{2N}^\star = Q_{2k}[\boldsymbol{x_N^\star} - \boldsymbol{x_N^r}] \tag{6.29}$$

and

$$0 = \frac{\partial \mathcal{H}_2}{\partial \boldsymbol{u_{2k}}} \tag{6.30}$$

where $\boldsymbol{x_k^\star}$ is the state vector when $\boldsymbol{u_{1k}}$ and $\boldsymbol{u_{2k}^\star}$ are applied to the system.

When considering the leader team optimal control $\boldsymbol{u_{1k}}$ and the follower reaction $\boldsymbol{u_{2k}^\star}$ in the system, the state sequence is $\boldsymbol{x_k^\star}$. Therefore $\boldsymbol{u_{2k}^\star}$ can be found from equation (6.30) as follows:

$$\boldsymbol{u_{2k}^\star} = -R_{22\,k}^{-1} \boldsymbol{B_{2k}^T} \lambda_{2k+1}^\star \tag{6.31}$$

where $\lambda_{2k}^\star$ can be assumed as:

$$\lambda_{2k}^\star = \mathcal{M}_k \boldsymbol{x_k^t} + \mathcal{Y}_k[\boldsymbol{x_k^\star} - \boldsymbol{x_k^t}] - \mathcal{D}_k \tag{6.32}$$

$\mathcal{M}$, $\mathcal{Y}$ and $\mathcal{D}$ are convenient matrices.

When applying $\boldsymbol{u_{1k}^t}$ and $\boldsymbol{u_{2k}^t}$ on the system, the state vector is $\boldsymbol{x_k^t}$

$$
\begin{aligned}
\boldsymbol{x_{k+1}^t} =& \boldsymbol{A}\boldsymbol{x_k^t} - \boldsymbol{B_{1k}}[\mathcal{F}_{11k}\boldsymbol{x_k^t} + \mathcal{F}_{12k}\mathcal{V}_{k+1}] \\
& - \boldsymbol{B_{2k}}[\mathcal{F}_{21k}\boldsymbol{x_k^t} + \mathcal{F}_{22k}\mathcal{V}_{k+1}].
\end{aligned}
\tag{6.33}
$$

By representing the equation (6.31), then $\boldsymbol{u_{2k}^\star}$ becomes

$$
\begin{aligned}
\boldsymbol{u_{2k}^\star} =& -R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}[\mathcal{M}_{k+1}\boldsymbol{x_{k+1}^t} \\
& + \mathcal{Y}_{k+1}[\boldsymbol{x_{k+1}^\star} - \boldsymbol{x_{k+1}^t}] - \mathcal{D}_{k+1}].
\end{aligned}
$$

By substituting equation (6.33), we obtain

$$
\begin{aligned}
\boldsymbol{u_{2k}^\star} =& -R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{M}_{k+1} \times [\boldsymbol{A_k}\boldsymbol{x_k^t}] \\
& + R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{M}_{k+1}\boldsymbol{B_{1k}} \times [\mathcal{F}_{11k}\boldsymbol{x_k^t} + \mathcal{F}_{12k}\mathcal{V}_{k+1}] \\
& + R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{M}_{k+1}\boldsymbol{B_{2k}} \times [\mathcal{F}_{21k}\boldsymbol{x_k^t} + \mathcal{F}_{22k}\mathcal{V}_{k+1}] \\
& - R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{Y}_{k+1} \times [\boldsymbol{x_{k+1}^\star} - \boldsymbol{x_{k+1}^t}] \\
& + R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{D}_{k+1}.
\end{aligned}
\tag{6.34}
$$

When the follower acts exactly as the leader expected, $\boldsymbol{u_{2k}^\star} = \boldsymbol{u_{2k}^t} = -\mathcal{F}_{21k}\boldsymbol{x_k^t} - \mathcal{F}_{22k}\mathcal{V}_{k+1}$ and $\boldsymbol{x_k^\star} = \boldsymbol{x_k^t}$. When equation (6.34) must hold for all the states, we have:

$$
\begin{aligned}
F_{21k} =& R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{M}_{k+1} \times [\boldsymbol{A_k} \\
& - \boldsymbol{B_{1k}}F_{11k} - \boldsymbol{B_{2k}}F_{21k}] \\
R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{D}_{k+1} =& R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{M}_{k+1} \\
& \times [[\boldsymbol{B_{1k}}F_{12k} + \boldsymbol{B_{2k}}F_{22k}] - F_{22k}]\mathcal{V}_{k+1}.
\end{aligned}
\tag{6.35}
$$

Based on $\boldsymbol{u_{2k}^t}$, the follower strategy $\boldsymbol{u_{2k}^\star}$, is obtained:

$$
\boldsymbol{u_{2k}^\star} = \boldsymbol{u_{2k}^t} - R_{22\,k}^{-1}\boldsymbol{B_{2k}^T}\mathcal{Y}_{k+1}[\boldsymbol{x_k^\star} - \boldsymbol{x_{k+1}^t}].
\tag{6.36}
$$

To determine the value of $\mathcal{Y}_{k+1}$, applying both $\boldsymbol{u}_{1k}$ and $\boldsymbol{u}_{2k}^{\star}$ to the state equation can give:

$$
\begin{aligned}
\boldsymbol{x}_{k+1}^{\star} =& \boldsymbol{A}_k \boldsymbol{x}_k^{\star} - \boldsymbol{B}_{1k}[\mathcal{F}_{11k}\boldsymbol{x}_k^t + \mathcal{F}_{12k}\mathcal{V}_{k+1}] - \boldsymbol{B}_{1k}[P_k^T \\
& [\boldsymbol{x}_k^{\star} - \boldsymbol{x}_k^t] + U_k^T] - \boldsymbol{B}_{2k}[\mathcal{F}_{21k}\boldsymbol{x}_k^t + \mathcal{F}_{22k}\mathcal{V}_{k+1}] - \\
& \boldsymbol{B}_{2k}R_{22\,k}^{-1}\boldsymbol{B}_{2k}^T\mathcal{Y}_{k+1}[\boldsymbol{x}_{k+1}^{\star} - \boldsymbol{x}_{k+1}^t] \\
=& [\boldsymbol{A}_k + \boldsymbol{B}_{1k}P_k^T][\boldsymbol{x}_k^{\star} - \boldsymbol{x}_k^t] + \boldsymbol{A}_k \boldsymbol{x}_{k+1}^t \\
& - \boldsymbol{B}_{1k}[\mathcal{F}_{11k}\boldsymbol{x}_k^t + \mathcal{F}_{12k}\mathcal{V}_{k+1}] \\
& - \boldsymbol{B}_{2k}[\mathcal{F}_{21k}\boldsymbol{x}_k^t + \mathcal{F}_{22k}\mathcal{V}_{k+1}] \\
& - \boldsymbol{B}_{1k}U_k^T - \boldsymbol{B}_{2k}R_{22\,k}^{-1}\boldsymbol{B}_{2k}^T\mathcal{Y}_{k+1} \\
& [\boldsymbol{x}_{k+1}^{\star} - \boldsymbol{x}_{k+1}^t]
\end{aligned}
\tag{6.37}
$$

or

$$
\begin{aligned}
\boldsymbol{x}_{k+1}^{\star} =& [\boldsymbol{A}_k + \boldsymbol{B}_{1k}P_k^T][\boldsymbol{x}_k^{\star} - \boldsymbol{x}_k^t] - \boldsymbol{B}_{1k}U_k^T \\
& + \boldsymbol{x}_{k+1}^t - \boldsymbol{B}_{2k}R_{22\,k}^{-1}\boldsymbol{B}_{2k}^T \\
& \mathcal{Y}_{k+1}[\boldsymbol{x}_{k+1}^{\star} - \boldsymbol{x}_{k+1}^t].
\end{aligned}
\tag{6.38}
$$

$\boldsymbol{x}_{k+1}^{\star} - \boldsymbol{x}_{k+1}^t$ can be found in terms of $\boldsymbol{x}_k^{\star} - \boldsymbol{x}_k^t$:

$$
\begin{aligned}
\boldsymbol{x}_{k+1}^{\star} - \boldsymbol{x}_{k+1}^t =& [I + B_{2k}R_{22\,k}^{-1}B_{2k}^T \\
& \mathcal{Y}_{k+1}]^{-1} \times [\boldsymbol{A}_k + B_{1k} \\
& P_k^T][\boldsymbol{x}_k^{\star} - \boldsymbol{x}_k^t] + [I + B_{2k} \\
& R_{22\,k}^{-1}B_{2k}^T\mathcal{Y}_{k+1}]^{-1} \times \boldsymbol{B}_{1k}U_k^T.
\end{aligned}
\tag{6.39}
$$

By substituting (6.32), (6.33) and (6.39) into equation (6.28), we obtain:

$$
\begin{aligned}
\mathcal{M}_k x_k^t &+ \mathcal{Y}_k[x_k^\star - x_k^t] - \mathcal{D}_k \\
&= Q_{2k}[x_k^\star - x_k^r] + [A_k + B_{1k}P_k^T]\mathcal{Y}_{k+1} \\
&\quad - P_k R_{21k}\mathcal{F}_{12k}\mathcal{V}_k - P_k R_{21k}[\mathcal{F}_{1k} + P_k^T]x_k^t+ \\
&\quad + P_k R_{21k}\mathcal{Y}_{k+1}[x_{k+1}^\star - x_{k+1}^t] \\
&= Q_{2k}[x_k^\star - x_k^r] + P_k R_{21k}U_k^T \\
&\quad + [A_k + B_{1k}P_k^T]^T[\mathcal{M}_{k+1}x_{k+1}^t - \mathcal{D}_{k+1}] \\
&\quad + [A_k + B_{1k}P_k^T][\mathcal{Y}_{k+1}[I + B_{2k}R_{22\ k}^{-1}B_{2k}^T\mathcal{Y}_{k+1}]^{-1}] \\
&\quad \times [A_k + B_{1k}P_k^T][x_k^\star - x_k^t] - P_k R_{21k}\mathcal{F}_{12k}\mathcal{V}_k \\
&\quad - P_k R_{21k}[\mathcal{F}_{11k} + P_k^T]x_k^t + [A_k + B_{1k}P_k^T] \\
\mathcal{Y}_{k+1} &\times [I + B_{2k}R_{22\ k}^{-1}B_{2k}^T\mathcal{Y}_{k+1}]^{-1} \times B_{1k}U_k^T.
\end{aligned}
\tag{6.40}
$$

The above equation is true for all the values of $x_k^\star$ and x_k^t when the following conditions hold. For all the values of x_k^t term,

$$
\begin{aligned}
\mathcal{M}_k - \mathcal{Y}_k &= -P_k R_{12k}[\mathcal{F}_{11k} + P_k^T] \\
&\quad + [A_k + P_k B_{1k}^T]\mathcal{M}_{k+1} \\
&\quad \times [A_k - B_{1k}\mathcal{F}_{11k} - B_{2k}\mathcal{F}_{21k}] \\
&\quad + [A_k + B_{1k}P_k^T]\mathcal{Y}_{k+1} \\
&\quad \times [I + B_{2k}R_{22\ k}^{-1}B_{2k}^T\mathcal{Y}_{k+1}]^{-1} \\
&\quad \times [A_k + B_{1k}P_k^T].
\end{aligned}
\tag{6.41}
$$

For all the values of $x_k^\star$ term

$$
\begin{aligned}
\mathcal{Y}_k &= Q_{2k} + P_k R_{21k}P_k^T \\
&\quad + [A_k + B_{1k}P_k^T]^T\mathcal{Y}_{k+1} \\
&\quad \times [I + B_{2k}R_{22\ k}^{-1}B_{2k}^T\mathcal{Y}_{k+1}]^{-1} \\
&\quad \times [A_k + B_{1k}P_k^T].
\end{aligned}
\tag{6.42}
$$

Based on the equation (6.40), all constant values are considered in

$$
\begin{aligned}
-\mathcal{D}_{k} = &-Q_{2k}\boldsymbol{x}_{k}^{r} - P_{k}R_{21k}\mathcal{F}_{12k}\mathcal{V}_{k+1} \\
&+ P_{k}R_{21k}U_{k}^{T}[\boldsymbol{x}_{k}^{\star} - \boldsymbol{x}_{k}^{r}] \\
&- [\boldsymbol{A}_{k} + \boldsymbol{B}_{1k}P_{k}^{T}]^{T}\mathcal{M}_{k+1} \\
&\times [\boldsymbol{B}_{1k}\mathcal{F}_{12k} + \boldsymbol{B}_{2k}\mathcal{F}_{22k}]\mathcal{V}_{k+1} \\
&- \mathcal{D}_{k+1} + [\boldsymbol{A}_{k} + \boldsymbol{B}_{1k}P_{k}^{T}]^{T}\mathcal{Y}_{k+1} \\
&\times [I + \boldsymbol{B}_{2k}R_{22\,k}^{-1}\boldsymbol{B}_{2k}^{T}\mathcal{Y}_{k+1}]^{-1}\boldsymbol{B}_{1k}U_{k}^{T}
\end{aligned}
\tag{6.43}
$$

where the gain matrix U_{k}^{T} can be calculated from equation (6.43). By substituting equation (6.42) into (6.41), the resultant equation is

$$
\begin{aligned}
\mathcal{M}_{k} = &Q_{2k} - P_{k}R_{21k}\mathcal{F}_{11k} \\
&+ [\boldsymbol{A}_{k} + \boldsymbol{B}_{1k}P_{k}^{T}]^{T} \times \mathcal{M}_{k+1} \\
&[\boldsymbol{A}_{k} - \boldsymbol{B}_{1k}\mathcal{F}_{11k} - \boldsymbol{B}_{21k}\mathcal{F}_{2k}].
\end{aligned}
\tag{6.44}
$$

If the matrices P_{k}^{T} and U_{k}^{T} can be found by the leader agent to achieve the optimal follower strategy, then this matrix satisfies all the above mentioned equations [105].

The Stackelberg equilibrium strategy is responsible for presenting the state feedback results, which is the main goal of the leader's team optimal solution. In the case of $\boldsymbol{x}_{k}^{\star} = \boldsymbol{x}_{k}^{t}$, the follower's control reaction is $\boldsymbol{u}_{2k}^{\star} = \boldsymbol{u}_{2k}^{t}$. When adding a special function $f(\boldsymbol{x}_{k}^{\star}, \boldsymbol{x}_{k}^{t})$ to the follower's reaction such as in equation (6.36), this function is chosen such that the follower control is optimal and the system is stable.

The procedure of solving a trajectory tracking error based on the Stackelberg game control is summarised as below. The first step is started to achieve the team optimal control and optimise the leader's cost function J_{1} by both the leader and the follower quadrotors $\boldsymbol{u}_{1k}^{t}$ and $\boldsymbol{u}_{2k}^{t}$, respectively, while the leader quadrotor has a total knowledge of the follower. Then the control reaction of the follower quadrotor can be induced to cooperate with the leader quadrotor and adopt $\boldsymbol{u}_{2k}^{t}$ to

minimise the follower cost function J_2. The dominating strategy of this algorithm procedure can be concisely introduced as follows.

Firstly, after obtaining both the $\mathcal{S}_{k+1}$ matrix and $\mathcal{V}_{k+1}$ vector, calculate the values of $\mathcal{F}_{11k}, \mathcal{F}_{12k}, \mathcal{F}_{21k}$ and $\mathcal{F}_{22k}$ from the following equations $(6.23), (6.24)$. Secondly, find the value of $\mathcal{M}_k$ from the equation (6.35) and the incentive weight matrix P_k using the equation (6.44). Then the values of both matrices $\mathcal{Y}_k$ and $\mathcal{D}_k$ can be determined utilising the equation (6.42). Finally, the additional term U_k is calculated from equation (6.43)

6.4 Simulation Results

The validation of the dynamic game controller is implemented through conducting a Matlab simulation. The dual cost tracking controller is tested in simulation to verify the behaviour of the system. A cable suspended load with two quadrotors by two cables is utilised to track potential desired trajectories. These trajectories are presented with different challenges associated with coordination vectors and are introduced in star, eight shape and combined circle-square trajectories. The multi-sharp edges (star) trajectory is presented as a first desired trajectory used to assess the controller stability achievement. Tracking the desired path is managed by following desired three-star coordinates. These coordinates are presented by six steps for $x^* = [xs1; xs2; xs3; xs4; xs5; xs6]$, y desired vector $y^* = [ys1; ys2; ys3; ys4; ys5; ys6]$ and z desired height $z^* = 0.1 * t$. Each step consists of a step function such that this function is comprised of a starting point, an ending point, a time and a simulation frequency $[xs] = step(xs_{start}, xs_{end}, time, traj.T)$. In one step of each vector, all the desired points can be obtained at each sampling time by starting from the first point of this desired step to the desired ending point of this specific step along the time vector, where the time vector of the three trajectories is started from the initial zero operating point with step sampling $0.02s$.lead to the final $30s$ as in $time = (0 : 0.02 : 30)^T$. This time includes 1500

control points through all these trajectories and the required sampling frequency is $50Hz$.

The second reference trajectory is eight shape trajectory. This is presented in $x^* = 0.1tsin(0.3t)$, $y^* = 0.1tsin(0.3t)$ and $z^* = 0.1 * t$. In the third trajectory, the circle-square path is proposed to show the system behaviour utilising new potential trajectory coordinate vectors. These vectors are presented in 8-steps of each vector considering all curves and lines in the combined reference path. The operational point of the two quadrotors with slung load system is proposed with rope angles $\alpha^1 = 45^o, \beta^1 = 0^o, \alpha^2 = 135^o, \beta^2 = 0^o$. The system weight matrices are the states and control weight matrices based on the generalised coordinates sequence $\boldsymbol{q} = [x_P, \dot{x}_P, y_P, \dot{y}_P, z_P, \dot{z}_P, \ \alpha^{1k}, \dot{\alpha}^{1k}, \beta^{1k}, \dot{\beta}^{1k}, \phi^{1k}, \dot{\phi}^{1k}, \theta^{1k}, \dot{\theta}^{1k}, \psi^{1k}, \dot{\psi}^{1k}$ $\alpha^{2k}, \dot{\alpha}^{2k}, \beta^{2k}, \dot{\beta}^{2k}, \phi^{2k}, \dot{\phi}^{2k}, \theta^{2k}, \dot{\theta}^{2k}, \ \psi^{2k}, \dot{\psi}^{2k}]^T$ and control sequence $\boldsymbol{u} = [F_z^{1k}, M_x^{1k}, M_y^{1k}, M_z^{1k}, F_z^{2k}, M_x^{2k}, M_y^{2k}, M_z^{2k}]^T$ in the form of diagonal matrices as:

$$
\begin{aligned}
Q_{1\boldsymbol{k}} =diag([&100, 1.4286, 100, 0.5882, 10, 1.4286, 5.7296, \\
&5.7296, 1.1459, 0.1146, 0.1146, 0.1146, 1.1459, \\
&0.1146, 0.1146, 0.1146, 0.1146, 0.2865, 0.1146, 0.1146, \\
&0.1146, 0.1146, 0.1146, 0.2865, 0.1146, 0.1146])
\end{aligned}
$$

$$
\begin{aligned}
Q_{2\boldsymbol{k}} =diag([&1000, 143, 33, 0.0001, 100, 0.1, 573, 6, 0.0001, \\
&0.0001, 0.0001, 0.0001, 0.0001, 0.0001, 0.0001, \\
&0.0001, 573, 6, 115, 0.0001, 0.0001, 0.0001, \\
&0.0001, 0.0001, 0.0001, 0.0001]).
\end{aligned}
$$

Also the leader and the follower weight matrices are as follows:

$$R_{11} = diag([1, 0.01, 0.01, 0.01])$$

$$R_{12} = diag([0.1, 0.001, 0.001, 0.001])$$

$$R_{21} = diag([0.1, 0.001, 0.001, 0.001])$$

$$R_{22} = diag([1, 0.01, 0.01, 0.01]).$$

The results are clarified by Figures 6.1 - 6.8 for star trajectory, Figures 6.17 for eight shape trajectory and Figures 6.19 and 6.20 for a combined circle-square trajectory, where the generated thrust of each quadrotor is transformed in Figure 6.1 to become $\boldsymbol{F}_Q^k = F_z^k \boldsymbol{e}_3^k$ utilising the body to inertial frame $\boldsymbol{S_e}$. The payload position is show through decreasing the tracking steady-state errors compared with LQR controller as presented in Figures 6.2 and 6.10. Meanwhile, both quadrotors are trying to maintain both their orientations as shown in Figures 6.3 and 6.4 with the two rope angles. This is required to meet the minimum swing angles $\alpha^1 = 45^o, \alpha^2 = 135^o, \beta^1 = \beta^2 = 0^o$ from the reference paths as clarified in Figures 6.5 and 6.6. As in all these figures, the errors are reduced to the minimum at each tracking point and the system stability is achieved. Handling tracking errors with game control implementation can be seen in the top view and three dimensions in Figures 6.7, 6.8, 6.15 and 6.16.
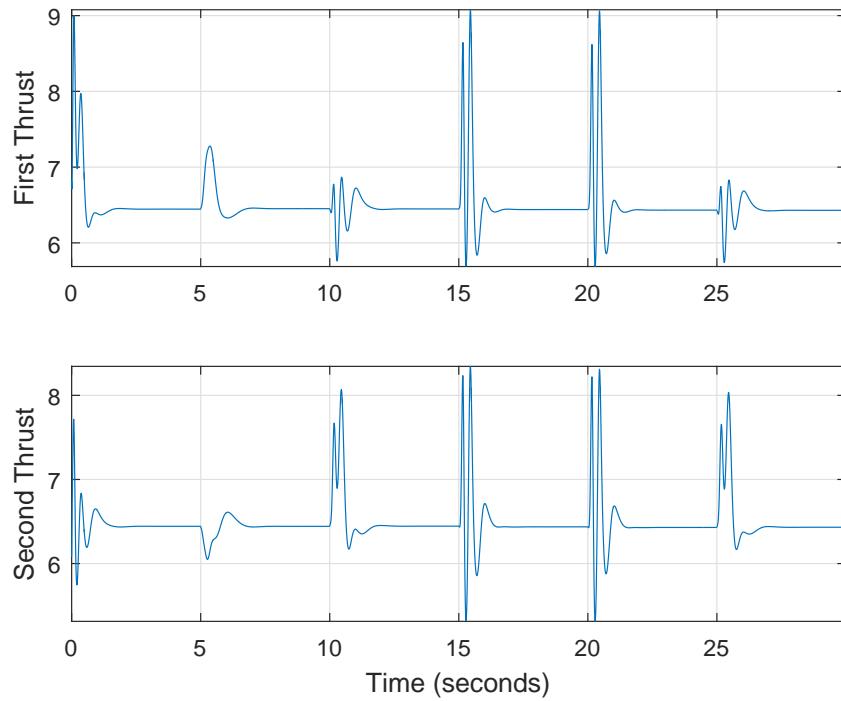
FIGURE 6.1: System thrusts for star trajectory using the dynamic game controller
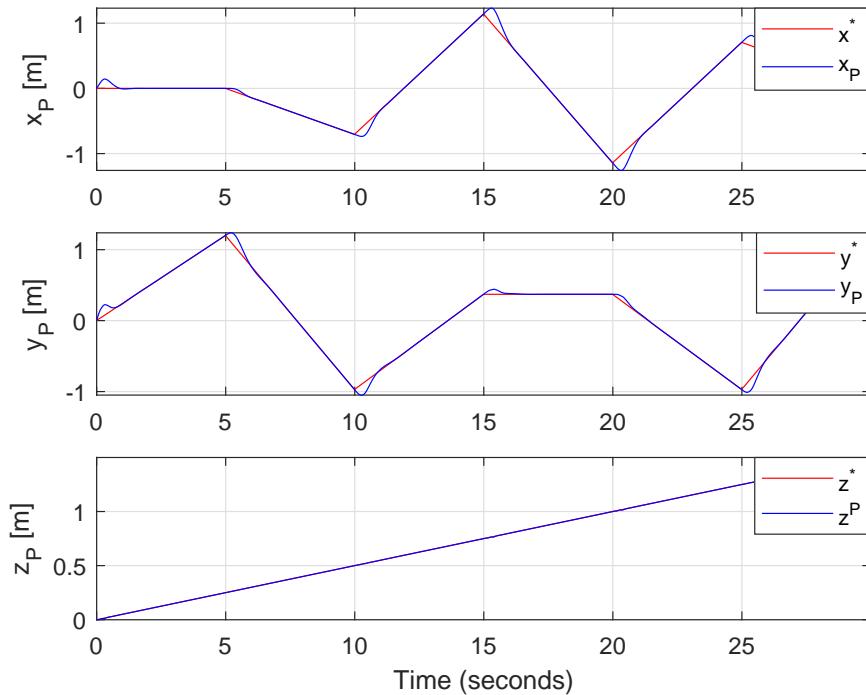
FIGURE 6.2: Payload position representation for star trajectory using the dynamic game controller
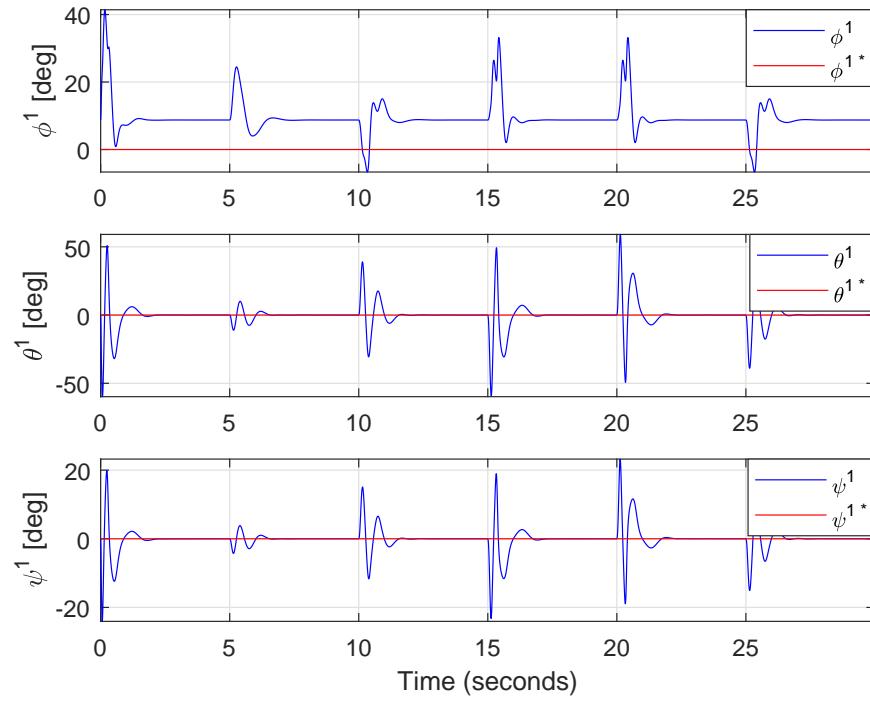
FIGURE 6.3: Angles of the first quadrotor for star trajectory using the dynamic game controller
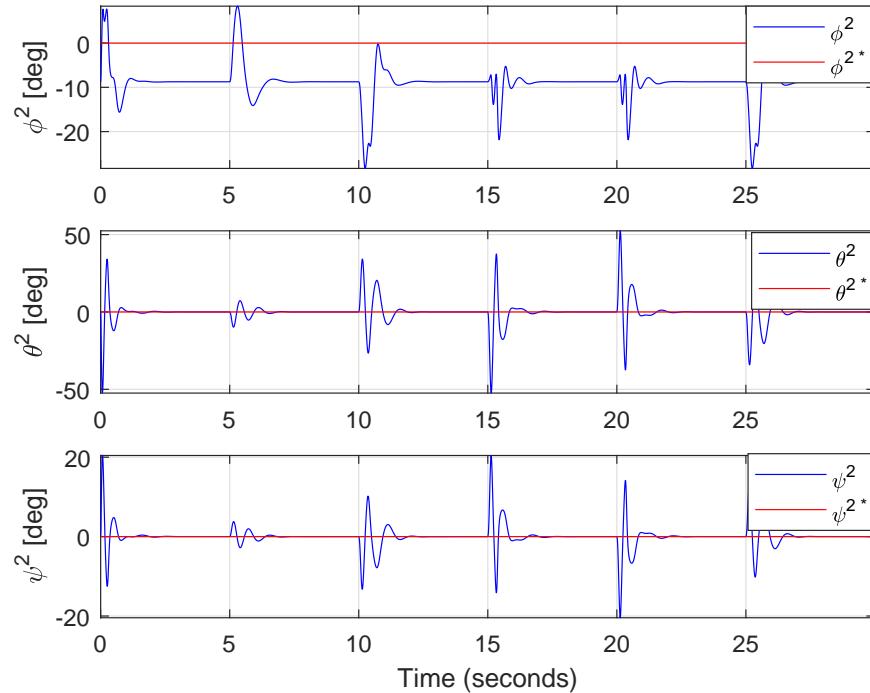
FIGURE 6.4: Angles of the second quadrotor for star trajectory using the dynamic game controller
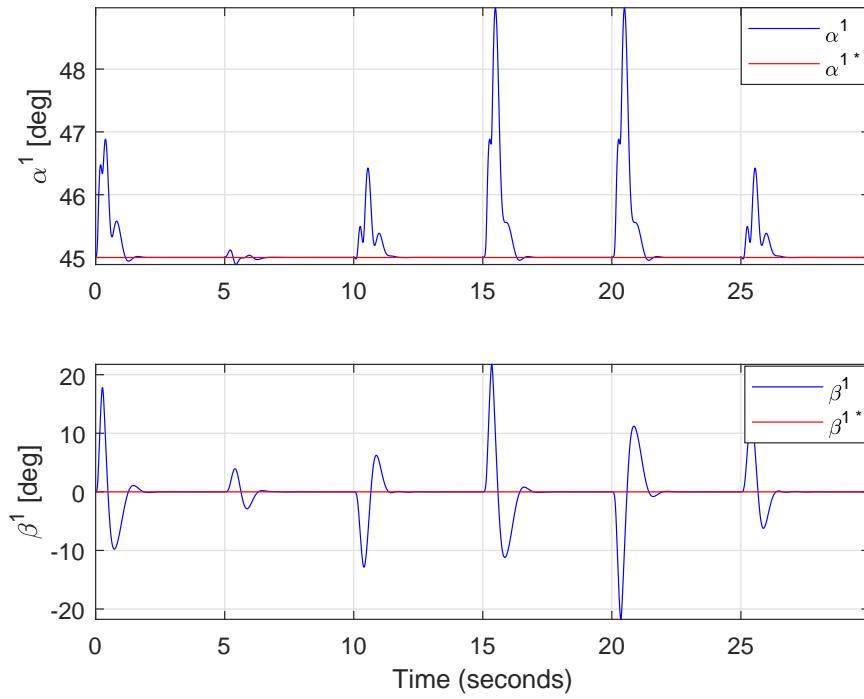
FIGURE 6.5: Rope angles of the first quadrotor-load for star trajectory using the dynamic game controller
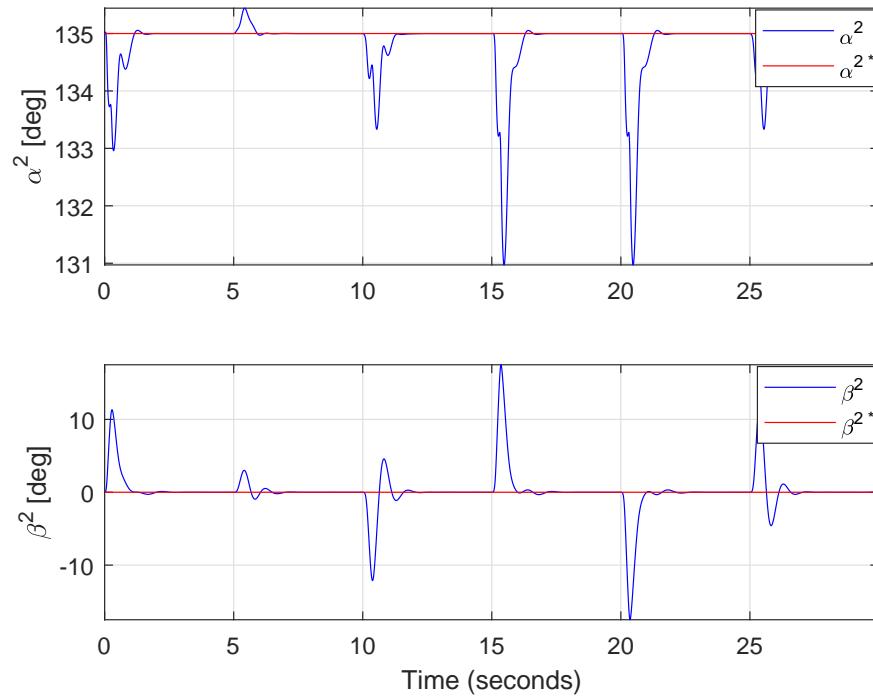
FIGURE 6.6: Rope angles of the second quadrotor-load for star trajectory using the dynamic game controller
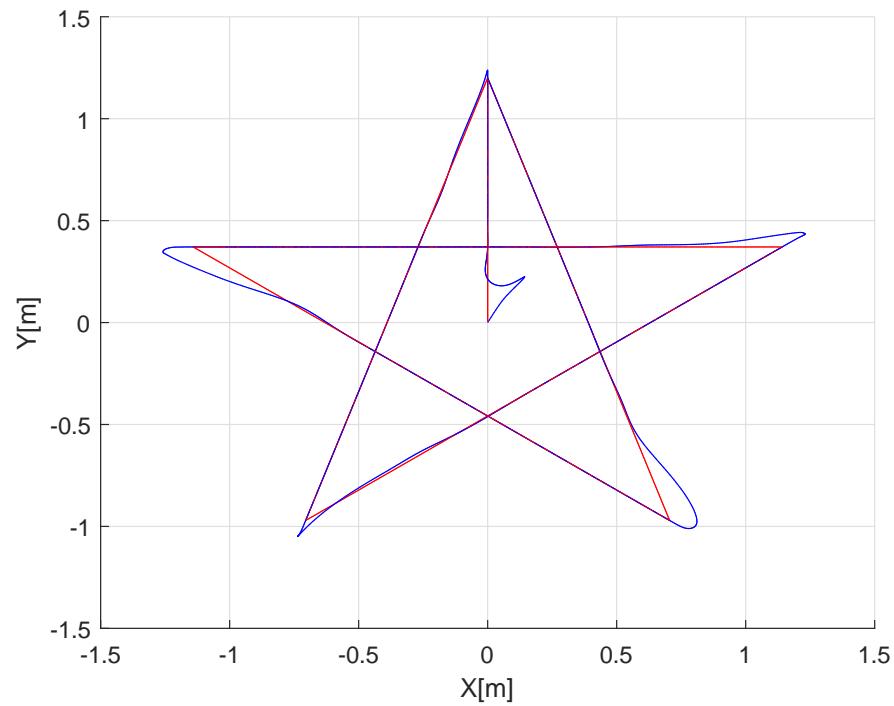
FIGURE 6.7: Load position for star trajectory using the dynamic game controller
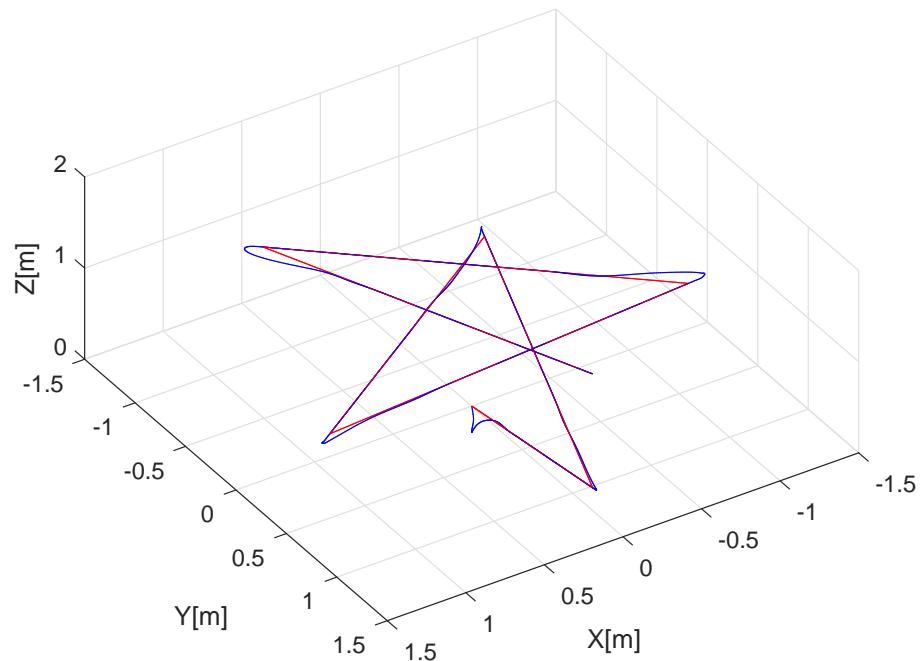
FIGURE 6.8: 3D load position for star trajectory using the dynamic game controller

	Star			Eight		
RMSE	$x_P(m)$	$y_P(m)$	$z_P(m)$	$x_P(m)$	$y_P(m)$	$z_P(m)$
Game	0.0053	0.0009	0.0011	0.00136	0.0054	0.0028
LQR	0.0124	0.0373	0.0132	0.0154	0.0142	0.0331

TABLE 6.1: Payload position RMSE values for the two trajectories under Game and LQR Controllers

The load position RMSE values for the two paths are demonstrated in Table 6.1

In the case of the single decision maker LQR controller, the animation results are obtained for all proposed trajectories using the same system model. The model thrusts and parameters are tested compared with the incentive dynamic game controller in order to verify the best model stability. Figure 6.9 shows the leader and follower thrusts using the LQR controller, while the payload position is displayed in Figure 6.10. In Figures 6.11 and 6.12 both the leader and follower quadrotor's behaviours are presented. The rope angles for all given paths can be clearly identified in Figures 6.13, and 6.14.

Figures 6.16, 6.18 and 6.22 show 3D trajectories. They are used to illustrate that the LQR controller is incapable of automatically taking a cooperative decision for each of the quadrotors, which shows this controller to be less effective than the game controller. The strategy adopted in the game controller allows the leader and the follower quadrotors to take individual decisions automatically in a way that meets the requirements of the strategic decisions for both quadrotors.

Using the LQR controller, the results of a star trajectory are clarified in Figures 6.9-6.16:
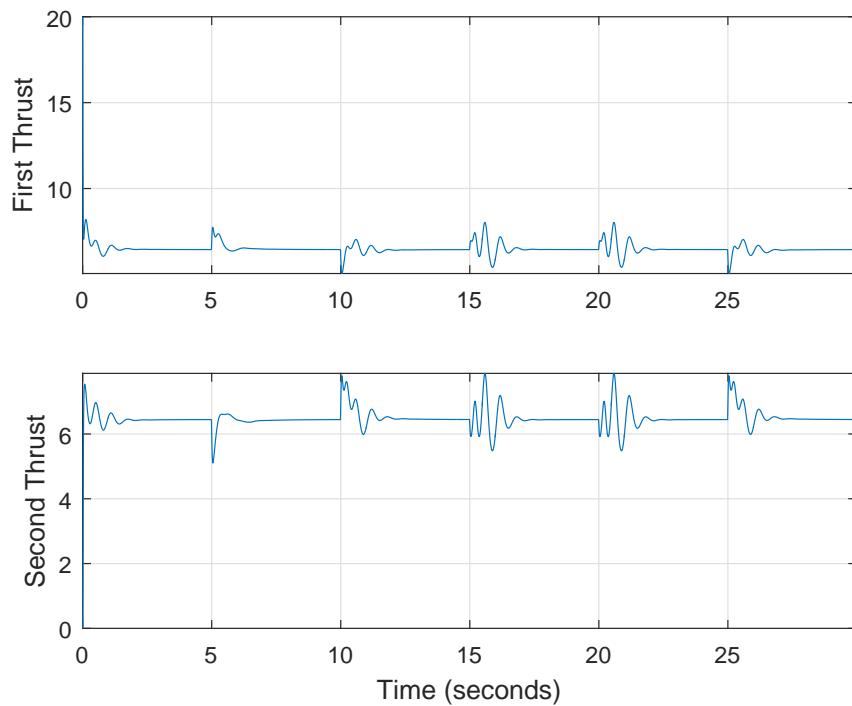
FIGURE 6.9: System thrusts for star trajectory using LQR controller
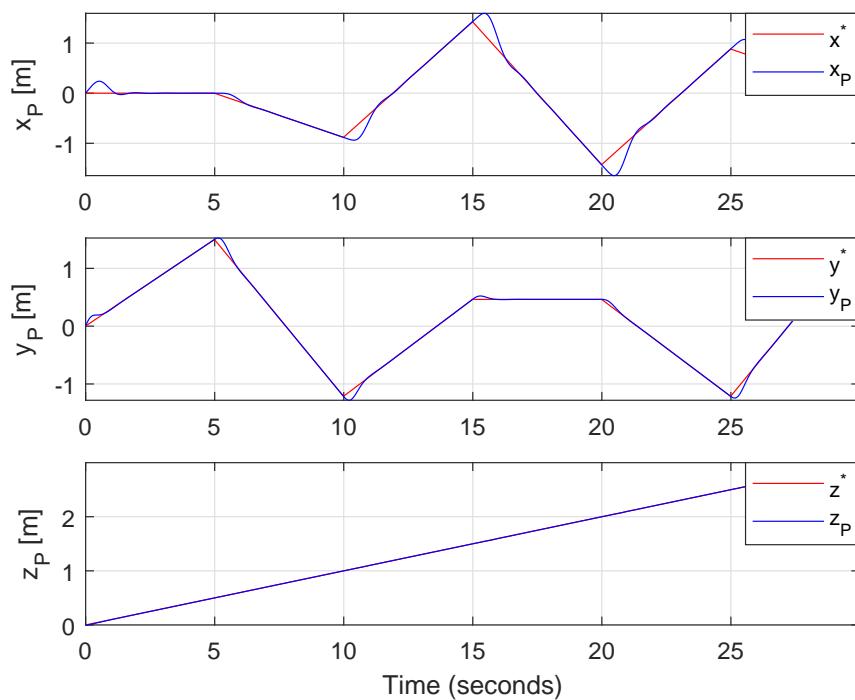
FIGURE 6.10: Payload position representation for star trajectory using LQR controller
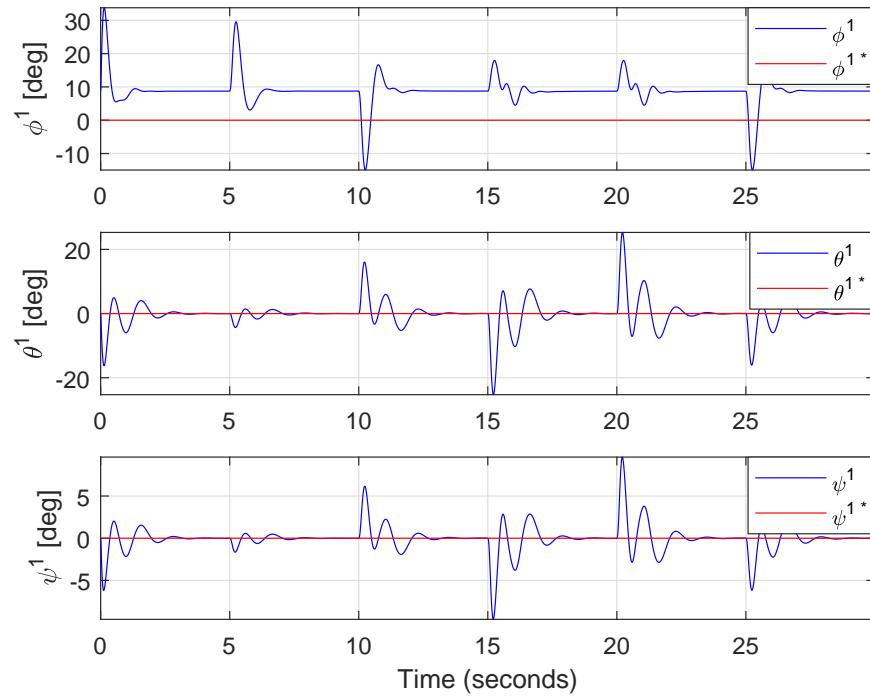
FIGURE 6.11: Angles of the first quadrotor for star trajectory using LQR controller
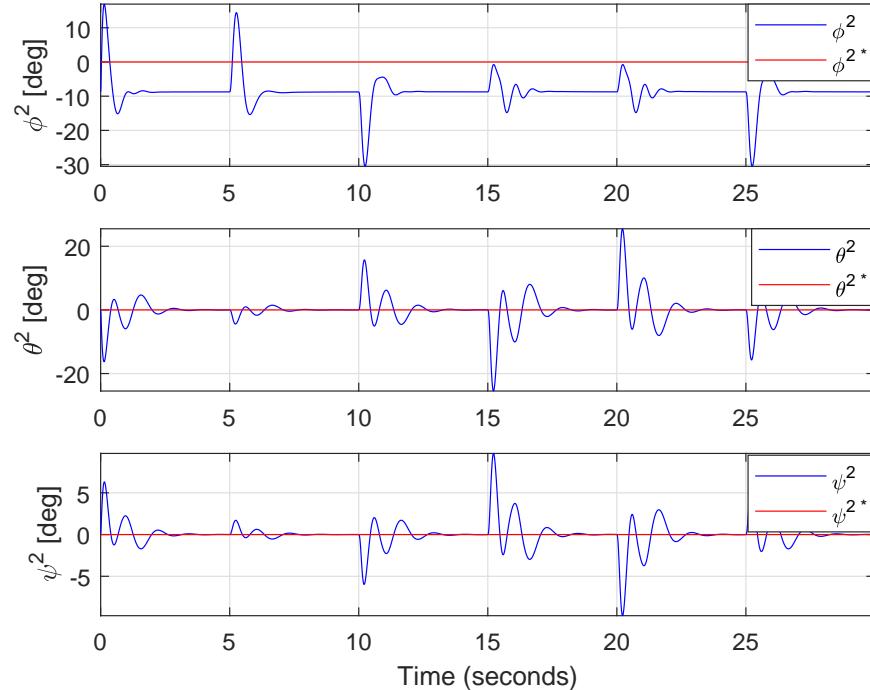
FIGURE 6.12: Angles of the second quadrotor for star trajectory using LQR controller
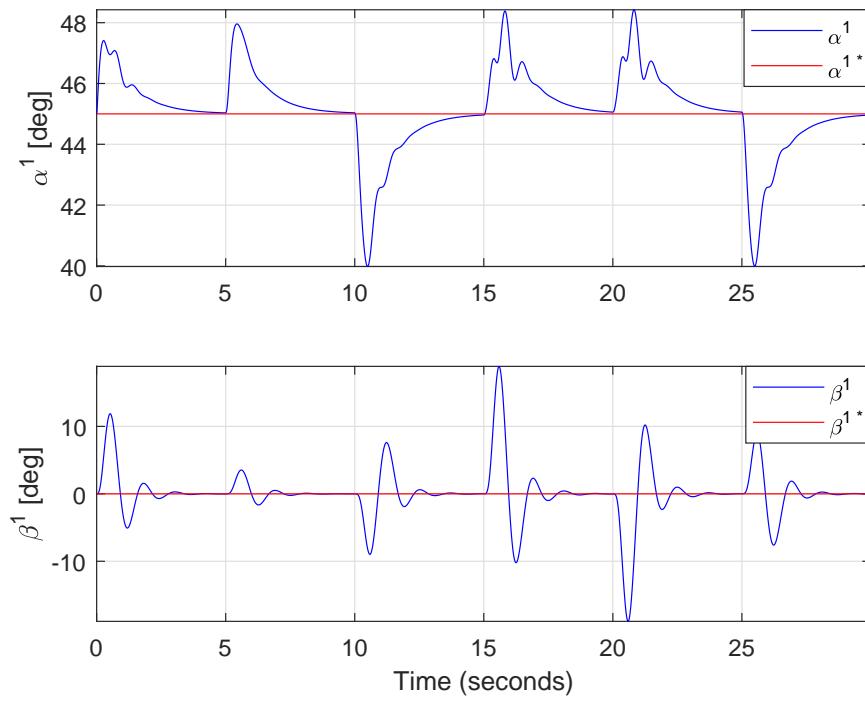
FIGURE 6.13: Rope angels of the first quadrotor-load for star trajectory using LQR controller
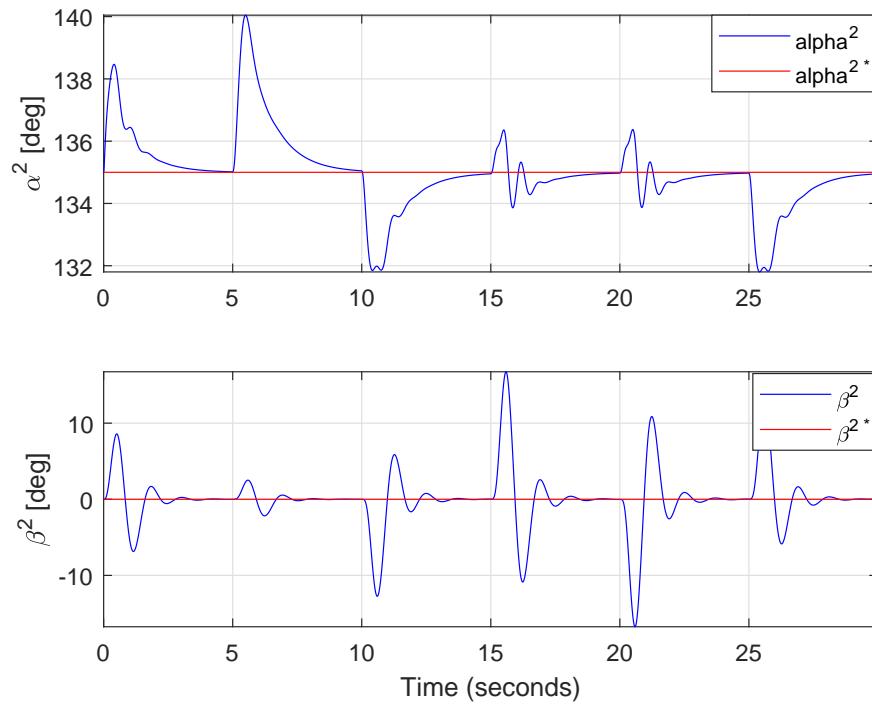
FIGURE 6.14: Rope angles of the second quadrotor-load for star trajectory using LQR controller
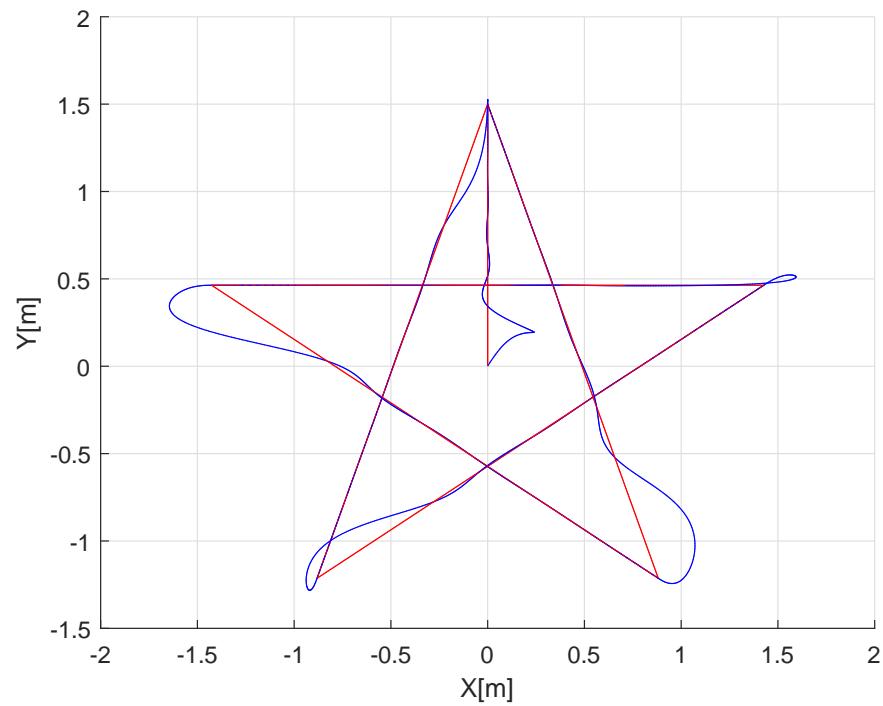
FIGURE 6.15: Load position for star trajectory using LQR controller
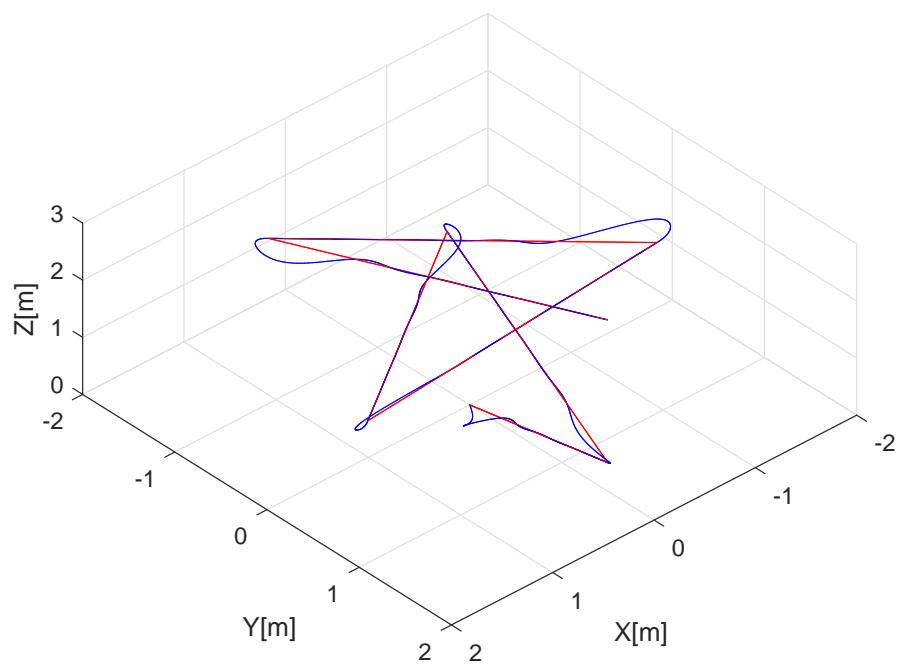
FIGURE 6.16: 3D load position for star trajectory using LQR controller

Likewise, the simulation results of the dynamic game controller are obtained through relying on different predefined paths subjected to different challenges, namely an eight-shape and a mixed circle-square reference paths. These two reference paths are presented to examine the accuracy of the game controller in tracking the trajectories and its capacity to achieve stability of the suspended payload in comparison with the LQR controller. The proposed trajectories are presented in Figures 6.17, 6.18, 6.19, 6.20, 6.21 and 6.22.
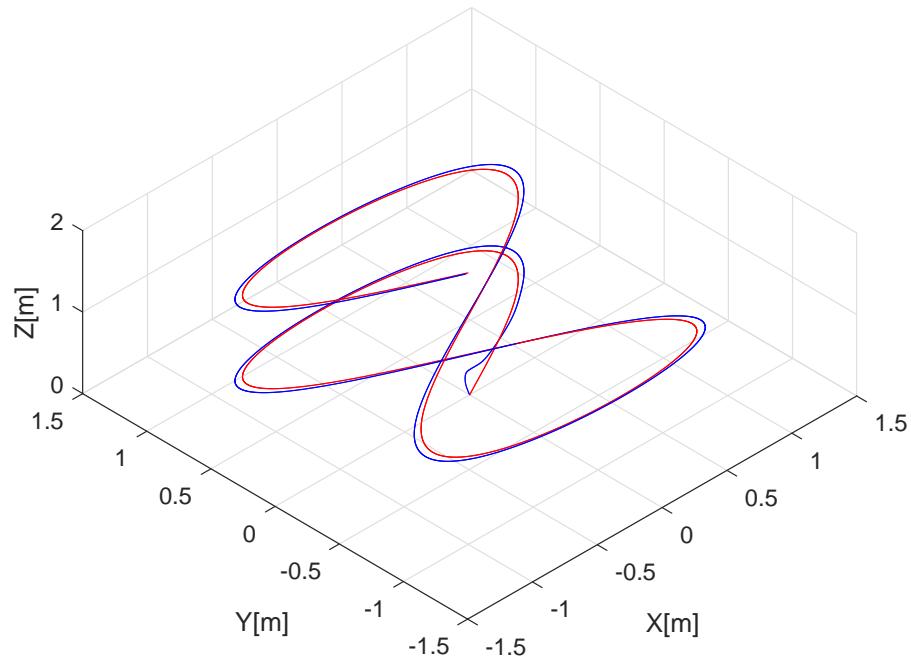
FIGURE 6.17: Load position for eight trajectory using the dynamic game controller
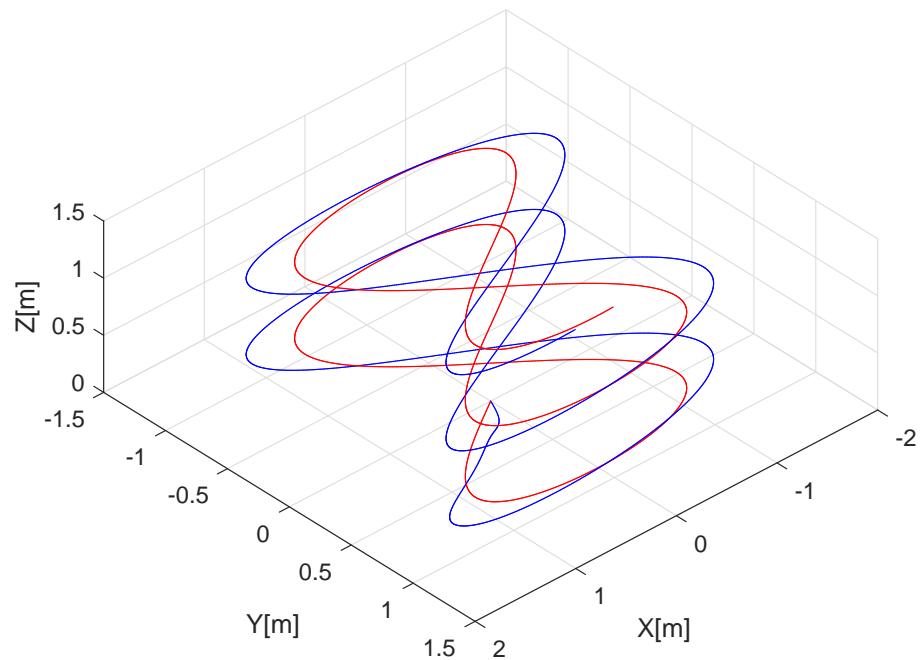
FIGURE 6.18: Load position for eight trajectory using LQR controller
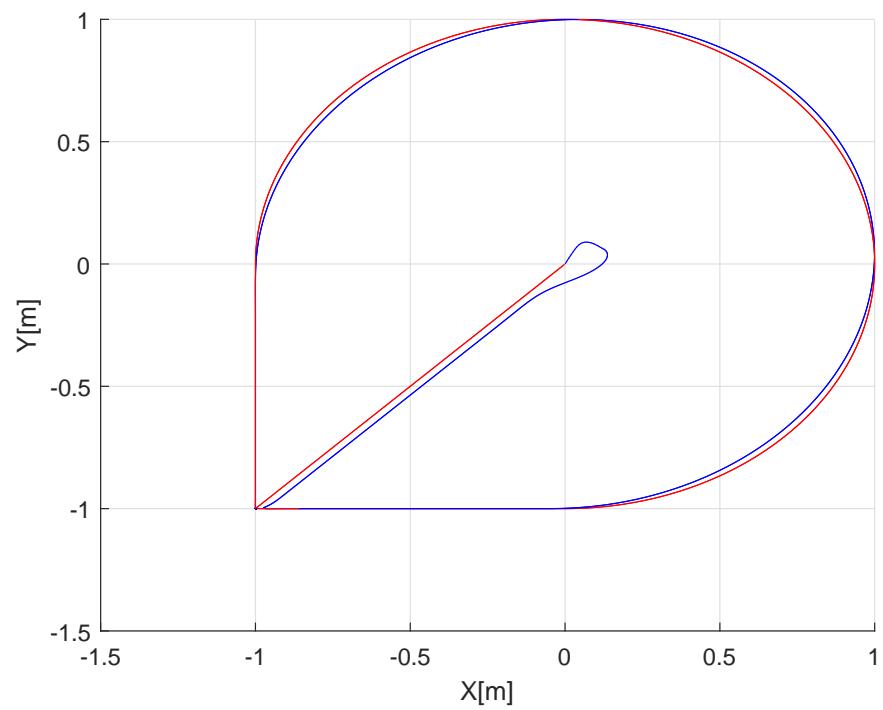
FIGURE 6.19: Load position for circle-square trajectory using the dynamic game controller
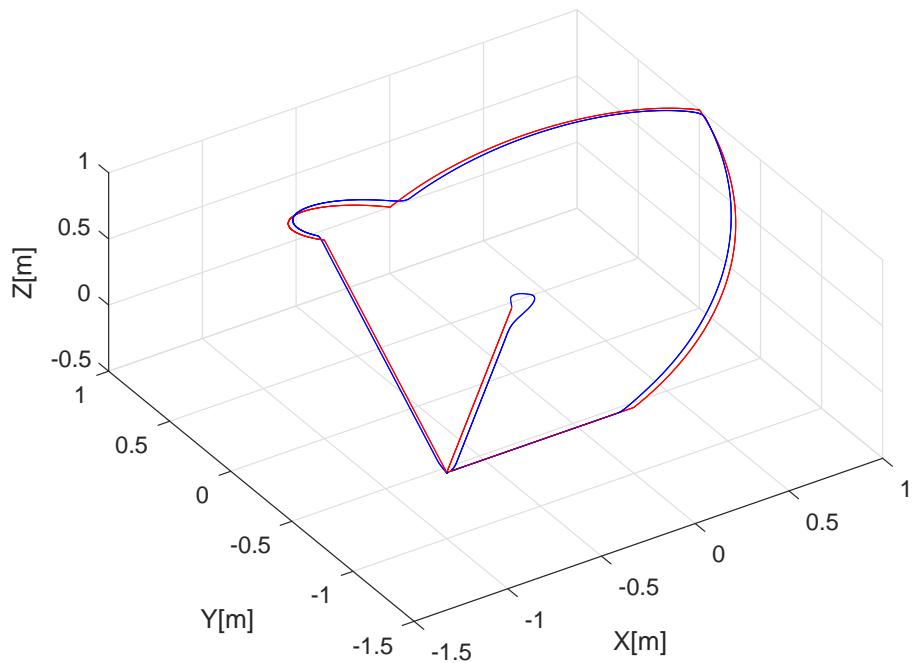
FIGURE 6.20: 3D load position for circle-square trajectory using the dynamic game controller
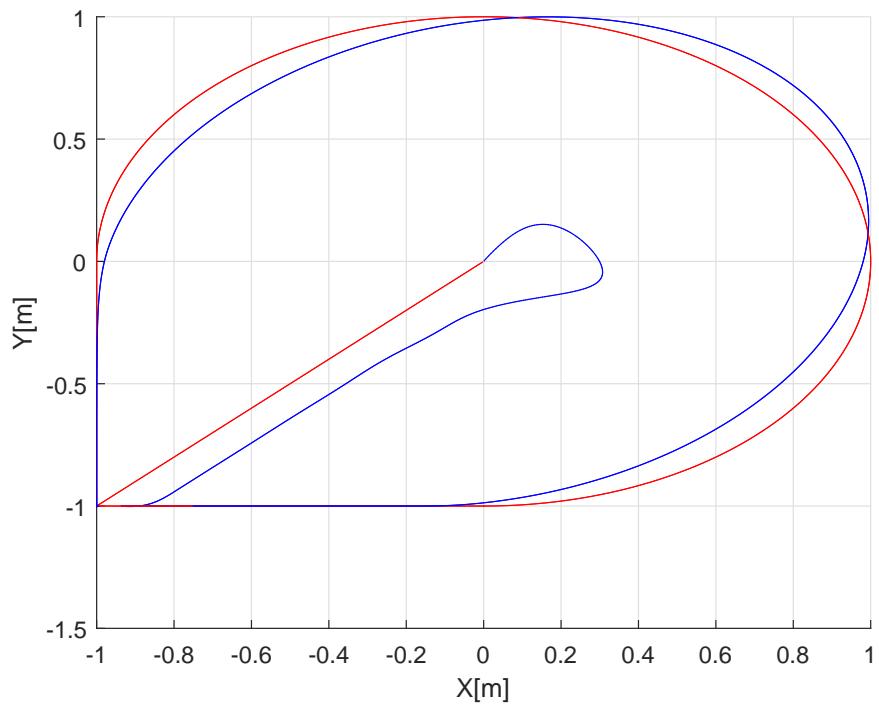
FIGURE 6.21: Load position for circle-square trajectory using LQR controller
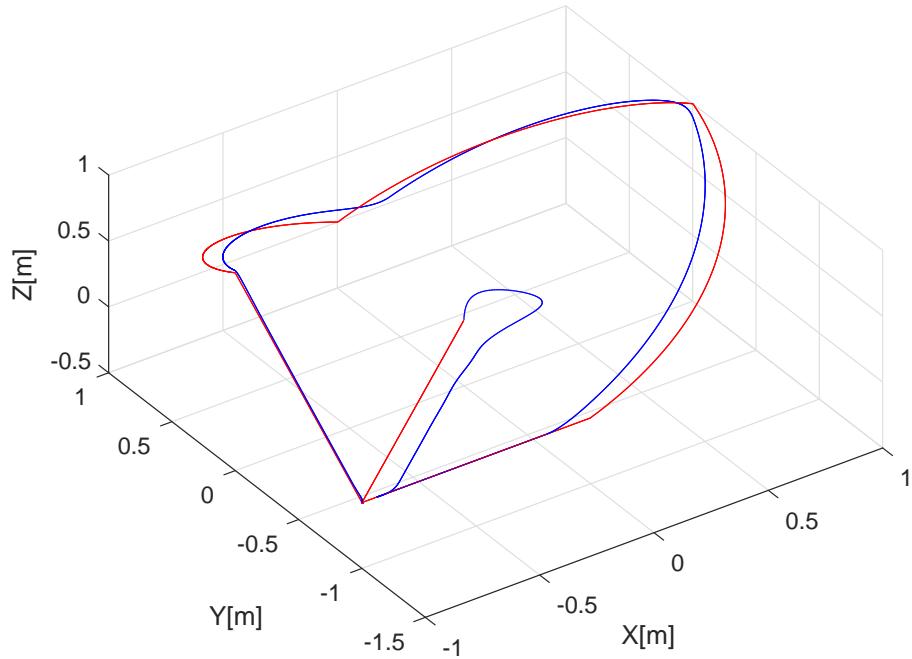
FIGURE 6.22: 3D load position for circle-square trajectory using LQR controller

6.5   Summary

A dynamic game controller was designed for a cable suspended load with two quadrotors to achieve the system stability during the trajectory tracking task in this chapter. Three paths were applied to test the system performance using the game controller and the LQR controller. Both controllers were used to make comparisons in different aspects of the payload transportation task. In comparison, the game controller was introduced to reduce the divergence of the system state relying on the mathematical consistency in a way that it can develop two automatic decision makers. This strategy motivated the need for two quadrotors' behavior choices to be achieved efficiently. Both the leader and the follower quadrotors made their own individual decisions. However, the LQR consisted of one cost function for the dynamic system to achieve stability.

The game performance showed an accurate following behavior for all the reference trajectories in terms of small errors of swing angles and stability achievement due to the flexibility in the dynamic game control parameter, whereas the parameters in the LQR caused the individual decision maker to produce inaccurate system performance and hinder the improvement of the system stability.

Chapter 7

Experimental Evaluation for Path Planning

7.1   Introduction

In the experimental work, the arena robot laboratory was provided with all practical platform requirements in the University of Essex (see Figure 7.1). These requirements are essential to be used in order to perform the tests in a suitable environment. In this chapter, a Vicon tracker system, Hummingbird quadrotors, payload and cables, XBee wireless communication and a computer were used. The capturing system (Vicon) was utilised in order to determine the navigation information for quadrotors and suspended load. The XBee sensor was used to transmit and receive the data collected from the real system to the computer.

As will be explained next, the experimental evaluation conducted in this chapter cannot test the controllers developed in the previous chapters directly, as the Hummingbird quadrotors are not allowed to access the low-level control system. Instead, the controllers developed in the previous chapters will be used as a path planner, which can generate a trajectory with the consideration of the model dynamics and constraints. Then a PD controller is used to control the quadrotors

to track the generated trajectory. In this way, the experiments will provide indirect results to justify the controllers developed in this thesis.

In this chapter, the optimised controllers are used in real experiments as a path planner. This planning procedure is implemented based on the two levels of microcontrollers. The low-level microcontroller, which is responsible for stabilizing the quadrotors and considered as an block box to which the IMU sensor is providing the speed and the angular velocity information. Whereas, the high-level microcontroller is considered to have a validity to be controlled directly using the quadrotor's estimated position information received from the Vicon tracker system. The indirect experiments are executed in this work based on the optimised controllers. These controllers are performed as a planner for the desired path to be followed by the PD controller. Where the system states, including the system speeds and positions, are optimised in MATLAB simulation. Then the optimised data of the predefined trajectory is imported into the C++ code created on the computer to be followed by the PD controller. This path planning method is providing the validity to control the high-level microcontroller through achieving the state optimisation and the system performance. In the real tests, the path planning results of two quadrotors with a suspended load by cables are clarified in this chapter to show the experimental trajectory planning using optimise controllers compared with a direct PD controller. Moreover, this way of path planning gives an efficient method to handle linear controllers and complexities of nonlinear controllers to be implemented, easy to meet the model dynamics of the planned trajectory.

FIGURE 7.1: University of Essex lab

7.1.1   Vicon System

The Vicon system is an optical tracking device which allows the cameras to find and obtain the system altitude, positions and attitudes of moving bodies in three-dimensional space in Figure 7.2. This capturing system employs 24 fixed tracking cameras distributed to cover the arena laboratory used for experimental tests. The information received from the actual system positions and attitudes are calculated by the Vicon capture system at 100 Hz, which is used to solve the tracking control problems in the computer device.

FIGURE 7.2: University of Essex Vicon cameras

7.1.2 UAV Quadrotors (Hummingbird)

Hummingbird unmanned aerial vehicles (UAVs) are commercial German brand aircraft which were designed in Ascending Technology Company and developed to support researchers in different applications (Figure 7.3). For example, these UAVs are equipped with efficient indoor and outdoor sensors, which are presented as processing units, communication units, and inertial measurement units (IMU) [114], [115], [116] and [117]. The processing units are comprised of two levels of microprocessors on board of the quadrotors. The first level is defined by a low-level processor (LLP) which can use the transmitted data from the remote sensor to control the UAV quadrotor's behaviour, while the second level works as a programmable high-level processor (HLP). The high level can be controlled autonomously by using data received by a computer.

FIGURE 7.3: Asc. Tec. Hummingbird quadrotor

The Hummingbird quadrotor has four motors placed at the end of each arm, which generate a vertical thrust and momentum at the centre of gravity for each propeller. The force $\boldsymbol{F}_i$ consists of two complements $\boldsymbol{f}_{iz}$ and $\boldsymbol{f}_{ih}$ as in equation (7.1) and equation (7.2), which are related to the angular speed Ω_i of propeller i:

$$\boldsymbol{F}_i^k = \boldsymbol{f}_{iz}^k + \boldsymbol{f}_{ih}^k, i = 1, 2, 3, 4$$
$$\boldsymbol{f}_{iz}^k = k_F(\boldsymbol{\Omega}_i^{\boldsymbol{k}})^2, i = 1, 2, 3, 4$$

(7.1)

while the momentum equation is presented as

$$\boldsymbol{f}_{ih}^k = k_M(\boldsymbol{\Omega}_i^{\boldsymbol{k}})^2$$
$$\boldsymbol{m}_{ih}^k = \boldsymbol{f}_{ih}^k$$

(7.2)

where both the thrust and momentum of each propeller depend on its angular velocity. The dynamic parameters of the force and momentum are created by the propellers as $k_F \approx 6.11 \times 10^{-10} \frac{N}{rpm^2}$ and $k_M \approx 1.5 \times 10^{-9} \frac{N_m}{rpm^2}$.

The slung load is presented by a point mass toy car and suspended from the top centre by cables and connected to the centre of gravity for each quadrotor. The setup positions and attitudes of single and two quadrotors with suspended payload in the experimental test were provided in Chapter 3.

7.1.3 Communications

The communication between the DELL computer placed in the robotic laboratory and the quadrotors with cable suspended payload was implemented via four serial link XBee modules (Figure 7.4). This standard equipment XBee-Pro 802.15.4OEM wireless link was used in two parts. The first two modules were connected with a computer to transmit the data, while the second two modules were mounted on the quadrotors to receive the data. The rate of transmission for the information between the computer and the actual system is 100 Hz.
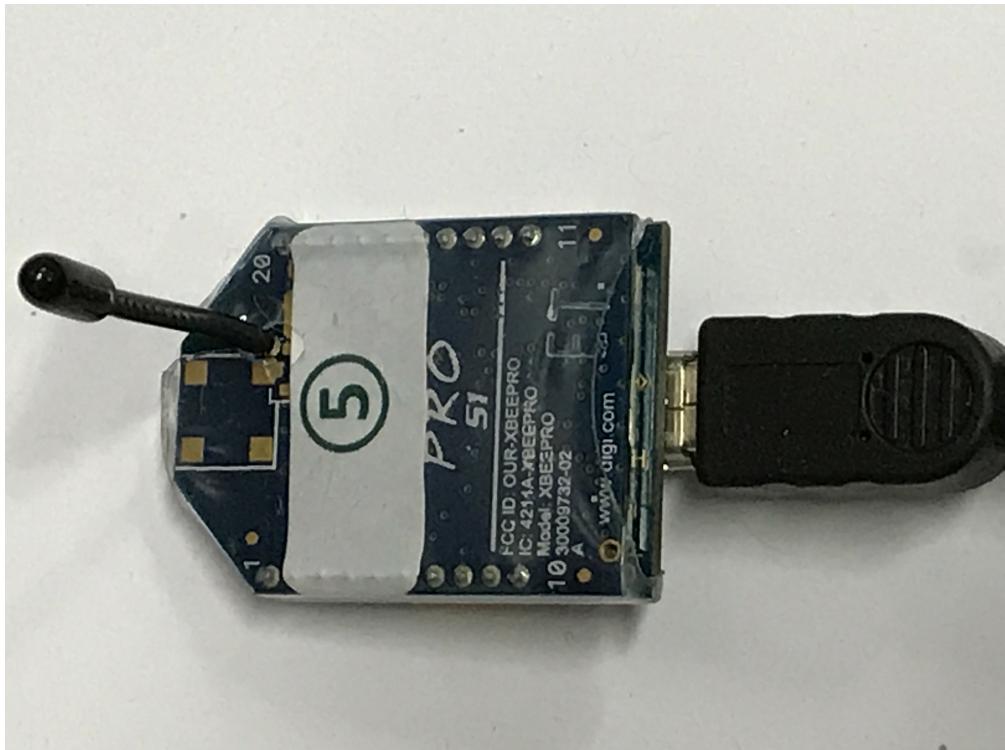
FIGURE 7.4: XBee wireless module

7.2 Path Generation Evaluation

The Hummingbird quadrotors are controlled by two onboard microcontrollers. The low-level microcontroller is used to stabilise the quadrotor. It takes the feedback from onboard IMU sensor and outputs the propeller rotation velocities. The high-level microcontroller is used to deal with the wireless communication model. The low-level microcontroller is not allowed to access. Thus, we cannot use the optimal controllers developed in the previous chapters to directly control the quadrotor.

The optimal controllers developed in the previous chapters are used as a path planner, which generates a trajectory for the quadrotor to track. The generated trajectory meets the model dynamics. Then PD controller is used to control the quadrotor to track the generated trajectory, see Figure 7.5. The experimental results from this setting will be used to evaluate the path generation performance using the optimal controllers. Based on the optimised data of the predefined trajectories obtained by the MATLAB simulation, the resulting tested data was included in the C++ code to be followed by the PD controller. Where these data was presented by the optimised states of both quadrotors obtained based on the planned control inputs in simulation and the states are used as input values in the experimental C++ code.
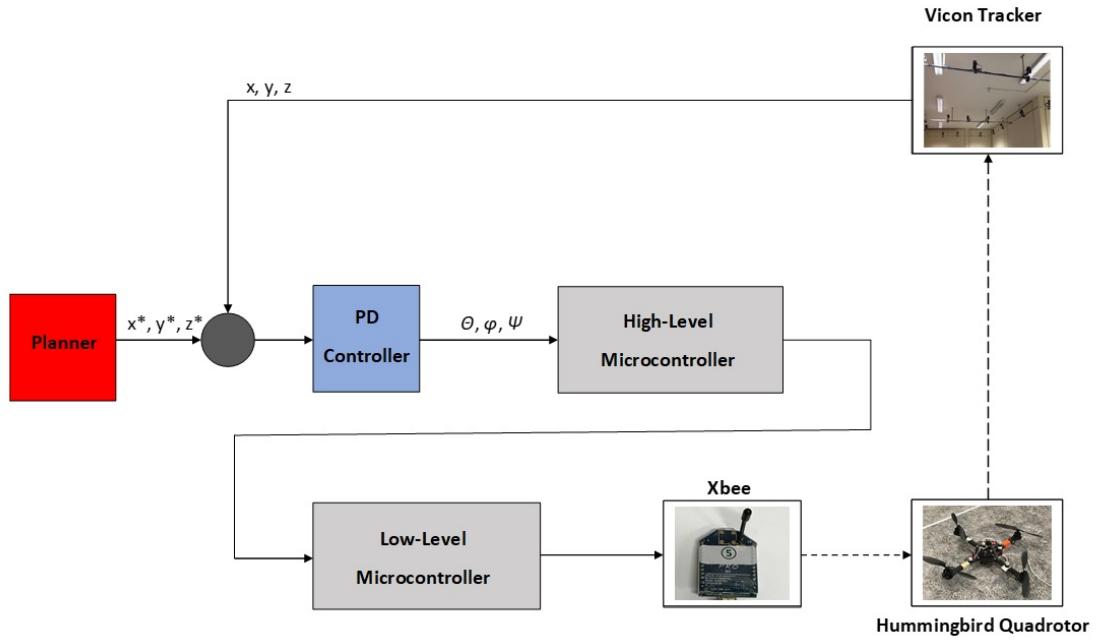
FIGURE 7.5: Experimental evaluation for path planning block diagram

7.3 Control Performance

The experimental tests were executed based on the optimised simulation data. The optimised simulation information was exported then imported into the experiment C++ code to be followed. This practical tracking process was performed with two scenarios. The first scenario was carried out to implement a load position tracking control in different proposed trajectories using a PD controller. The second scenario was represented by path planning data to be followed by using the PD controller. We want to show the control performance improvement when the path generated from the optimal controllers are used, compared with no path plan data is used. The planned paths were proposed to be followed by a cable suspended payload with quadrotors, where these trajectories are: eight, star and spiral trajectories.

The origin point located in the middle of the robotic laboratory arena was considered by all the trajectories when taking off to the altitude $1.2m$. This height

was increased accumulatively depending on the required altitude of the trajectory to be followed. For instance, in the spiral path, the desired hovering altitude was $1.6m$, which meant that before starting the tracking process, the accumulation had to become $1.2m + 0.4m = 1.6m$.

Experimentally, the promising results are illustrated in Figures $7.6 - 7.18$, where the red line (–) is the reference trajectory, while the blue (–) represents the actual testing performance. Figures 7.6, 7.7, 7.8 and 7.9 show the PD and the LQR tracking controllers for two quadrotors with suspended payload respectively, where a eight predefined path was selected to be followed with $30s$ time consuming. Figures 7.6 and 7.7 illustrate in three dimensions the performance of a cable suspended payload with a quadrotor using PD controller directly, while Figures 7.8 and 7.9 present the LQR optimal controller, where the desired path was defined by eight trajectory. Figures 7.10 and 7.11 clarify the system performance tracking eight reference trajectory optimized by ILQR controller, while Figures 7.12 and 7.13 were investigated using game controller. The Figures 7.14, 7.15 and 7.16 demonstrate the system behaviour when following a star predefined path using the LQR, ILQR and the game controllers. By implementing the NMPC controller, the results are illustrated in Figures 7.17 and 7.18 considering constraints in simulation.

The tracking control results of the NMPC controller compared with that of the LMPC are illustrated in Figures 7.17 and 7.18. These figures show the ability of the NMPC controller to handle states and control constraints with limited input external disturbances. The results of the dynamic game controller were compared with those of the LQR controller. In the depicted results, the optimal controllers' accomplishment was good enough to achieve small steady-state errors and stability payload transportation by performing small load swing angles. Moreover, the results show that the MPC controller was able to handle the system state and control input constraints.
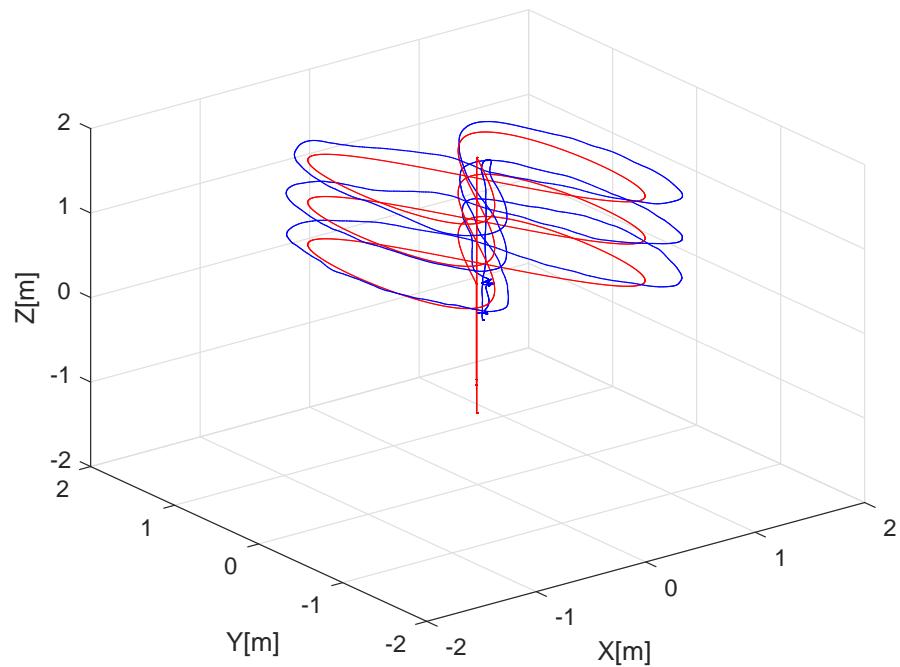
FIGURE 7.6: 3D for the eight trajectory using the PD controller
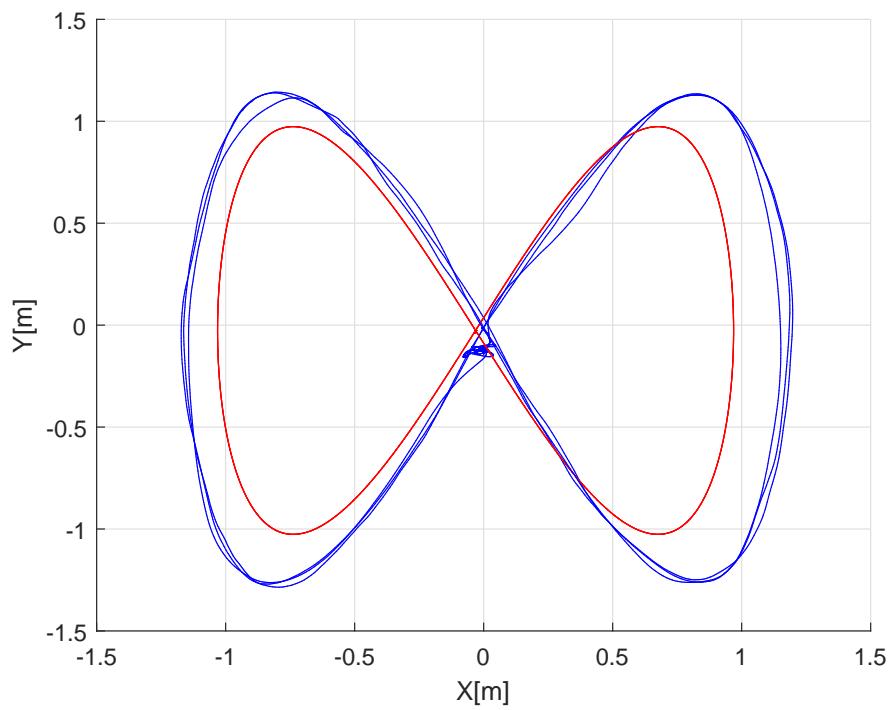
FIGURE 7.7: 2D for the eight trajectory using the PD controller
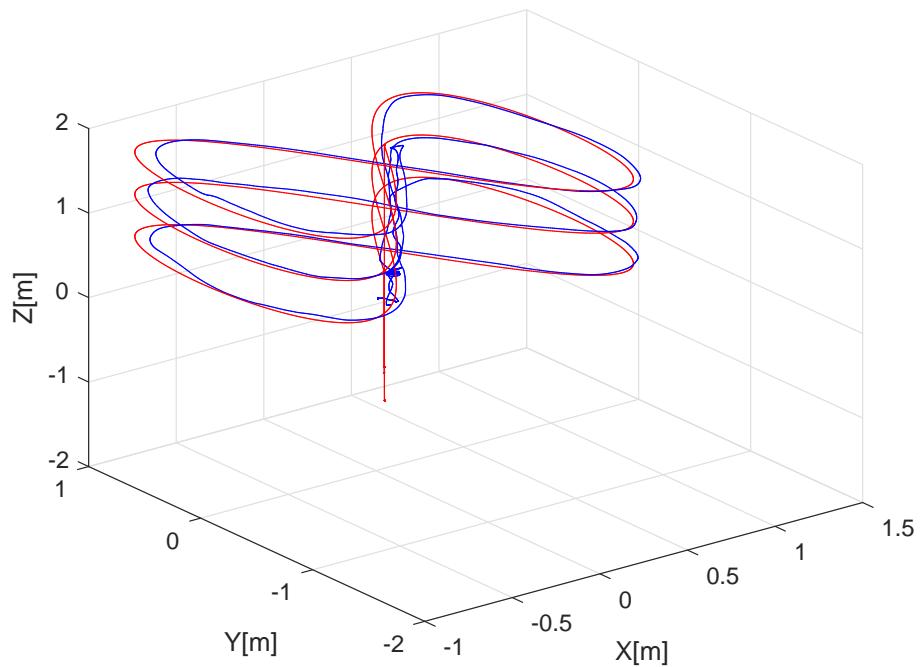
FIGURE 7.8: 3D for the eight trajectory using the LQR controller
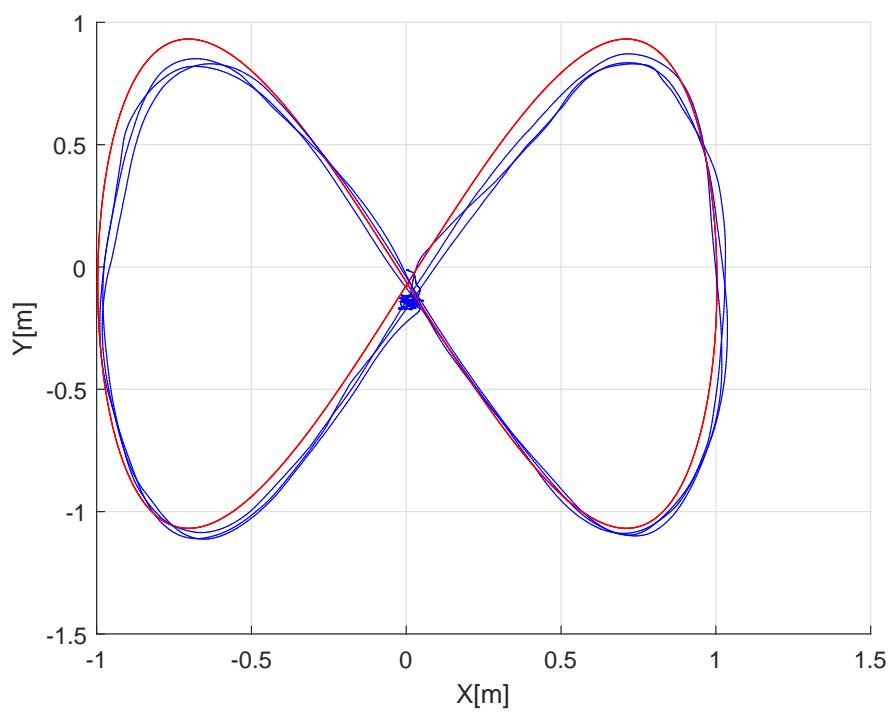
FIGURE 7.9: 2D for the eight trajectory using the LQR controller
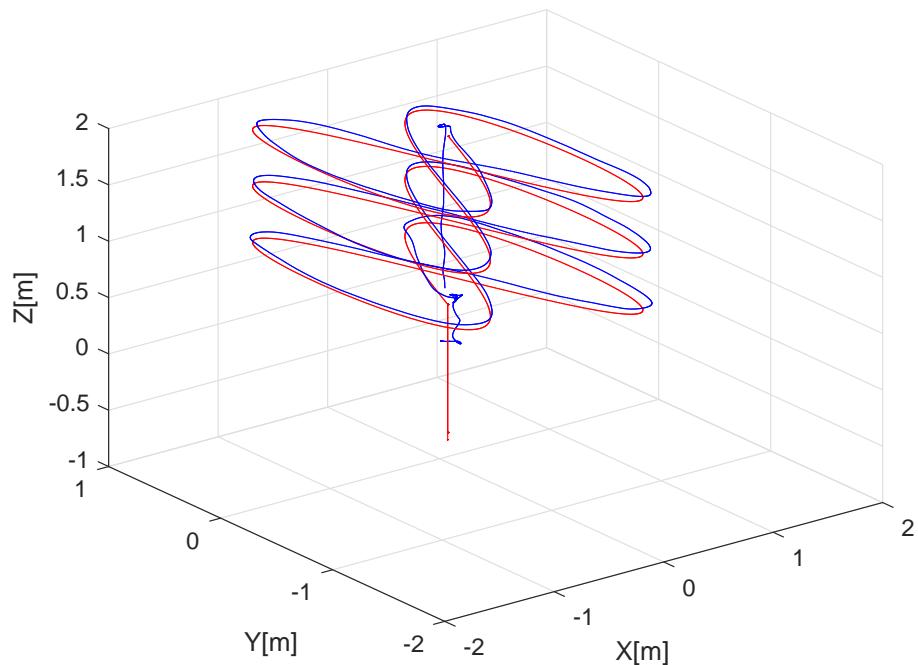
FIGURE 7.10: 3D for the eight trajectory using the ILQR controller
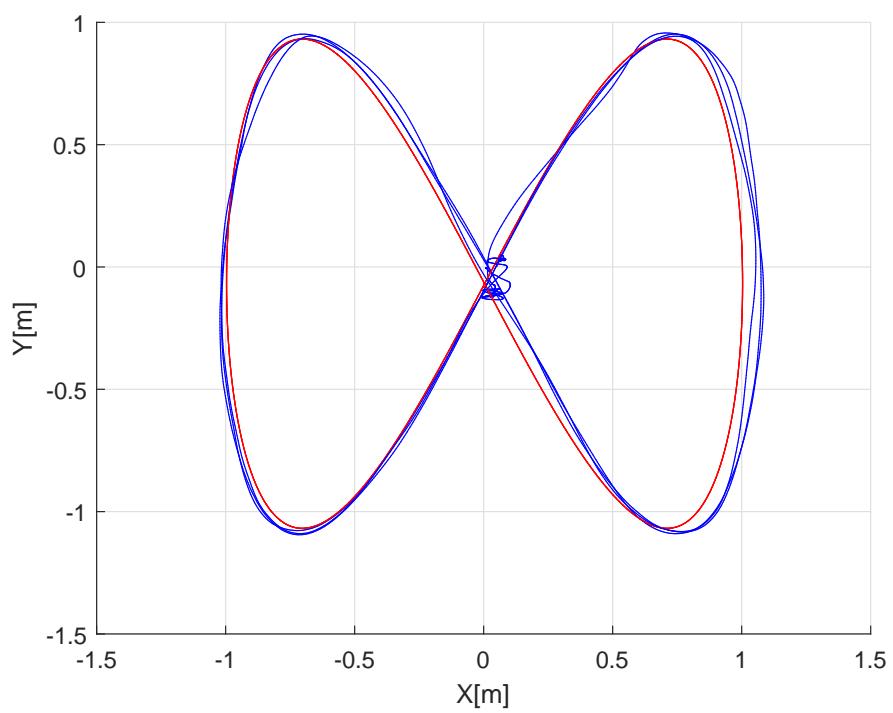
FIGURE 7.11: 2D for the eight trajectory using the ILQR controller
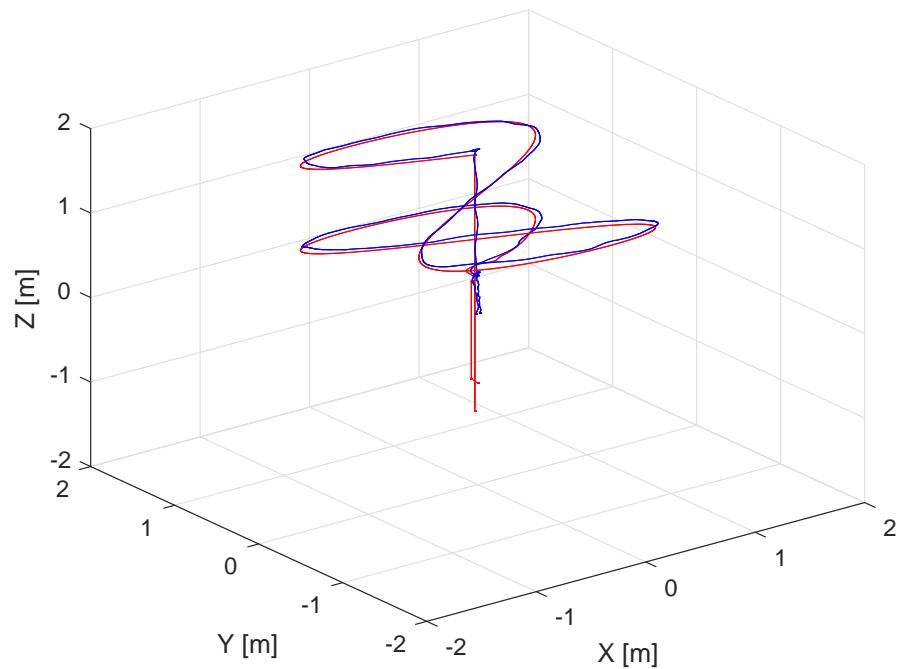
FIGURE 7.12: 3D for the eight trajectory using the dynamic game controller
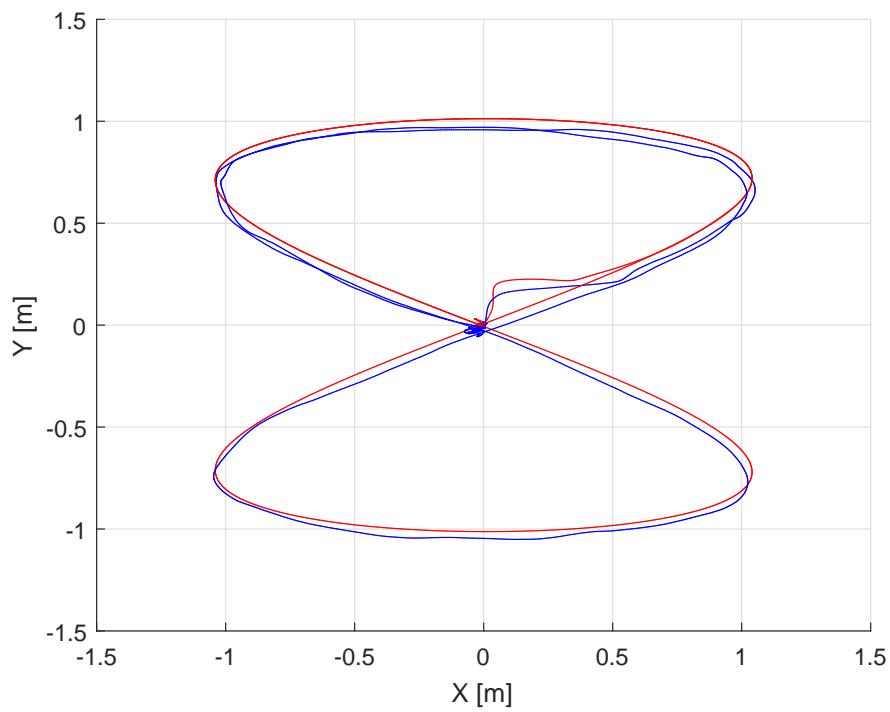
FIGURE 7.13: 2D for the eight trajectory using the dynamic game controller
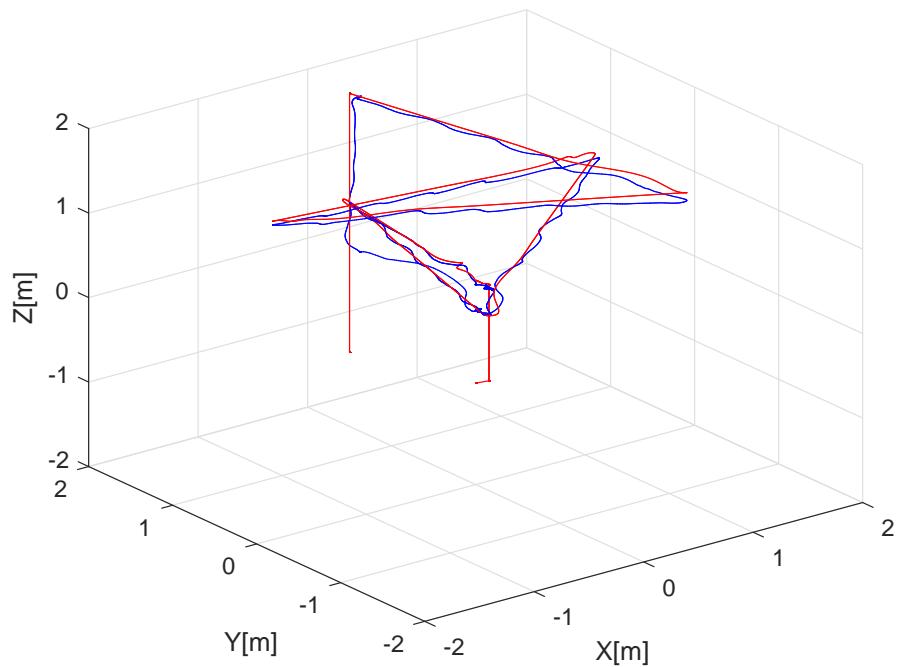
FIGURE 7.14: 3D for the star trajectory using the LQR controller
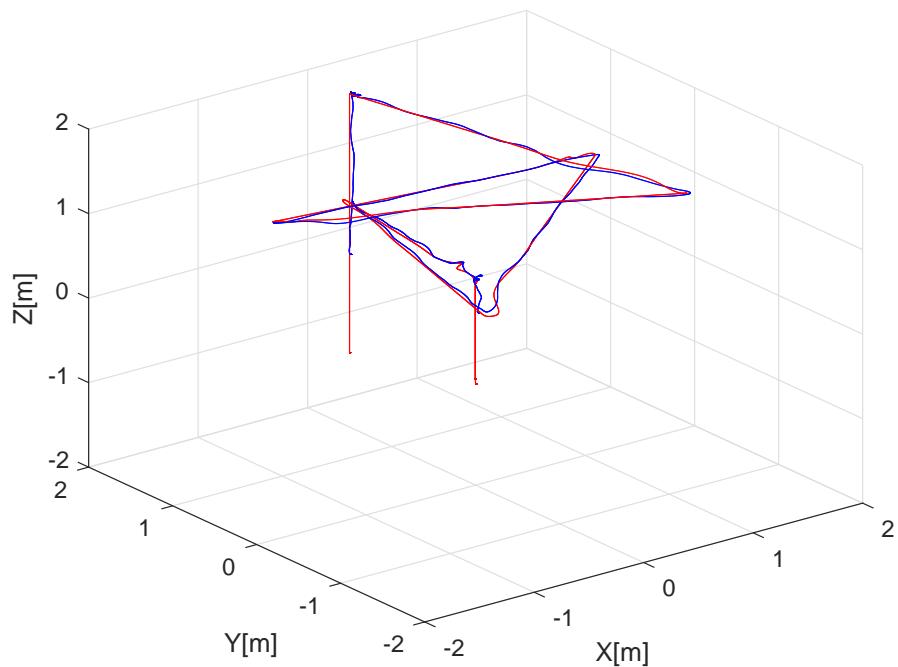
FIGURE 7.15: 3D for the star trajectory using the ILQR controller
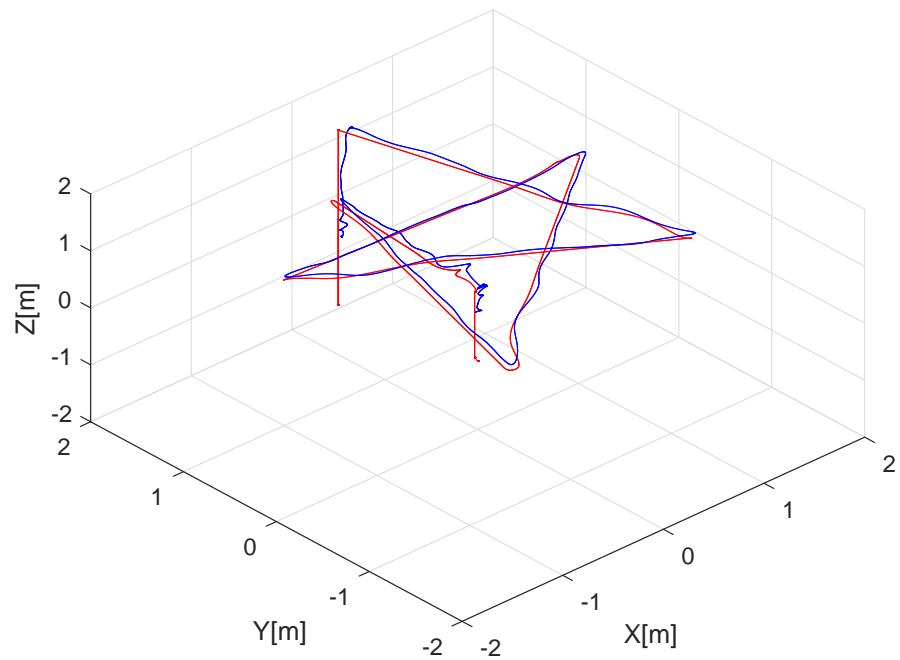
FIGURE 7.16: 3D for the star trajectory using the dynamic game controller
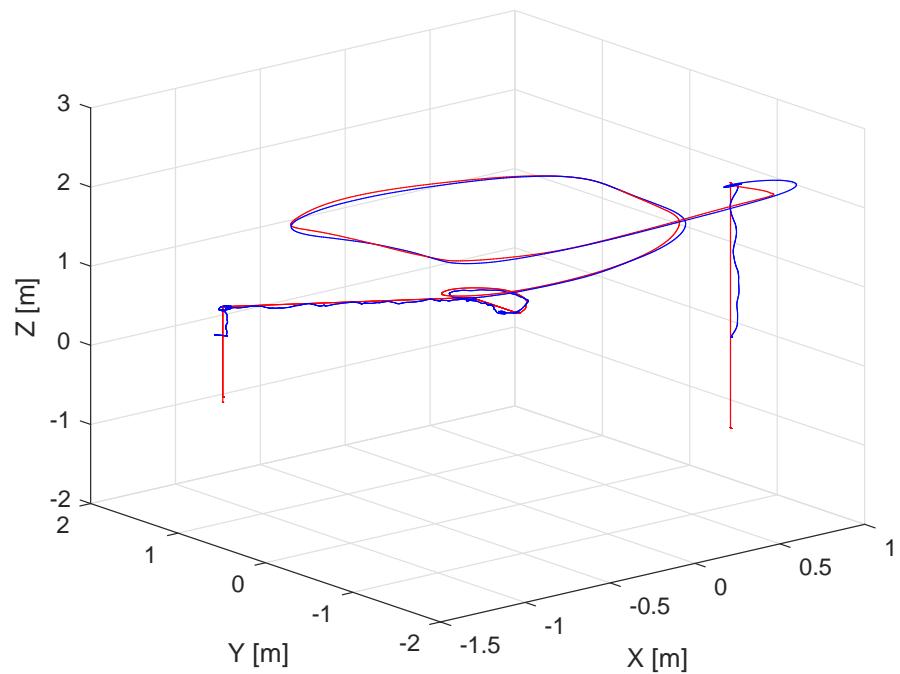
FIGURE 7.17: 3D for the spiral trajectory using the NMPC controller
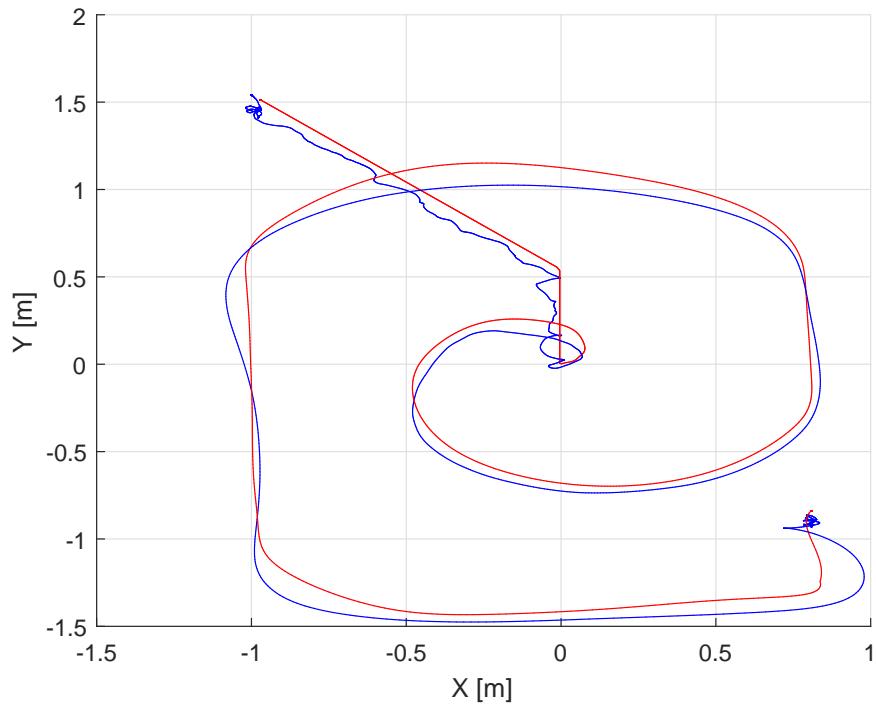
FIGURE 7.18: 2D for the spiral trajectory using the NMPC controllers

7.4 Summary

In this chapter, the optimal controllers were examined on a real Hummingbird quadrotors carrying a payload by cables. But they were not directly used to the quadrotors. Instead, they were used to generate feasible paths for the quadrotors to follow. These paths were previously planned using a simulation tracking test and then the optimised data obtained from the simulation output was imported into the practical code. The classical PD controller was implemented based on C++ code to follow these data. In this experimental flight test, the optimal controllers demonstrated the ability to achieve stability and accuracy in performance, and handle the constraints.

Chapter 8

Conclusions and Future Work

8.1 Conclusions

This research aimed at investigating two main issues in cable suspended payload with multi-quadrotors, namely control of stabilisation and path tracking, through a number of simulations and tests. In particular, the study focused on designing four controllers: LQR, ILQR, MPC, and game controller. The LQR controller was responsible for achieving system stability, while the ILQR's task was to cover the system's nonlinearity while tracking the desired trajectory and consequently reduce the system state errors. The MPC controller was in charge of handling the system state and control constraints. Finally, the game controller was responsible for individual decision making due to considerations of the multiple cost function and for providing flexible behaviour. Each of the above controllers was illustrated by theoretical derivation, stability achievement, simulations results and experimental tests. In addition, all these controllers were verified by comparing their simulation results with each other and with those of the PD controller. Two dynamic models were presented in this research, cable suspended load with single and two quadrotors, while considering dynamic system nonlinearities.

This chapter includes a summary of the results and contributions of the study. It also highlights its limitations and offers suggestions and guidelines for future research.

8.2 Research Summary

LQR Controller The simulation results showed that the performance of the LQR control was generally good. This controller displayed an optimal solution while showing very marginal steady state errors. One significant feature of using the LQR controller lies in the fact that it ensures stability and even automatically guarantees the controller when the two parameters are carefully selected. However, one shortcoming of this controller is that the system becomes linearised under one operating condition, thus posing a problem that needs to be overcome.

ILQR Controller This controller was utilised to address the aforementioned problem, i.e. the linearisation of the system. An ILQR controller is primarily based on an LQR controller with an iteration. In the current study, the ILQR controller outperformed the LQR controller when tracking the Eular Angles, tracking different paths for cable suspended payload with multi-quadrotors and preserving the leader-follower distance with regards to the speed of catching the desired target and reducing steady state errors. These results are due to the linearisation method used for the ILQR controller compared to the single operation point linearisation technique employed in the LQR algorithm. After four iterations, there was evidence of a slight improvement in the performance as a whole.

MPC Controller This controller was applied in this thesis twice, once on a linearised model LMPC and another on a nonlinear model NMPC for tracking cable suspended payload with two quadrotors. Upon considering the constraints on the control input and state vectors, it was found that the LMPC controller outperformed the LQR controller in controlling the system. Applying this controller proved to be of a great benefit for the system since the constraints play an important role in dealing with the limitations of the environment and the power

consumption of the quadrotors. Moreover, utilising the constraints in this controller had the advantage of avoiding the replanning of a new trajectory for the system. The simulation results for this controller also revealed that the NMPC controller performed better than the LMPC when external disturbances were imposed, suggesting that the LMPC struggled in dealing with these disturbance when subjected to the constraints. It is worth noting that this controller adopted a single cost function.

Game Controller, the game controller used a multi-cost function in order to give flexibility to the system for marking decisions in the tracking process. The results of applying this controller to the system were compared to those obtained from the LQR controller. In comparison, the game controller was able to reduce the divergence of the payload position from the reference by relying on mathematical consistency and developing two decision makers. By virtue of this controller, both the leader and the follower quadrotors were able to make individual decisions. In other words, the flexibility which the game controller enjoyed led to small errors in swing angles and consequently improved the follower's tracking accuracy.

To summarise, the LQR controller made significant improvement in terms of system stability in comparison with the PD controller. As for the ILQR, it proved very effective in compensating for the system nonlinearity and increasing stability. The LMPC, however, was able to handle the constraints efficiently, while the LQR was not. Among all the controllers, the NMPC was perhaps the most useful as it was able to handle the disturbances while maintaining constraints in order to test the nonlinearlity in comparison with the LMPC. Finally, the game controller managed to increase the system's flexibility to a significant extent based on the multi-cost function.

8.3   Summary of Research Results

In summary, the performance of the LQR controller compared with the PD controller has been demonstrated in terms of load position RMS error values. These

errors are clarified in Table 4.2, where the minimum LQR payload RMSE values, such as 0.0123(m) and 0.086(m), are verified for the PD controller using a single quadrotor. The ILQR controller shows significant error reduction for the suspended payload position to become 0.0026(m)- 0.0027(m) using single and two quadrotors and is then compared with the LQR controller in Tables 4.3 and 4.4. In the other circumstances, the system constraints validation is accompanied with maintaining small RMSE 0.0012(m) using LMPC, compared with the linear optimized LQR controller in Table 5.1, whereas the nonlinear control action NMPC controller is tested based on handling the system constraints and robustness by applying disturbances to achieve control performance and improve the system stability through reducing RMSE to 0.0012(m) in Table 5.2. In the game controller, the incentive strategy is based on the uncentralized player decisions, which provide flexible information change, and this leads to improving the system stability by dropping the RMSE to become 0.0009(m) in Table 6.1.

8.4 Future Work

Despite the very important findings of this research, represented by the stability and accuracy of the optimal controllers and the robustness of the NMPC and its ability to handle constraints, there are a number of limitations in the current study which could be addressed in future research.

One of these limitations lies in the fact that external disturbances were not included in the control design due to the high non-linearity of the dynamic model. Including these disturbances after designing the controllers would have aggravated the complexity of control development, which was already complicated. One possibility for future research is to incorporate these disturbances in the control design in order to guarantee robust performance of the controller.

Another suggestion for future research is to implement the experimental tests outdoors using GPS or visionary equipment, such as cameras and laser sonars, to avoid limited applicability. In this case, it would be very interesting to practically

apply the optimal controllers on real vehicles to further illustrate the validity of the simulation results.

Moreover, in this thesis, only two quadrotors with a relatively heavy payload were used in the simulations and the experimental tests. Further studies may replicate the experiment with more than two quadrotors and a heavier payload in order to check the consistency of the findings.

Additionally, As an optimal performance with the use of the LQR, ILQR, MPC and Game controllers were found, stability was achieved and handling the high nonlinearity and constraints. The future step toward uncertainty issue is to perform a fuzzy logic controller by gathering with the MPC controller to overcome the system constraints indoor and handling the noise presented in the real world and especially in outdoor environments.

Finally, one area for further study is apply more than one controller in the system and use these in a hybrid way for better stability and accuracy outdoors. In line with this, more nonlinear controllers could be utilised to improve the performance of the system.

Bibliography

[1] B.-C. Min, C. H. Cho, K. M. Choi, and D. Kim, "Development of a micro quad-rotor UAV for monitoring an indoor environment," in *FIRA RoboWorld Congress*, pp. 262–271, Springer, 2009.

[2] T. F. Villa, F. Gonzalez, B. Miljievic, Z. D. Ristovski, and L. Morawska, "An overview of small unmanned aerial vehicles for air quality measurements: Present applications and future prospectives," *Sensors*, vol. 16, no. 7, p. 1072, 2016.

[3] X. Zhou, X. Zhang, J. Zhang, and R. Liu, "Stabilization and trajectory control of a quadrotor with uncertain suspended load," *arXiv preprint arXiv:1612.04324*, 2016.

[4] K. Klausen, "Coordinated control of multirotors for suspended load transportation and fixed-wing net recovery," *PhD. Thesis, Faculty of Information Technology and Electrical Engineering, Norwegian University of Science and Technology*, 2017.

[5] R. J. F. dos Santos, "Load transportation using rotary-wing UAVs," *Aerospace Engineering Master's thesis at Lisboa, portugal*, 2015.

[6] N. A. S. Ortiz, E. Laroche, R. Kiefer, and S. Durand, "Controller tuning strategy for quadrotor MAV carrying a cable-suspended load," *International Micro Air Vehicle Conference and Flight Competition (IMAV)*, 2016.

[7] A. Faust, I. Palunko, P. Cruz, R. Fierro, and L. Tapia, "Aerial suspended cargo delivery through reinforcement learning," *Department of Computer Science, University of New Mexico, Tech. Rep*, 2013.

[8] J. Estevez Sanz, "Quadrator team modeling and control for deformable linear objects (DLO) transportation," *Department of Computer Science and Artificial Intelligence, Universidad del País Vasco, spain*, 2016.

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

[10] W. Jasim and D. Gu, "H infinite for quadrotor attitude stabilization," in *Control (CONTROL), 2014 UKACC International Conference on*, pp. 19–24, IEEE, 2014.

[11] L. R. G. Carrillo, A. Dzul, and R. Lozano, "Hovering quad-rotor control: A comparison of nonlinear controllers using visual feedback," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 48, no. 4, pp. 3159–3170, 2012.

[12] A. Azzam and X. Wang, "Quad rotor 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, IEEE, 2010.

[13] A. Sorensen, "Autonomous control of a miniature quadrotor following fast trajectories," *Control Engineering Master's Thesis, Aalborg University, Denmark*, 2010.

[14] A. Sharma and A. Barve, "Controlling of quad-rotor UAV using PID controller and fuzzy logic controller," *International Journal of Electrical, Electronics and Computer Engineering*, vol. 1, no. 2, pp. 38–41, 2012.

[15] 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, IEEE, 2010.

[16] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Design and control of an indoor micro quadrotor," in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 5, pp. 4393–4398, IEEE, 2004.

[17] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *Proceedings of the 2005 IEEE International Conference on*, pp. 2247–2252, IEEE, 2005.

[18] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *Inteligent robot and systems, IEEE/RSJ international conference on*, pp. 153–158, IEEE, 2007.

[19] I. Palunko, P. Cruz, and R. Fierro, "Agile load transportation: Safe and efficient load manipulation with aerial robots," *Robotics & Automation Magazine, IEEE*, vol. 19, no. 3, pp. 69–79, 2012.

[20] K. Sreenath, N. Michael, and V. Kumar, "Trajectory generation and control of a quadrotor with a cable-suspended load a differentially-flat hybrid system," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 4888–4895, IEEE, 2013.

[21] F. A. Goodarzi, D. Lee, and T. Lee, "Geometric stabilization of a quadrotor UAV with a payload connected by flexible cable," in *American Control Conference (ACC), 2014*, pp. 4925–4930, IEEE, 2014.

[22] A. Faust, I. Palunko, P. Cruz, R. Fierro, and L. Tapia, "Learning swing-free trajectories for UAVs with a suspended load," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 4902–4909, IEEE, 2013.

[23] 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, IEEE, 2012.

[24] S. Sadr, S. A. A. Moosavian, and P. Zarafshan, "Dynamics modeling and control of a quadrotor with swing load," *Journal of Robotics*, vol. 2014, 2014.

[25] P. Cruz and R. Fierro, "Autonomous lift of a cable-suspended load by an unmanned aerial robot," in *Control Applications (CCA), 2014 IEEE Conference on*, pp. 802–807, IEEE, 2014.

[26] C. Raimúndez and J. L. Camaño, "Transporting hanging loads using a scale quad-rotor," in *CONTROLO'2014–Proceedings of the 11th Portuguese Conference on Automatic Control*, pp. 471–482, Springer, 2015.

[27] C. Raimúndez and J. L. Camaño, "Transporting hanging loads using a scale quad-rotor," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pp. 6149–6154, IEEE, 2014.

[28] S. Tang, V. Wüest, and V. Kumar, "Aggressive flight with suspended payloads using vision-based control," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1152–1159, 2018.

[29] P. Kotaru, G. Wu, and K. Sreenath, "Dynamics and control of a quadrotor with a payload suspended through an elastic cable," in *American Control Conference (ACC), 2017*, pp. 3906–3913, IEEE, 2017.

[30] F. A. Goodarzi and T. Lee, "Dynamics and control of quadrotor UAVs transporting a rigid body connected via flexible cables," *arXiv preprint arXiv:1501.04910*, 2015.

[31] T. Lee, "Geometric control of multiple quadrotor UAVs transporting a cable-suspended rigid body," *arXiv preprint arXiv:1403.3684*, 2014.

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

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

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

[35] K. Sreenath and V. Kumar, "Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots," *Proceedings of Robotics: Science and Systems, Berlin, Germany*, June 2013.

[36] P. Kotaru, G. Wu, and K. Sreenath, "Differential-flatness and control of quadrotors with a payload suspended through flexible cables," in *Indian Control Conference (ICC). 2018*, pp. 352–357, IEEE, 2018.

[37] G. Wu and K. Sreenath, "Geometric control of multiple quadrotors transporting a rigid-body load," in *53rd IEEE Conference on Decision and Control*, pp. 6141–6148, IEEE, 2014.

[38] K. A. Ghamry and Y. Zhang, "Fault-tolerant cooperative control of multiple uavs for forest fire detection and tracking mission," in *Control and Fault-Tolerant Systems (SysTol), 2016 3rd Conference on*, pp. 133–138, IEEE, 2016.

[39] I. H. B. Pizetta, A. S. Brandão, and M. Sarcinelli-Filho, "Cooperative quadrotors carrying a suspended load," in *Unmanned Aircraft Systems (ICUAS), 2016 International Conference on*, pp. 1049–1055, IEEE, 2016.

[40] 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, IEEE, 2015.

[41] 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, IEEE, 2014.

[42] W. Li and E. Todorov, "Iterative linear quadratic regulator design for non-linear biological movement systems.," in *ICINCO*, pp. 222–229, 2004.

[43] 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.

[44] J. van den Berg, "Iterated LQR smoothing for locally-optimal feedback control of systems with non-linear dynamics and non-quadratic cost," in *American Control Conference (ACC), 2014*, pp. 1912–1918, IEEE, 2014.

[45] Y. Alothman, W. Jasim, and D. Gu, "Quad-rotor lifting-transporting cable-suspended payloads control," in *Automation and Computing (ICAC), 2015 21st International Conference on*, pp. 1–6, IEEE, 2015.

[46] M. W. Mueller and R. D'Andrea, "Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 45–52, IEEE, 2014.

[47] K. A. 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, IEEE, 2015.

[48] B.-Y. Lee, S.-M. Hong, D.-W. Yoo, H.-I. Lee, G.-H. Moon, and M.-J. Tahk, "Design of a neural network controller for a slung-load system lifted by 1 quad-rotor," *Journal of Automation and Control Engineering Vol*, vol. 3, no. 1, 2015.

[49] S. Tang and V. Kumar, "Mixed integer quadratic program trajectory generation for a quadrotor with a cable-suspended payload," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 2216–2222, IEEE, 2015.

[50] K. Klausen, T. I. Fossen, and T. A. Johansen, "Nonlinear control of a multirotor uav with suspended load," in *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, pp. 176–184, IEEE, 2015.

[51] C. de Crousaz, F. Farshidian, M. Neunert, and J. Buchli, "Unified motion control for dynamic quadrotor maneuvers demonstrated on slung load and rotor failure tasks," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 2223–2229, IEEE, 2015.

[52] C. de Crousaz, F. Farshidian, and J. Buchli, "Aggressive optimal control for agile flight with a slung load," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) Workshop on Machine Learning in Planning and Control of Robot Motion*, 2014.

[53] J. E. Trachte, L. F. Gonzalez Toro, and A. McFadyen, "Multi-rotor with suspended load: System dynamics and control toolbox," in *Aerospace Conference, 2015 IEEE*, pp. 1–9, IEEE, 2015.

[54] F. Gonzalez, A. Heckmann, S. Notter, M. Zürn, J. Trachte, and A. McFadyen, "Non-linear model predictive control for UAVs with slung/swung load," *International Conference on Robotics and Automation, Washington State Convention and Trade Center (WSCC), Seattle, Washington, USA*, 2015.

[55] E. Todorov and W. Li, "A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *American Control Conference, 2005. Proceedings of the 2005*, pp. 300–306, IEEE, 2005.

[56] S. Notter, A. Heckmann, A. Mcfadyen, and F. Gonzalez, "Modelling, simulation and flight test of a model predictive controlled multirotor with heavy slung load," *20th IFAC Symposium on Automatic Control in AerospaceACA 2016*, vol. 49, no. 17, pp. 182–187, 2016.

[57] B. Yu, Y. Zhang, I. Minchala, and Y. Qu, "Fault-tolerant control with linear quadratic and model predictive control techniques against actuator faults in

a quadrotor UAV," in *Control and Fault-Tolerant Systems (SysTol), 2013 Conference on*, pp. 661–666, IEEE, 2013.

[58] M. Iskandarani, S. N. Givigi, C. A. Rabbath, and A. Beaulieu, "Linear model predictive control for the encirclement of a target using a quadrotor aircraft," in *Control & Automation (MED), 2013 21st Mediterranean Conference on*, pp. 1550–1556, IEEE, 2013.

[59] M. Iskandarani, S. N. Givigi, G. Fusina, and A. Beaulieu, "Unmanned aerial vehicle formation flying using linear model predictive control," in *Systems Conference (SysCon), 2014 8th Annual IEEE*, pp. 18–23, IEEE, 2014.

[60] M. Iskandarani, A. T. Hafez, S. N. Givigi, A. Beaulieu, and C. A. Rabbath, "Using multiple quadrotor aircraft and linear model predictive control for the encirclement of a target," in *Systems Conference (SysCon), 2013 IEEE International*, pp. 620–627, IEEE, 2013.

[61] A. S. Imam and R. Bicker, "Quadrotor model predictive flight control system," *International Journal of Current Engineering and Technology*, vol. 4, no. 1, pp. 355–365, IEEE, 2014.

[62] M. Zürn, K. Morton, A. Heckmann, A. McFadyen, S. Notter, and F. Gonzalez, "MPC controlled multirotor with suspended slung load: System architecture and visual load detection," in *Aerospace Conference, 2016 IEEE*, pp. 1–11, IEEE, 2016.

[63] M. W. Mueller and R. D'Andrea, "A model predictive controller for quadrocopter state interception," in *Control Conference (ECC), 2013 European*, pp. 1383–1389, IEEE, 2013.

[64] M. Bangura and R. Mahony, "Real-time model predictive control for quadrotors," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 11773–11780, 2014.

[65] A. Bemporad, C. A. Pascucci, and C. Rocchi, "Hierarchical and hybrid model predictive control of quadcopter air vehicles," *IFAC Proceedings Volumes*, vol. 42, no. 17, pp. 14–19, 2009.

[66] A. Bemporad and C. Rocchi, "Decentralized hybrid model predictive control of a formation of unmanned aerial vehicles," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 11900–11906, 2011.

[67] K. Alexis, G. Nikolakopoulos, and A. Tzes, "Experimental model predictive attitude tracking control of a quadrotor helicopter subject to wind-gusts," in *Control & Automation (MED), 2010 18th Mediterranean Conference on*, pp. 1461–1466, IEEE, 2010.

[68] K. Alexis, C. Papachristos, G. Nikolakopoulos, and A. Tzes, "Model predictive control scheme for the autonomous flight of an unmanned quadrotor," in *Industrial Electronics (ISIE), 2011 IEEE International Symposium on*, pp. 2243–2248, IEEE, 2011.

[69] R. Hedjar, "Robust one-step-ahead model predictive control of VTOL-UAVs," in *Control and Decision Conference (CCDC), 2015 27th Chinese*, pp. 3053–3058, IEEE, 2015.

[70] G. Tartaglione, E. D'Amato, M. Ariola, P. S. Rossi, and T. A. Johansen, "Model predictive control for a multi-body slung-load system," *Robotics and Autonomous Systems*, vol. 92, pp. 1–11, 2017.

[71] K. Alexis, C. Papachristos, G. Nikolakopoulos, and A. Tzes, "Model predictive quadrotor indoor position control," in *Control & Automation (MED), 2011 19th Mediterranean Conference on*, pp. 1247–1252, IEEE, 2011.

[72] J. M. Carson, B. Açikmeşe, R. M. Murray, and D. G. MacMynowski, "A robust model predictive control algorithm with a reactive safety mode," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 13175–13181, 2008.

[73] M. Santos, B. Rego, G. Raffo, and A. Ferramosca, "Suspended load path tracking control strategy using a tilt-rotor UAV," *Journal of Advanced Transportation, 19 October*, 2017.

[74] J. A.-G. G. V. Gabriela M. T. Miranda, Leandro M. F. Machado†, "A multi-core software design of a model predictive control for a tilt-rotor UAV," in

XIII Simposio Brasileiro de Automa¸cao Inteligente Porto Alegre, pp. 2345–2351, SBAI, 2017.

[75] M. Saska, Z. Kasl, and L. Přeucil, "Motion planning and control of formations of micro aerial vehicles," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 1228–1233, 2014.

[76] A. Mancini, A. Benini, E. Frontoni, P. Zingaretti, and S. Longhi, "Coalition formation for unmanned quadrotors," in *Proceedings of the 7th International ASME/IEEE Conference on Mechatronics, Embedded Systems, Applications*, pp. 315–320, 2011.

[77] G. Vasan, A. K. Singh, and M. Krishna, "Model predictive control for micro aerial vehicle systems (MAV) systems," *arXiv preprint arXiv:1412.2356*, 2014.

[78] S. Mao, W. K. Tan, and K. H. Low, "Autonomous formation flight of indoor uavs based on model predictive control," in *AIAA Infotech@ Aerospace*, pp. 0515, 2016.

[79] A. Aswani, P. Bouffard, and C. Tomlin, "Extensions of learning-based model predictive control for real-time application to a quadrotor helicopter," in *American Control Conference (ACC), 2012*, pp. 4661–4666, IEEE, 2012.

[80] M. Kamel, M. Burri, and R. Siegwart, "Linear vs nonlinear MPC for trajectory tracking applied to rotary wing micro aerial vehicles," *arXiv preprint arXiv:1611.09240*, 2016.

[81] C. Liu, W.-H. Chen, and J. Andrews, "Piecewise constant model predictive control for autonomous helicopters," *Robotics and Autonomous Systems*, vol. 59, no. 7, pp. 571–579, 2011.

[82] K. Kunanusont, R. D. Gaina, J. Liu, D. Perez-Liebana, and S. M. Lucas, "The N-tuple bandit evolutionary algorithm for automatic game improvement," *arXiv preprint arXiv:1705.01080*, 2017.

[83] I. Harmati and K. Skrzypczyk, "Robot team coordination for target tracking using fuzzy logic controller in game theoretic framework," *Robotics and Autonomous Systems*, vol. 57, no. 1, pp. 75–86, 2009.

[84] I. Harmati, "Multiple robot coordination for target tracking using semi-cooperative stackelberg equilibrium," in *IEE International Control Conference*, 2006.

[85] Y. Liu, T. Wu, J. Huang, and S. Jia, "A stackelberg-game-based power control algorithm for wireless mesh networks," in *Abstract and Applied Analysis*, Hindawi Publishing Corporation, 2013.

[86] M. Bloem, T. Alpcan, and T. Başar, "A stackelberg game for power control and channel allocation in cognitive radio networks," in *Proceedings of the 2nd international conference on Performance evaluation methodologies and tools*, p. 978-963, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007.

[87] R. Borndörfer, B. Omont, G. Sagnol, and E. Swarat, "A stackelberg game to optimize the distribution of controls in transportation networks," in *International Conference on Game Theory for Networks*, pp. 224–235, Springer, 2012.

[88] M. Somasundaram and R. Sivakumar, "Game theory based security in wireless body area network with stackelberg security equilibrium," *The Scientific World Journal*, vol. 2015, 2015.

[89] P. Paruchuri, J. P. Pearce, J. Marecki, M. Tambe, F. Ordonez, and S. Kraus, "Playing games for security: An efficient exact algorithm for solving bayesian stackelberg games," in *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pp. 895–902, 2008.

[90] W. Tushar, W. Saad, H. V. Poor, and D. B. Smith, "Economics of electric vehicle charging: A game theoretic approach," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1767–1778, 2012.

[91] C. Dextreit and I. V. Kolmanovsky, "Game theory controller for hybrid electric vehicles," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 2, pp. 652–663, 2014.

[92] K. Schüller, K. Staňková, and F. Thuijsman, "Game theory of pollution: National policies and their international effects," *Games*, vol. 8, no. 3, p. 30, 2017.

[93] R. Vidal, O. Shakernia, H. J. Kim, D. H. Shim, and S. Sastry, "Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation," *IEEE transactions on robotics and automation*, vol. 18, no. 5, pp. 662–669, 2002.

[94] K. Skrzypczyk, "Game theory based task planning in multi robot systems," *International Journal of Simulation*, vol. 6, no. 6, pp. 50–60, 2005.

[95] W. Inujima, K. Nakano, and S. Hosokawa, "Multi-robot coordination using switching of methods for deriving equilibrium in game theory," in *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2013 10th International Conference on*, pp. 1–6, IEEE, 2013.

[96] H. Zhang, M. Bennis, L. A. DaSilva, and Z. Han, "Multi-leader multi-follower stackelberg game among wi-fi, small cell and macrocell networks," in *Global Communications Conference (GLOBECOM), 2014 IEEE*, pp. 4520–4524, IEEE, 2014.

[97] C. U. Solis, A. S. Poznyak, and J. B. Clempner, "Solving stackelberg security games for multiple defenders and multiple attackers," *Department of Control Automatics, Mexico*, 2015.

[98] K. Stanková, B. Ranjbar-Sahraei, G. Weiss, and K. Tuyls, "Staco: Stackelberg-based coverage approach in robotic swarms," in *The Fifth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE)*, pp. 71–76, 2013.

[99] A. Wilczynski, A. Jakóbik, and J. Kolodziej, "Stackelberg security games: Models, applications and computational aspects," *Journal of Telecommunications and Information Technology*, no. 3, p. 70–79, 2016.

[100] L. Banjanovic-Mehmedovic, E. Halilovic, I. Bosankic, M. Kantardzic, and S. Kasapovic, "Autonomous vehicle-to-vehicle (V2V) decision making in roundabout using game theory," *International journal of advanced computer science and applications*, vol. 7, no. 8, pp. 292–298, 2016.

[101] Y. Meng, "Multi-robot searching using game-theory based approach," *International Journal of Advanced Robotic Systems*, vol. 5, no. 4, p. 341-350, 2008.

[102] Y. Meng and K. Cao, "Game-theory based multi-robot searching approach," *Department of Electrical and Computer Engineering Stevens Institute of Technology, Hoboken, New Jersey, USA*, 2008.

[103] X. Na and D. J. Cole, "Game theoretic modelling of a human driver's steering interaction with vehicle active steering collision avoidance system," *Department of Engineering, University of Cambridge, IEEE*, 2014.

[104] M. Li, J. Cruz, and M. A. Simaan, "An approach to discrete-time incentive feedback stackelberg games," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 32, no. 4, pp. 472–481, 2002.

[105] S. Kelouwani, "Two agent paths planning collaboration based on the state feedback stackelberg dynamic game," *Open Journal of Optimization*, vol. 2, no. 3, pp. 61-71, 2013.

[106] M. Guo, Y. Su, and D. Gu, "Mixed $H2/Hinfinity$ tracking control with constraints for single quadcopter carrying a cable-suspended payload," *IFAC conference, Toulouse France*, vol. 50, no. 1, pp. 4869–4874, 2017.

[107] S. A. Frank, "D'alembert's direct and inertial forces acting on populations: The price equation and the fundamental theorem of natural selection," *Entropy*, vol. 17, no. 10, pp. 7087–7100, 2015.

[108] V. I. Arnol'd, *Mathematical methods of classical mechanics*, vol. 60. Springer Science & Business Media, 2013.

[109] 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, IEEE, 2004.

[110] J. M. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

[111] M. Mejari, A. Gupta, N. M. Singh, and F. Kazi, "Trajectory tracking of quadrotor with bounded thrust using model predictive control," in *Proceedings of Conference on Advances In Robotics*, pp. 1–6, ACM, 2013.

[112] R. L. Fox, " Optimization methods for engineering design," Addison-Wesley Pub. Co., 1971.

[113] J. Lofberg, "YALMIP: A toolbox for modeling and optimization in matlab," in *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pp. 284–289, IEEE, 2004.

[114] M. Jabeur, A. McFadyen, K. Morton, and F. Gonzalez, "Visual servoing of a quadrotor with suspended slung load for object detection and tracking," in *Aerospace Conference, 2017 IEEE*, pp. 1–13, IEEE, 2017.

[115] L. Rosa, "Elective in Robotics, AscTec Hummingbird Quadrotor," *in Sapienza University Di Roma*, 2011.

[116] M. Guo, Y. Su, and D. Gu, "System identification of the quadrotor with inner loop stabilisation system," *International Journal of Modelling, Identification and Control*, vol. 28, no. 3, pp. 245–255, 2017.

[117] A. T. GmbH, "Asctec hummingbird with autopilot user's manual," *Ascending Technologies GmbH*, 2011.