

UNIVERSITY OF APPLIED SCIENCES
Ravensburg-Weingarten

MASTER THESIS ON

Development of the Control Algorithms for Autonomous Landing of Unmanned Aerial Vehicles

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE
DEGREE OF
Master of Science in Mechatronics

Behnam Nourghassemi

SUPERVISORS:

Prof. Dr.-Ing. Holger Voos
Prof. Dr.-Ing. Ralf Stetter

WINTER 2009

THIS PAGE INTENTIONALLY LEFT BLANK.

Acknowledgment

Over the past two years I have had the pleasure and privilege of learning from, and working on various projects under supervision of Prof. Holger Voos during my graduate studies at the University of Applied Sciences Ravensburg-Weingarten. I would like to thank him for all his support and guidance, and for instilling in me a genuine passion for mobile robotics.

I would like also to thank my graduate study professors and advisers, Profs. Ralf Stetter, and Bernd Altmann. Thank you for all those in-class and after-class hours you took for me to explain and solve my problems.

Thanks are also due to Achim Feucht, Philipp Ertle, and all other members of the mobile robotics lab for their contribution and key suggestions during our regular thesis progress report meetings.

Last but not least, I would like also to thank my family for the patience and the support in all my choices.

THIS PAGE INTENTIONALLY LEFT BLANK.

Preface

In this thesis work, control algorithms for navigation and autonomous landing of a Quadrotor on an onboard mobile helipad were developed. The main activities could be divided into three groups. First and main activity was the development of the overall control algorithm. For that reason, an autopilot based on the feedback linearization method was implemented for stabilization and attitude control of the available model of the Quadrotor. Command to line-of-sight (CLOS) system was also implemented for navigation purpose and a behavior-based control system consists of three states of cruise, align, and land mode was proposed for accomplishment of the overall landing task. The next activity was to study the measurability of the proposed control variables with regard to the available sensor technology to keep the work realizable. For that reason few concepts for pose detection of the Quadrotor relative to the mobile helipad were proposed. The last activity was evaluation of the proposed control algorithms by simulation of the overall system in Matlab-Simulink. This activities were conducted from October 2008 to March 2009.

The project was carried out by myself. Most of the studies were performed in the Mobile Robotics Lab, University of Applied Sciences Ravensburg-Weingarten, Weingarten, Germany. Many thanks to Prof. Dr.-Ing. Holger Voos (my thesis supervisor from Electrical Engineering Department) and Prof. Dr.-Ing. Ralf Stetter (my thesis co-supervisor from Mechanical Engineering Department) for their contribution.

Weingarten, March 2009
Behnam Nourghassemi

THIS PAGE INTENTIONALLY LEFT BLANK.

To my parents, who have always supported me.

THIS PAGE INTENTIONALLY LEFT BLANK.

Contents

Acknowledgment	ii
Preface	iv
List of Figures	xiv
List of Tables	xvi
1 Introduction	1
2 State of Art	4
2.1 Study on the Quadrotor Dynamic and Controlling	4
2.2 Autonomous Navigation and Landing of UAV	5
2.3 Study of the Wind Effect	6
2.4 Sensors	6
3 Quadrotor Model	7
3.1 Quadrotor Structure	7
3.2 Mathematical Model	8
4 Attitude Stabilization	13
4.1 Attitude Control System	13
4.2 Linear Velocity Control System	17
5 Command to Line-Of-Sight Navigation	21
5.1 CLOS System	21
5.2 Adaption of the CLOS System for Navigation Control Design	27
5.3 Adjusting k ₁ , k ₂ Based on CLOS System	30
5.4 Special Case Study	33
6 Altitude Controller	35
7 Onboard Mobile Helipad Model	37

8 Cruise, Align, and Land Control	39
8.1 Behavior-Based Finite State Model for Control Task Coordination	39
9 Sensor Fusion	47
9.1 Introduction	47
9.2 Control Parameters in CLOS-Based Navigation	48
9.3 Attitude Sensor	50
9.4 Altitude Measurement	50
9.5 Differential GPS	51
9.6 On-board Camera	54
9.7 Time Of Flight Camera	54
9.8 Intensity Sensor	54
9.9 Ultrasonic Based Localization System	55
10 Modeling of the Wind as a Disturbance to the System	58
11 Simulation Model	61
11.1 Overall System Model	61
11.2 Quadrotor Model	62
11.3 Autopilot Model	63
11.4 CLOS Navigation Control System Model	63
11.5 Altitude Control Model	64
11.6 On-board Mobile Helipad Model	64
11.7 Wind Model	65
11.8 Range and Altitude Sensor Model	65
11.9 Land Terminator Block	65
12 Simulation Results	67
12.1 Experiment 1: High Rate of Changes in Path Direction of Helipad	67
12.2 Experiment 2: Reduced Rate of Changes in Path Direction of Helipad	72
12.3 Experiment 3: Stationary Helipad	76
13 Summary and Future Works	80
A S-Function for Quadrotor	82
B M-File for Autopilot	85
C M-File for Tracking Model	88
D M-File for Altitude Controller	90

E S-Function for On-board Mobile Helipad	92
F S-Function for Wind Model	94
G M-File for Range Sensor	96
Bibliography	97

THIS PAGE INTENTIONALLY LEFT BLANK.

List of Figures

3.1	Free body diagram of a Quadrotor structure.	8
3.2	Attitude control of a Quadrotor; (a) roll angle, (b) pitch angle, (c) yaw angle	9
4.1	Schematic diagram of the attitude (inner) control loop.	17
4.2	Outer linear velocity control loop; the inner attitude control loop assumed to be quick enough and neglected.	18
4.3	The schematic diagram of the overall control loop.	19
5.1	CLOS system for a two-dimensional scenario; $R = \sqrt{\Delta x^2 + \Delta y^2}$ is the planer distance between Quadrotor and the elevated center of the helipad, mapped on the xy-plane containing Quadrotor; $Range = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$ is the distance between Quadrotor and center of the helipad.	22
5.2	Engagement geometry of Quadrotor and helipad.	22
5.3	Normal and tangential acceleration of the Quadrotor.	23
5.4	Controlling Quadrotor motion path by adjusting α_Q	24
5.5	Coordination of the Quadrotor and helipad within Δt seconds of navigation.	24
5.6	Collision path in CLOS system.	26
8.1	3D imaginary state transition geometric shape on the helipad for behavior-based navigation control system transition.	40
8.2	Relation between range and altitude inside the imaginary 3D geometric shape.	41
8.3	Location of the Quadrotor relative to the helipad depends on R and h.	42
8.4	State transition diagram for landing task.	43
8.5	Behavior-based cruise control system.	44
8.6	Behavior-based align control system.	45
8.7	Behavior-based land control system.	46
9.1	Quadrotor and helipad in reference coordinate system.	48

9.2	Relative position of the Quadrotor and helipad; assuming the helipad on the center and Quadrotor on the circumference of the circle	49
9.3	MTi in <i>NE</i> Reference Coordinate System (Adapted from [23]).	50
9.4	The Navstar Global Positioning System consists of three fundamental segments: Space, Control, and User. (Adapted from [38].)	52
9.5	A localization system based on ultrasonic signal, adapted from GPS system.	56
9.6	Measurement of pseudoranges; (a) radio signals and all ultrasonic signals with different frequencies transmitted simultaneously form helipad, (b) on Quadrotor, radio signal will received simultaneously, while ultrasonic signals will be received with some delay proportional to the distance (pseudoranges)	56
10.1	Representation of the wind in the local North-East coordinate system.	59
11.1	The model of the overall scenario in Simulink	62
11.2	Land terminator block in Simulink	66
12.1	Performance of the navigation control system for a big rate of changes in path direction, <i>frequency</i> = $2Hz$; (a) top view of scenario during simulation time of $25s$, (b) magnification of the last few seconds of path tracking	69
12.2	Performance of the behavior-based navigation control system in align and land mode for a big rate of changes in path direction, <i>frequency</i> = $2Hz$; (a) altitude of the Quadrotor during $25s$ of simulation, (b) planer range between Quadrotor and helipad during $25s$ of simulation, (c) altitude fluctuation caused by alternative transitions of the control system to align mode and back to cruise mode, (d) alternative entering and exiting of Quadrotor inside and outside of the alignment geometric shapes, indicated by the green dotted boxes	70
12.3	Attitude controller performance for a big rate of changes in path direction, (a) roll angle control performance, (b) roll angle error, (c) magnification of the roll angle control, (d) magnification of the roll angle error	71
12.4	Attitude controller performance for a big rate of changes in path direction, (a) pitch angle control performance, (b) pitch angle error, (c) magnification of the pitch angle control, (d) magnification of the pitch angle error	72

12.5 Performance of the navigation control system for a reduced rate of changes in path direction, <i>frequency</i> = 1Hz; (a) top view of scenario during simulation time, (b) magnification of the last few seconds of path tracking	74
12.6 Performance of the behavior-based navigation control system in align and land modes for a reduced rate of changes in path direction, <i>frequency</i> = 1Hz; (a) altitude of the Quadrotor during simulation time, (b) planer range between Quadrotor and helipad during simulation time, (c) control transition to land mode	75
12.7 Attitude control system performance for a reduced rate of changes in path direction, (a) roll angle control system, (b) roll angle error.	76
12.8 Attitude control system performance for a reduced rate of changes in path direction, (a) pitch angle control system, (b) pitch angle error.	76
12.9 Performance of the controller in cruise mode for a stationary helipad; (a) top view of path tracking diagram within 25s, (b) magnification of the last few seconds of path tracking	78
12.10 Performance of the navigation control system in align and land mode for a stationary helipad; (a) altitude of the Quadrotor during simulation time, (b) planer range between Quadrotor and helipad during simulation time.	79

THIS PAGE INTENTIONALLY LEFT BLANK.

List of Tables

3.1	Definition of the parameters used in the Quadrotor model . . .	10
9.1	Potential error sources for measured pseudoranges (Adapted from [38])	53

THIS PAGE INTENTIONALLY LEFT BLANK.

Chapter 1

Introduction

This thesis work focuses on the study and development of control algorithms for navigation and autonomous landing of a Quadrotor, on a onboard mobile helipad.

Quadrotor, as an Unmanned Aerial Vehicle (UAV), is a rotorcraft equipped with four powered rotors laid up symmetrically around its center and it could be classified as a Vertical Take-Off and Landing (VTOL) aircraft. Because of its unique characteristics, such as low dimension, good maneuverability, simple mechanics, payload capability, and vertical, stationary and low speed flight, this structure recently gathered popularity among several applications, in particular for surveillance, imaging, dangerous environments, indoor navigation, mapping, pipeline inspection, football stadium security, etc.

As main drawback, the high energy consumption of the Quadrotor can be mentioned and because of the technical insufficiencies in manufacturing of high durable batteries, the operational time of Quadrotors restricted to barely more than an hour. In addition because of its payload limitations, it is impossible to mount spare power supplies on the Quadrotor.

Therefore, in order to realize the unique abilities of the Quadrotor empirically, regular recharging of its power supplies is necessary. Furthermore, in order to realize autonomy, during any predefined mission, Quadrotor could be able to fly from any initial position, within its mission field, back to the recharge station and be able to land autonomously on the helipad. In addition, to increase the efficiency of the system, recharge station should be mobile and approaches itself as close as possible to the mission field in an attempt to reduce the flight distance and time of the Quadrotor for recharging purpose.

The main focus in this thesis work is to develop appropriate path planning and control algorithms for the Quadrotor in order to accomplish the task of tracking and landing on a onboard mobile helipad autonomously, with the wind effect, as the main source of disturbance for the overall system, in the corner of attention. For that reason, a Simulink model, consisting several subsystems, has been developed. According to the goals of this project, the research has been very detailed in both control algorithms design and simulation.

To improve this project, several modifications can be done. For example, by including the aerodynamic and friction effects, a more realistic model of Quadrotor can be replaced for more accurate simulation results. As another example, the wind effect can be studied and modeled in more details.

Furthermore, although in development of control algorithms, special attention has been payed for measurability of the proposed system variables with respect to the available sensor technology, and although the proposed control algorithms have been shown to be efficient in simulation results, however, empirical realization of these proposed algorithms requires separate research.

This thesis is structured as follows:

Chapter 2 gives an overview of the state of the art of the research area. Other related works are cited to show what has already been done in this field.

Chapter 3 represents a simplified Quadrotor model in state space model with twelve states, consists of Quadrotor position, (x, y, z) , translational speed, $\dot{x}, \dot{y}, \dot{z}$, attitude, (ϕ, θ, ψ) , and angular speed, $(\dot{\phi}, \dot{\theta}, \dot{\psi})$. Derivation of the differential equations representing the underlying dynamic of the Quadrotor are described briefly.

Chapter 4 focuses on a step by step autopilot design to stabilize the Quadrotor. For that reason, feedback linearization technique is applied.

Chapter 5 provides a path tracking control algorithm for a mobile target based on command to line-of-sight (CLOS) method. Control algorithm design is discussed in details. Particular attention is given to special case of stationary target.

Chapter 6 provides a simple altitude controller. A simple proportional controller is used for that purpose.

Chapter 7 provides a model for onboard mobile helipad. A sinusoidal motion path with adjustable amplitude, frequency, and speed is assumed and represented with differential equations. The height of the helipad in the reference coordinate system is chosen to be constant with some disturbances due to vibration of the onboard mobile helipad during its motion.

Chapter 8 provides a behavior-based control algorithm for the Quadrotor to cruise toward, align with, and land on the onboard mobile helipad.

Chapter 9 briefly discusses about measuring concepts and provides some suggestions for sensor choice to realize control algorithm empirically. Where it was possible, relative works and literature about measurement of variables of interest for this thesis work are also introduced.

Chapter 10 provides a brief model for the wind as the main disturbance source for the system. The derived equations directly inserted in the Quadrotor's differential equations.

Chapter 11 shows a step by step procedure to develop a simulator for evaluation of the overall system performance. For that reason, Matlab-Simulink program is used and development of the subsystems and blocks are described briefly.

Chapter 12 provides the simulation results to evaluate different aspects of the proposed control algorithms performance.

Chapter 13 summarizes the goals of this thesis, evaluates the performance and the results of the project and proposes solutions to improve the work in future.

Appendices A, B, C, D, E, F, and G provides the source codes of the Simulink models.

Chapter 2

State of Art

In the last few years, the state of the art in Vertical Take-Off and Landing (VTOL) Unmanned Aerial Vehicle (UAV), and especially the Quadrotor, has received several contributes. This chapter will covers some of the available reports and articles from different prospectives relative to this thesis work.

2.1 Study on the Quadrotor Dynamic and Controlling

Various aspects of the Quadrotor as UAV, have been studied extensively within a X4-Flyer Testbed called STARMAC. Hoffmann and *et al.* in [33] have been studied the aerodynamic effects pertaining to the Quadrotor flight within STARMAC. Waslander and *et al.* in [24] have been compared integral sliding mode and reinforcement learning control design techniques for outdoor altitude control of STARMAC. Hoffmann and *et al.* in [13] have been proposed a distributed algorithm for multi-vehicle coordinated localization of a stationary target.

Beyond STARMAC, there are other related works on the Quadrootrs; Voos in [29], [27], and [28] has been developed nonlinear control system for Quadrotor, based on state-dependent Riccati equations (SDRE) of UAV, a combination of state-dependent Riccati equations (SDRE) and neural networks, and decomposition of the model into a nested structure and feedback linearization, respectively.

Moreover, Tayebi and *et al.* in [5] have been proposed a quaternion based PD2 feedback control scheme for exponential attitude stabilization of a Quadrotor. Mokhtari and *et al.* in [36] have been designed a mixed robust feedback linearization with linear $GH\infty$ controller for Quadrotor. Bouabdallah and *et al.* in [31] have been developed a PID for a simplified model of Quadrotor and LQ for a more complete one. Chen and *et al.* in [3] have been designed a $H\infty$ loop shaping flight controller for position control of a Quadrotor. Bouabdallah and *et al.* in [11] have been presented the me-

chanical design, dynamic modeling, sensing, and control of an indoor VTOL autonomous robot.

In addition, there are articles about the realization of more realistic prototypes of Quadrotor; Pounds and *et al.* in [19] and [41] have been developed a highly reliable experimental platform of a X-4 Flyer Quadrotor using custom-built chassis and avionics with off-the-shelf motors and batteries.

2.2 Autonomous Navigation and Landing of UAV

There are several methods which proposed for navigation and autonomous landing of UAV on a landing target. However, most of the attention has been focused on vision based autonomous landing of UAV; Shakernia and *et al.* in [15] have been used a computer vision as a feedback sensor in the control loop to extract the position and velocity of the UAV relative to the landing pad based on the discrete and differential versions of the *ego-motion estimation* together. Saripalli and *et al.* in [47, 46, 1] have been used a combination of vision and GPS for navigation of an autonomous helicopter. In addition Bosse and *et al.* in [9] have been used an onboard camera as an additional navigation sensor for an autonomous helicopter. Merz and *et al.* in [7] have been only used an on-board visual navigation system and inertial sensors, without any GPS, for an autonomous precision landing of an unmanned helicopter. Sharp and *et al.* in [10] have been introduced the design of a landing target which significantly simplifies the computer vision tasks such as corner detection and correspondence matching of a landing target. Tournier and *et al.* in [14] have been attempted to acquire the six degree of freedom estimation of a Quadrotor vehicle using a single camera relative to a novel target that incorporates the use of moiré patterns.

Yu and *et al.* in [40] have been proposed a vibration free method for estimating the height over the ground based on a 3D vision system without using complicated gimbal or active vision mechanism.

Wise and *et al.* in [43] have been compared several different methodologies for tracking a moving target with multiple Unmanned Aerial Vehicles. STARMAC research group also have been contributed on the path tracking of UAV in several articles; Hoffmann and *et al.* in [34, 39] have been developed an autonomous vehicle trajectory tracking algorithm through space-indexed way-points for the STARMAC platform. Voos in [37] has been developed a see and avoid system for UAVs based on command to line-of-sight method. Ha and *et al.* in [12] have been provided an application of feed-back linearization technique to design a command to line-of-sight (CLOS) guidance law for short-range surface-to-air missiles in which the key idea was lied in converting the three-dimensional CLOS guidance problem to tracking problem of a time-varying nonlinear system. Rysdyk and *et al.* in [44, 45] have been implemented a guidance law to orient the trajectories of UAV

about the target based on ‘good helmsman’ behavior with an observer to estimate wind data.

2.3 Study of the Wind Effect

Osborne and *et al.* in [50] have been developed a guidance algorithm for an autonomous Unmanned Aerial Vehicle (UAV) which includes an observer based wind estimator. Moreover, Etele in [30] have been studied the available methods for quantifying a wind gust in an attempt to use this quantification in the prediction of wind effect on UAV stability and to provide a realistic understanding of the environmental issues associated with UAV operations in urban and mountainous environments.

2.4 Sensors

Some of the literature discussing about methods and proposed sensors for measuring quantities relevant to this thesis work, have already been introduced in previous sections. Bouabdallah and *et al.* in [42] have been implemented an active control system for a Quadrotor with the help of ultra sound sensors for detecting obstacles and controlling altitude. Stoleru and *et al.* in [49] have been designed a *Walking GPS* as an effective localization solution.

Chapter 3

Quadrotor Model

3.1 Quadrotor Structure

A Quadrotor is a rotor-craft equipped with four rotors laid up symmetrically around its center. Because of its abilities to hover, forward flight and vertical take off and landing, it could be classified as a rotarywing VTOL¹ aircraft.

Traditionally the Quadrotor configuration has not been used in the aerospace industry, mainly because of its high power consumption. However, Quadrotor possesses some unique characteristics, such as superior payload capacity, simplicity of its mechanical structure and low cost maintenance, and possibility of controlling both of its attitude and linear velocity just by independently adjusting the speed of each rotor. These are some reasons out of many that brought Quadrotor structure back to the attention of many research groups recently.

Figure 3.1 shows a free body diagram of a Quadrotor structure. As it could be conceived from this diagram, the Quadrotor structure, consists of two pairs of rotors, rotating in opposite directions; usually the front and rear rotors are rotating clockwise, while the right and left rotors are rotating counterclockwise. In this way the rotors will compensate the rotational torque of each other and hence the overall rotational torque of the rotors will be zero during the hover position.

Another privilege of such structure is the possibility to control attitude and hence the linear velocity of the Quadrotor just by independently adjusting the speed of each rotor. Figure 3.2 shows this ability.

¹Vertical Take-Off and Landing

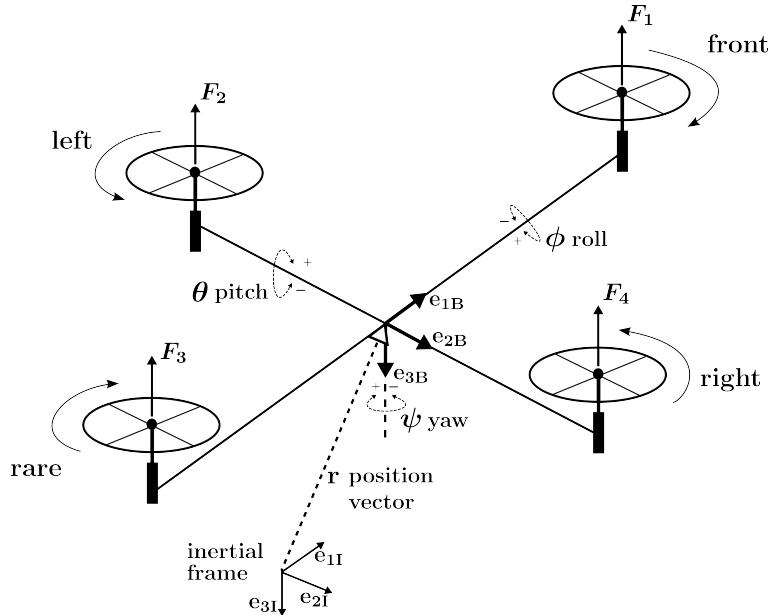


Figure 3.1: Free body diagram of a Quadrotor structure.

3.2 Mathematical Model

In this thesis work, a mathematical description of a simplified model of Quadrotor has been applied, which mainly based on Voos work, [28], with some minor alteration by adding of the position of Quadrotor as new state space variables. Table 3.1 provides definition of the parameters which are used in this model.

In his model, transformation matrix form body to inertia frame, \mathbf{R} , defined as follow:

$$\mathbf{R} = \begin{pmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{pmatrix} \quad (3.1)$$

In addition the torque applied to the vehicle's body, \mathbf{M} , defined as follow:

$$\mathbf{M} = \begin{pmatrix} Lb(\omega_2^2 - \omega_4^2) \\ Lb(\omega_1^2 - \omega_3^2) \\ d(\omega_1^2 + \omega_3^2 - \omega_2^2 - \omega_4^2) \end{pmatrix} \quad (3.2)$$

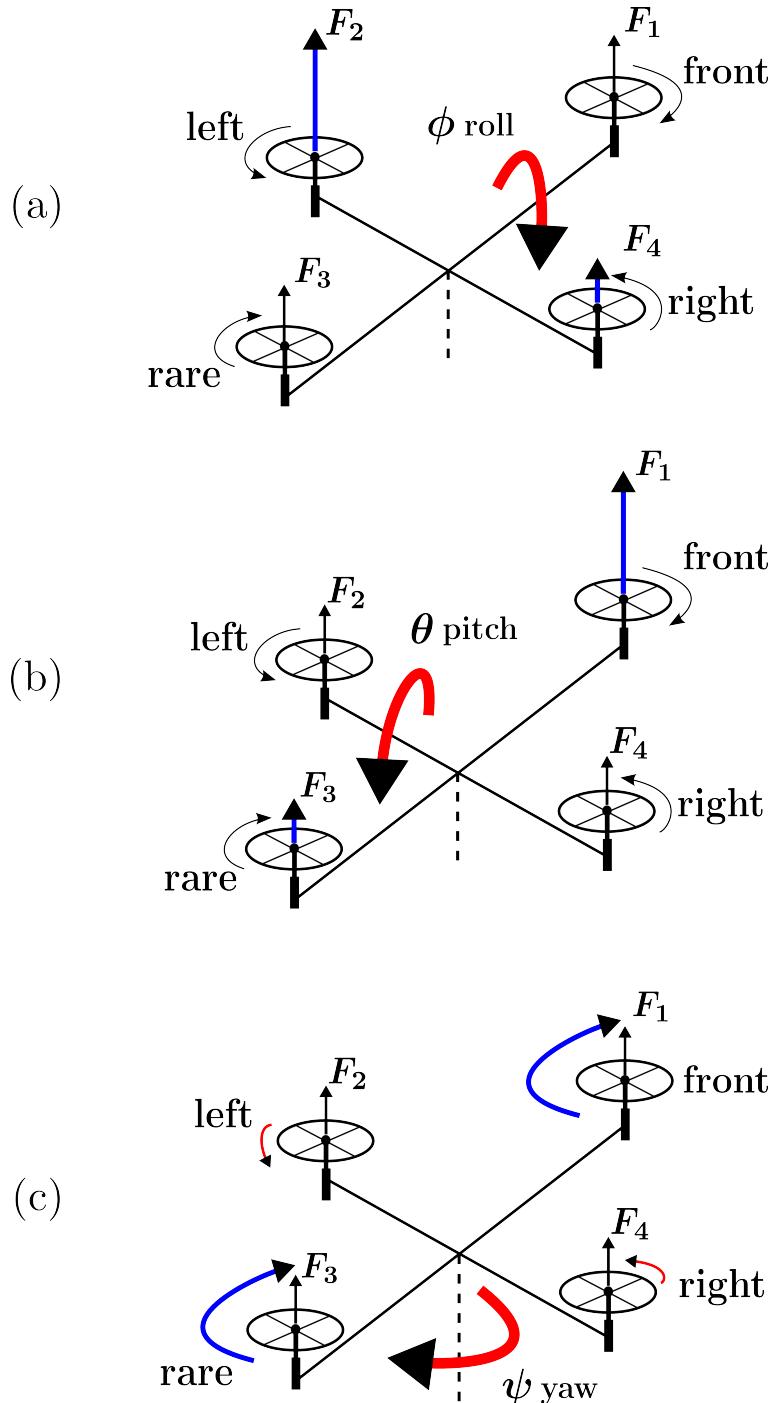


Figure 3.2: Attitude control of a Quadrotor; (a) roll angle, (b) pitch angle, (c) yaw angle

Notation	Description
g	Gravity of the Earth
m	Mass of the Quadrotor
L	Lever's Length
ω_i	Rotational Velocity of i_{th} Motor ($i = 1, \dots, 4$)
d	Drag Factor
b	Thrust Factor
F_i	Thrust Force of i_{th} Motor ($F_i = b \cdot \omega_i^2$)
Ω	Orientation (ϕ, θ, ψ)
ϕ	Roll Angle
θ	Pitch Angle
ψ	Yaw Angle
r	Position (x, y, z)

Table 3.1: Definition of the parameters used in the Quadrotor model

Rotor inertia is $\mathbf{J}_R = \mathbf{I}_R$, in which the inertia matrix, \mathbf{I} , is an identity matrix and defined as follow:

$$\mathbf{I} = \begin{pmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{pmatrix} \quad (3.3)$$

Gyroscopic torque (caused by rotation of vehicle) could be formulated as equation (3.4), in which $\mathbf{g}(u)$ is a function of control inputs, $(\omega_1, \omega_2, \omega_3, \omega_4)$, and defined in equation (3.5).

$$\mathbf{M}_G = I_R \cdot (\dot{\Omega} \times \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}) \cdot \underbrace{(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2)}_{\mathbf{g}(u)} \quad (3.4)$$

$$\mathbf{g}(u) = \omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2 \quad (3.5)$$

Although the actual control inputs of the system are the rotational speed of the rotors, however, for simplification, the control inputs of the system chosen as a function of rotational speed of the rotors and defined in equation (3.6).

$$\bar{\mathbf{u}} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ b(\omega_2^2 - \omega_4^2) \\ b(\omega_1^2 - \omega_3^2) \\ d(\omega_1^2 + \omega_3^2 - \omega_2^2 - \omega_4^2) \end{pmatrix} \quad (3.6)$$

Also, for more compact formulation, I_x , I_y , and I_z have been replaced with I_1 , I_2 , and I_3 , which are defined in equation (3.7).

$$\begin{aligned} \mathbf{I}_1 &= \frac{I_y - I_z}{I_x} \\ \mathbf{I}_2 &= \frac{I_z - I_x}{I_y} \\ \mathbf{I}_3 &= \frac{I_x - I_y}{I_z} \end{aligned} \quad (3.7)$$

Choosing the state space variables of the Quadrotor model to be as (3.8), the underling mathematical model of the Quadrotor in the state space model, could be represented by equation (3.9), in which g is the gravity of the Earth, while $g(u)$ is driven from equation (3.5).

$$\mathbf{X} = \begin{pmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \phi \\ \theta \\ \psi \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{pmatrix} \quad (3.8)$$

$$\dot{\mathbf{X}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = \begin{cases} \dot{x}_1 = x_4 \\ \dot{x}_2 = x_5 \\ \dot{x}_3 = x_6 \\ \dot{x}_4 = -(cosx_7sinx_8cosx_9 + sinx_7sinx_9) \cdot \frac{u_1}{m} \\ \dot{x}_5 = -(cosx_7sinx_8sinx_9 + sinx_7cosx_9) \cdot \frac{u_1}{m} \\ \dot{x}_6 = g - (cosx_7cosx_8) \cdot \frac{u_1}{m} \\ \dot{x}_7 = x_{10} \\ \dot{x}_8 = x_{11} \\ \dot{x}_9 = x_{12} \\ \dot{x}_{10} = x_{11}x_{12}I_1 - \frac{I_R}{I_x}x_{11}g(u) + \frac{L}{I_x}u_2 \\ \dot{x}_{11} = x_{10}x_{12}I_2 - \frac{I_R}{I_y}x_{10}g(u) + \frac{L}{I_y}u_3 \\ \dot{x}_{12} = x_{10}x_{11}I_3 + \frac{1}{I_z}u_4 \end{cases} \quad (3.9)$$

It is obvious that this is a very simplified model valid only for evaluation of the proposed control algorithms in simulation environment, in which important effects such as the aerodynamic, frictional, and blade elasticity effects are ignored.

Chapter 4

Attitude Stabilization

Since the Quadrotor is intrinsically an unstable nonlinear system, before developing of any control algorithm for maneuvering and navigation of the Quadrotor, it is essential to stabilize it in hover position. The nonlinearity of the underlying differential equations, make the attitude stabilization of the Quadrotor a challenging problem; however, this problem has been contributed in several literature and various methods for attitude stabilization of the Quadrotor have been proposed [36, 5, 28, 6, 27, 29].

In [28], Voos has been proposed a method for attitude stabilization of the Quadrotor based on feedback linearization, which is mainly adapted in this thesis work.

4.1 Attitude Control System

In chapter 3, the Quadrotor model has been formulated by a set of differential equations represented by equation (3.9). From this model $M1$ could be chosen to represent the last three state variables, $(\ddot{\phi}, \ddot{\theta}, \ddot{\psi})$.

$$\begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = \begin{pmatrix} \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} = \begin{cases} x_{11}x_{12}I_1 - \frac{I_R}{I_x}x_{11}g(u) + \frac{L}{I_x}u_2 \\ x_{10}x_{12}I_2 - \frac{I_R}{I_y}x_{10}g(u) + \frac{L}{I_y}u_3 \\ x_{10}x_{11}I_3 + \frac{1}{I_z}u_4 \end{cases} \quad (4.1)$$

To simplify $M1$, rotor inertia, I_R could be neglected, as it has relatively small value.

$$\mathbf{M1} : \begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = \begin{pmatrix} \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} \simeq \begin{cases} x_{11}x_{12}I_1 + \frac{L}{I_x}u_2 \triangleq f_2 \\ x_{10}x_{12}I_2 + \frac{L}{I_y}u_3 \triangleq f_3 \\ x_{10}x_{11}I_3 + \frac{1}{I_z}u_4 \triangleq f_4 \end{cases} \quad (4.2)$$

One possible method for controlling nonlinear system of $M1$ is feedback linearization. In this method, control input, u^* , could be defined in a way to linearize the overall system. For that purpose, a set of new control inputs defined as follow.

$$\begin{cases} u_2 = f_2 + u_2^* \\ u_3 = f_3 + u_3^* \\ u_4 = f_4 + u_4^* \end{cases} \quad (4.3)$$

Substituting the new inputs of (4.3) in equation (4.2) and rearranging the resultant system yields

$$\begin{pmatrix} \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{pmatrix} = \begin{cases} x_{11}x_{12}I_1 + \frac{L}{I_x}(f_2 + u_2^*) \\ x_{10}x_{12}I_2 + \frac{L}{I_y}(f_3 + u_3^*) \\ x_{10}x_{11}I_3 + \frac{1}{I_z}(f_4 + u_4^*) \end{cases} \quad (4.4)$$

$$\begin{cases} x_{11}x_{12}I_1 + \frac{L}{I_x}f_2 = \dot{x}_{10} - \frac{L}{I_x}u_2^* \\ x_{10}x_{12}I_2 + \frac{L}{I_y}f_3 = \dot{x}_{11} - \frac{L}{I_y}u_3^* \\ x_{10}x_{11}I_3 + \frac{1}{I_z}f_4 = \dot{x}_{12} - \frac{1}{I_z}u_4^* \end{cases} \quad (4.5)$$

From (4.5) the resultant linear system could be chosen as follow.

$$\begin{cases} \dot{x}_{10} - \frac{L}{I_x}u_2^* \triangleq K_2 \cdot x_{10} \\ \dot{x}_{11} - \frac{L}{I_y}u_3^* \triangleq K_3 \cdot x_{11} \\ \dot{x}_{12} - \frac{1}{I_z}u_4^* \triangleq K_4 \cdot x_{12} \end{cases} \quad (4.6)$$

in which K_2, K_3, K_4 are the control parameters and should be chosen in a way to stabilize the system.

Rearranging 4.6, the resultant linearized system will be as follow.

$$\begin{cases} \dot{x}_{10} = K_2 \cdot x_{10} + \frac{L}{I_x} u_2^* \\ \dot{x}_{11} = K_3 \cdot x_{11} + \frac{L}{I_y} u_3^* \\ \dot{x}_{12} = K_4 \cdot x_{12} + \frac{1}{I_z} u_4^* \end{cases} \quad (4.7)$$

The stability of the resultant linear system could be assured by Lyapunov criteria. For that reason, an appropriate Lyapunov function should be defined.

$$V(x_{10}, x_{11}, x_{12}) = 0.5(x_{10}^2 + x_{11}^2 + x_{12}^2) \quad (4.8)$$

Choosing $u_2^* = u_3^* = u_4^* = 0$ and $(x_{10}, x_{11}, x_{12}) = (\dot{\phi}, \dot{\theta}, \dot{\psi}) = (0, 0, 0)$, as operational point of the system, the derivative of above mentioned Lyapunov function will be as follow

$$\dot{V} = x_{10}\dot{x}_{10} + x_{11}\dot{x}_{11} + x_{12}\dot{x}_{12} \quad (4.9)$$

$$\dot{V} = K_2 \cdot x_{10}^2 + K_3 \cdot x_{11}^2 + K_4 \cdot x_{12}^2 \quad (4.10)$$

Lyapunov criteria assures the stability of the system for all values of K_2 , K_3 , and K_4 in which $\dot{V} < 0$. Therefore, the system will be stable if $K_2 < 0$, $K_3 < 0$, and $K_4 < 0$. In addition, attitude control is a time critical task and K_2 , K_3 , K_4 should be designed in a way that the overall control system has small response time.

Reconsidering equation (4.7), desired attitude, $(x_{7d}, x_{8d}, x_{9d},) = (\phi_d, \theta_d, \psi_d)$, is achievable by appropriate choice of control inputs, u_i^* .

$$\ddot{\phi} = \ddot{x}_7 = K_2 \cdot \dot{x}_7 + \frac{L}{I_x} u_2^* \quad (4.11)$$

$$\ddot{\theta} = \ddot{x}_8 = K_3 \cdot \dot{x}_8 + \frac{L}{I_y} u_3^* \quad (4.12)$$

$$\ddot{\psi} = \ddot{x}_9 = K_4 \cdot \dot{x}_9 + \frac{1}{I_z} u_4^* \quad (4.13)$$

$$u_2^* = a_2 \cdot (x_{7d} - x_7) \quad (4.14)$$

$$u_3^* = a_3 \cdot (x_{8d} - x_8) \quad (4.15)$$

$$u_4^* = a_4 \cdot (x_{9d} - x_9) \quad (4.16)$$

in which a_2, a_3, a_4 should be chosen appropriately in order to have small response time for attitude control system.

Substitute (4.14), (4.15), and (4.16) in (4.11), (4.12), and (4.13) respectively, the underlying relations representing the dynamic of the attitude control system could be derived.

$$\ddot{\phi} = K_2 \cdot \dot{\phi} + \frac{L}{I_x} (a_2 \cdot (\phi_d - \phi)) \quad (4.17)$$

$$\ddot{\theta} = K_3 \cdot \dot{\theta} + \frac{L}{I_y} (a_3 \cdot (\theta_d - \theta)) \quad (4.18)$$

$$\ddot{\psi} = K_4 \cdot \dot{\psi} + \frac{1}{I_z} (a_4 \cdot (\psi_d - \psi)) \quad (4.19)$$

Taking *Laplace Transformation* from equation (4.17) with respect to ϕ_d as desired input, ϕ as output and considering K_2 , L , I_x , and a_2 as constants, the underlying dynamic of the roll angle control system will be presented by the following relation.

$$F_2(s) = \frac{\phi(s)}{\phi_d(s)} = \frac{a_2}{\frac{I_x}{L}s^2 - K_2 \frac{I_x}{L}s + a_2} \quad (4.20)$$

in which, for simplification, a_2 could be chosen as follow.

$$a_2 = \left(\frac{K_2}{2}\right)^2 \cdot \frac{I_x}{L} \quad (4.21)$$

Similarly the underlying dynamics of the pitch and yaw angle control system could be derived.

$$F_3(s) = \frac{\theta(s)}{\theta_d(s)} = \frac{a_3}{\frac{I_y}{L}s^2 - K_3 \frac{I_y}{L}s + a_3} \quad (4.22)$$

$$a_3 = \left(\frac{K_3}{2}\right)^2 \cdot \frac{I_y}{L} \quad (4.23)$$

$$F_4(s) = \frac{\psi(s)}{\psi_d(s)} = \frac{a_4}{\frac{I_z}{L}s^2 - K_4 \frac{I_z}{L}s + a_4} \quad (4.24)$$

$$a_4 = \left(\frac{K_4}{2}\right)^2 \cdot \frac{I_z}{L} \quad (4.25)$$

In order to achieve desired dynamic of the attitude control system, with respect to the Lyapunov criteria of $K_i < 0$, appropriate values of K_2 , K_3 , and K_4 could be chosen by the aid of *rltool* toolbox in MATLAB.

A schematic diagram of the closed loop attitude control system has been depicted in the figure (4.1).

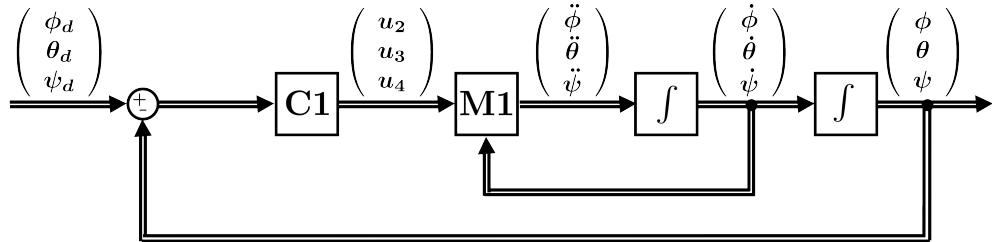


Figure 4.1: Schematic diagram of the attitude (inner) control loop.

4.2 Linear Velocity Control System

In Quadrotor system which represented by equation (3.9), the subsystem $M2$ could be chosen to represent the linear acceleration, $(\ddot{x}, \ddot{y}, \ddot{z})$.

$$\mathbf{M2} : \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{pmatrix} = \begin{cases} -(cosx_7d sinx_8d cosx_9d + sinx_7d sinx_9d) \cdot \frac{u_1}{m} \\ -(cosx_7d sinx_8d sinx_9d + sinx_7d cosx_9d) \cdot \frac{u_1}{m} \\ g - cosx_7d cosx_8d \cdot \frac{u_1}{m} \end{cases} \quad (4.26)$$

In the last section the dynamic of the attitude controller has been formulated by equations (4.20), (4.22), and (4.24). K_2 , K_3 , and K_4 could be adjusted appropriately to achieve a desired fast attitude control dynamic in which the Quadrotor is able to follow the desired attitude quickly; if the attitude controller is quick enough, it could be assumed as a stationary block in the overall control loop and its dynamic could be neglected.

Neglecting the attitude control dynamic, a linear velocity controller, $C2$, could be developed for the subsystem $M2$. A schematic diagram of the outer closed loop linear velocity controller has been depicted in the figure (4.2).

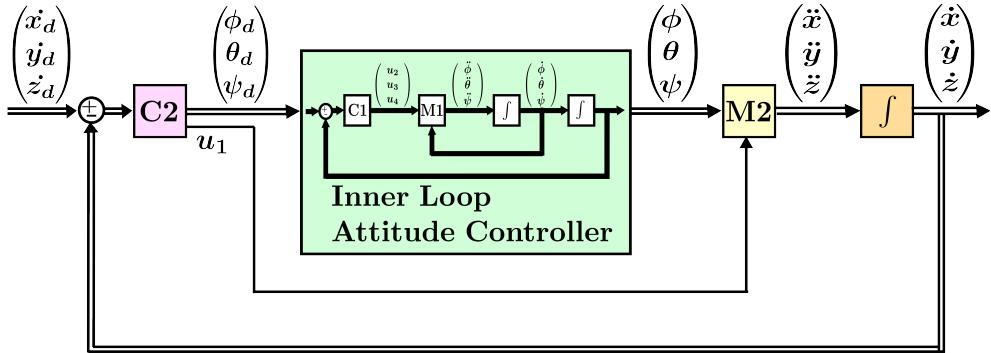


Figure 4.2: Outer linear velocity control loop; the inner attitude control loop assumed to be quick enough and neglected.

Since M_2 is a nonlinear system, feedback linearization could also be applied. For that reason, a set of control inputs, $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3$, have been chosen and the resultant linear system is as follow.

$$\text{Control Inputs : } \begin{cases} \tilde{u}_1 = K_x(x_{4d} - x_4) = K_x(\dot{x}_d - \dot{x}) \\ \tilde{u}_2 = K_y(x_{5d} - x_5) = K_y(\dot{y}_d - \dot{y}) \\ \tilde{u}_3 = K_z(x_{6d} - x_6) = K_z(\dot{z}_d - \dot{z}) \end{cases} \quad (4.27)$$

$$\text{Linearized M2 : } \begin{cases} \dot{x}_4 = \tilde{u}_1 \\ \dot{x}_5 = \tilde{u}_2 \\ \dot{x}_6 = \tilde{u}_3 \end{cases} \quad (4.28)$$

Substitute equations (4.26) and (4.27) in (4.28) yields

$$\begin{cases} K_x(x_{4d} - x_4) = -(cosx_7d sinx_8d cosx_9d + sinx_7d sinx_9d) \cdot \frac{u_1}{m} \\ K_y(x_{5d} - x_5) = -(cosx_7d sinx_8d sinx_9d + sinx_7d cosx_9d) \cdot \frac{u_1}{m} \\ K_z(x_{6d} - x_6) = g - cosx_7d cosx_8d \cdot \frac{u_1}{m} \end{cases} \quad (4.29)$$

or

$$\begin{cases} K_x(\dot{x}_d - \dot{x}) = -(cos\phi_d sin\theta_d cos\psi_d + sin\phi_d sin\psi_d) \cdot \frac{u_1}{m} \\ K_y(\dot{y}_d - \dot{y}) = -(cos\phi_d sin\theta_d sin\psi_d - sin\phi_d cos\psi_d) \cdot \frac{u_1}{m} \\ K_z(\dot{z}_d - \dot{z}) = g - cos\phi_d cos\theta_d \cdot \frac{u_1}{m} \end{cases} \quad (4.30)$$

In equation (4.30), K_x , K_y , K_z are the control parameters which could be chosen appropriately in order to achieve desired dynamic. $(\dot{x}_d, \dot{y}_d, \dot{z}_d)$ are desired linear velocity vector, and $(\dot{x}, \dot{y}, \dot{z})$ are linear velocities in reference coordinate system and could be measured by the aids of an appropriate sensor, such as IMU¹.

The schematic diagram of the overall control loop has been depicted in the figure (4.3).

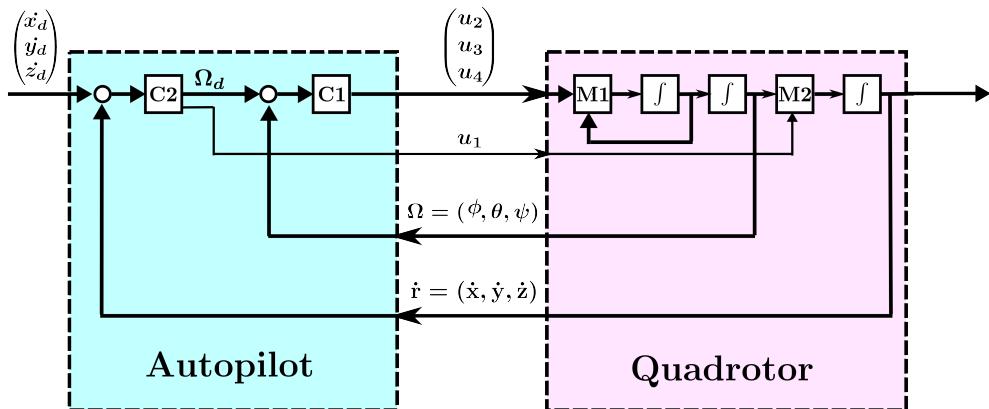


Figure 4.3: The schematic diagram of the overall control loop.

After choosing appropriate values for control parameters, K_x , K_y , and K_z , a set of three equations with four variables, $(\psi_d, \theta_d, \phi_d)$, u_1 have been concluded. However, since control inputs, $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3$, are chosen according to linear velocities, $\dot{x}, \dot{y}, \dot{z}$, and since yaw angle, ψ , has no effect in the linear velocities, it could be eliminated from the equations.

$$x_{9d} = \psi_d = 0 \quad (4.31)$$

¹Inertia Measurement Unit

Eliminating ψ in (4.30), a set of three equations with three unknown parameters has been resulted.

$$\begin{cases} K_x(\dot{x}_d - \dot{x}) = -\cos\phi_d \sin\theta_d \cdot \frac{u_1}{m} \\ K_y(\dot{y}_d - \dot{y}) = \sin\phi_d \cdot \frac{u_1}{m} \\ K_z(\dot{z}_d - \dot{z}) = g - \cos\phi_d \cos\theta_d \cdot \frac{u_1}{m} \end{cases} \quad (4.32)$$

Solving this equations gives the desired variables, namely $(\psi_d, \theta_d, \phi_d)$, u_1 , which are the inputs of the inner attitude control system.

In order to solve these nonlinear equations analytically, two intermediate variables, α , β , could be defined as follow.

$$\begin{cases} \alpha \triangleq \sin\phi_d \\ \beta \triangleq \sin\theta_d \end{cases} \Rightarrow \begin{cases} \cos\phi_d = \pm\sqrt{1 - \alpha^2} \\ \cos\theta_d = \pm\sqrt{1 - \beta^2} \end{cases} \quad (4.33)$$

Substituting equation (4.33) in (4.32), will result in

$$\begin{cases} K_x(\dot{x}_d - \dot{x}) = \mp\sqrt{1 - \alpha^2} \cdot \beta \cdot \frac{u_1}{m} \\ K_y(\dot{y}_d - \dot{y}) = \alpha \cdot \frac{u_1}{m} \\ K_z(\dot{z}_d - \dot{z}) = g - (\mp\sqrt{1 - \alpha^2}) \cdot (\pm\sqrt{1 - \beta^2}) \cdot \frac{u_1}{m} \end{cases} \quad (4.34)$$

ϕ_d , θ_d , u_1 could be calculated analytically from equation (4.34).

Chapter 5

Command to Line-Of-Sight Navigation

One of the classical techniques for missile guidance is Command to Line-Of-Sight (CLOS) system, which is mainly adapted in this thesis work for navigation of the Quadrotor. The CLOS system uses only the angular coordinates between the missile and the target to ensure the collision. The missile will have to be in the line of sight between the launcher and the target (LOS), and controller is responsible for correcting any deviation of the missile in relation to this line.

In Quadrotor navigation scenario, the helipad hasn't any motion along the z-axis and therefore, its height above the ground could be assumed constant; this assumption will simplify the navigation of the Quadrotor to a two dimensional problem in a reference coordinate system and the CLOS system has to keep the Quadrotor moving along the line of sight between Quadrotor and elevated center of the helipad, mapped on the xy-plane containing Quadrotor, which has been depicted in the figure 5.1.

5.1 CLOS System

Underlying kinematic of the Quadrotor relative to the helipad, which has been depicted in the figure (5.2), could be formulated as follow respectively in the Cartesian and polar coordinate system.

$$\dot{x} = V_P \cdot \cos \alpha_P - V_Q \cdot \cos \alpha_Q \quad (5.1a)$$

$$\dot{y} = V_P \cdot \sin \alpha_P - V_Q \cdot \sin \alpha_Q \quad (5.1b)$$

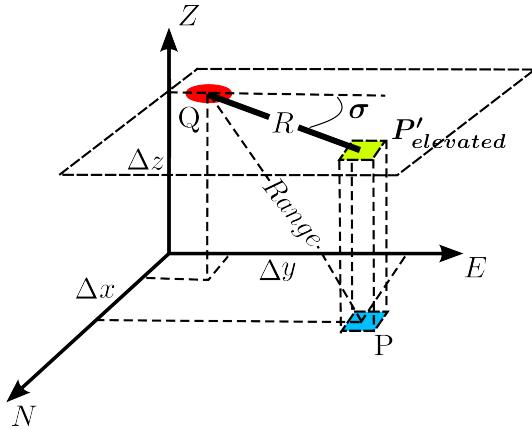


Figure 5.1: CLOS system for a two-dimensional scenario; $R = \sqrt{\Delta x^2 + \Delta y^2}$ is the planer distance between Quadrotor and the elevated center of the helipad, mapped on the xy-plane containing Quadrotor; $\text{Range} = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$ is the distance between Quadrotor and center of the helipad.

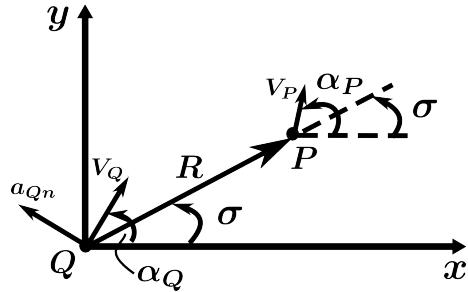


Figure 5.2: Engagement geometry of Quadrotor and helipad.

$$\dot{R} = V_P \cdot \cos(\alpha_P - \sigma) - V_Q \cdot \cos(\alpha_Q - \sigma) \quad (5.2a)$$

$$R \cdot \dot{\sigma} = V_P \cdot \sin(\alpha_P - \sigma) - V_Q \cdot \sin(\alpha_Q - \sigma) \quad (5.2b)$$

The normal and tangential acceleration components, which have been depicted in the figure (5.3), could be calculated as follow.

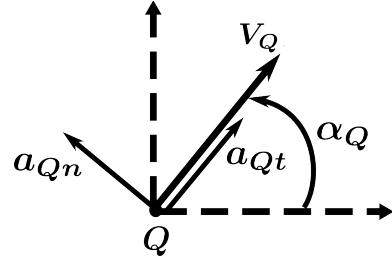


Figure 5.3: Normal and tangential acceleration of the Quadrotor.

$$\dot{V}_Q = \frac{d}{dt} \left(V_Q \cdot \begin{bmatrix} \cos \alpha_Q \\ \sin \alpha_Q \end{bmatrix} \right) = \underbrace{\dot{V}_Q \cdot \begin{pmatrix} \cos \alpha_Q \\ \sin \alpha_Q \end{pmatrix}}_{=a_{Qt}} + \underbrace{V_Q \cdot \begin{pmatrix} -\sin \alpha_Q \\ \cos \alpha_Q \end{pmatrix} \cdot \dot{\alpha}_Q}_{=a_{Qn}} \quad (5.3)$$

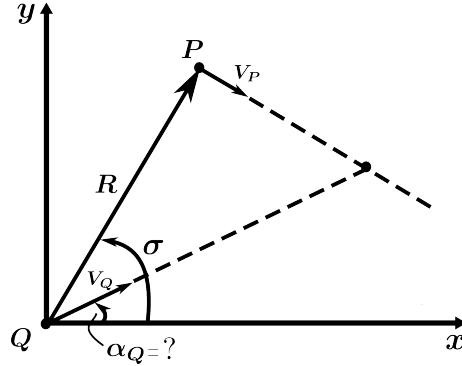
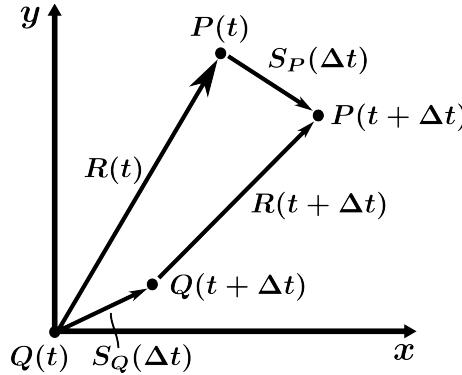
While \underline{a}_{Qt} is not controllable, the motion path of the Quadrotor could be maneuvered by \underline{a}_{Qn} .

$$\underline{a}_{Qn} = V_Q \cdot \dot{\alpha}_Q \cdot \begin{pmatrix} -\sin \alpha_Q \\ \cos \alpha_Q \end{pmatrix} \quad (5.4)$$

$$|\underline{a}_{Qn}| = V_Q \cdot \dot{\alpha}_Q \quad (5.5)$$

Assuming V_P , V_Q , and α_P to be constant, as depicted in the figure (5.4), α_Q could be adjusted to maneuver Quadrotor to cruise toward and align with the center of the helipad.

The relative coordination of the Quadrotor and helipad within Δt seconds of navigation has been depicted in the figure (5.5). From this diagram, the kinematic of the motion path could be formulated as follow.

Figure 5.4: Controlling Quadrotor motion path by adjusting α_Q .Figure 5.5: Coordination of the Quadrotor and helipad within Δt seconds of navigation.

$$\underline{R}(t) + \underline{S}_P(\Delta t) = \underline{S}_Q(\Delta t) + \underline{R}(t + \Delta t) \quad (5.6)$$

$$\underline{R}(t + \Delta t) - \underline{R}(t) = \underline{S}_P(\Delta t) - \underline{S}_Q(\Delta t) \quad (5.7)$$

$$\frac{\underline{R}(t + \Delta t) - \underline{R}(t)}{\Delta t} = \frac{\underline{S}_P(\Delta t)}{\Delta t} - \frac{\underline{S}_Q(\Delta t)}{\Delta t} \quad (5.8)$$

$$\lim_{\Delta t \rightarrow 0} \frac{\underline{R}(t + \Delta t) - \underline{R}(t)}{\Delta t} = \lim_{\Delta t \rightarrow 0} \left(\frac{\underline{S}_P(\Delta t)}{\Delta t} - \frac{\underline{S}_Q(\Delta t)}{\Delta t} \right) \quad (5.9)$$

$$\dot{\underline{R}}(t) = \underline{V}_P(t) - \underline{V}_Q(t) \quad (5.10)$$

Since V_P , α_P , V_Q , and α_Q assumed to be constant, therefore, $\dot{\underline{R}}(t)$ will also be constant. In order that the Quadrotor would be able to align with the helipad within $t = t_{go}$, the following condition should be fulfilled.

$$\underline{R}(t) + \dot{\underline{R}}(t) \cdot t_{go} \triangleq 0 \quad (5.11)$$

Equation (5.11) implies that \underline{R} and $\dot{\underline{R}}$ should be parallel. In the other words, to assure collision within $t = t_{go}$ seconds, $\dot{\underline{R}}$ should not contain any component vertical to \underline{R} . $\dot{\underline{R}}$ could be calculated as follow.

$$\dot{\underline{R}}(t) = \frac{d}{dt} \left(\underline{R}(t) \begin{pmatrix} \cos \sigma \\ \sin \sigma \end{pmatrix} \right) = \underbrace{\dot{\underline{R}}(t) \begin{pmatrix} \cos \sigma \\ \sin \sigma \end{pmatrix}}_{\text{Parallel Component of } \underline{R}} + \underbrace{\underline{R}(t) \begin{pmatrix} -\sin \sigma \\ \cos \sigma \end{pmatrix} \dot{\sigma}(t)}_{\text{Vertical Component of } \underline{R}} \quad (5.12)$$

Therefore, the following condition for the collision path could be concluded.

$$\begin{aligned} & \text{Vertical Component of } \underline{R} \stackrel{!}{=} 0 \\ & \underline{R}(t) \cdot \begin{pmatrix} -\sin \sigma \\ \cos \sigma \end{pmatrix} \cdot \dot{\sigma}(t) \stackrel{!}{=} 0 \\ & \dot{\sigma}(t) = 0 \end{aligned} \quad (5.13)$$

Equation (5.13) implies that in the collision path the line of sight angle, σ , should be constant as the ratio of changes in line of sight angle is zero. The collision path in CLOS system has been depicted in the figure (5.6).

Substituting collision condition of (5.13) in underlying kinematic of the Quadrotor relative to the helipad, represented in equation (5.2b), the collision course, α_{QC} , could be calculated as follow.

$$\begin{aligned} 0 &= V_P \cdot \sin(\alpha_P - \sigma) - V_Q \cdot \sin(\alpha_{QC} - \sigma) \\ \alpha_{QC} &= \sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \end{aligned} \quad (5.14)$$

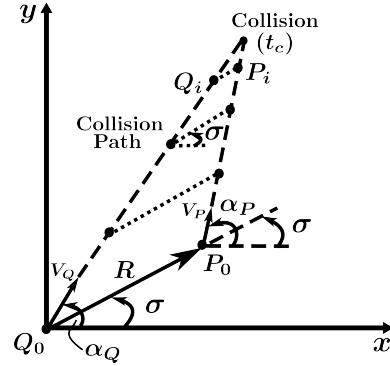


Figure 5.6: Collision path in CLOS system.

With the collision course, the velocity vector of the Quadrotor could be characterized as follow.

$$\begin{aligned} V_{Qx} &= V_Q \cdot \cos \alpha_{QC} \\ &= V_Q \cdot \cos \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \right) \end{aligned} \quad (5.15a)$$

$$\begin{aligned} V_{Qy} &= V_Q \cdot \sin \alpha_{QC} \\ &= V_Q \cdot \sin \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \right) \end{aligned} \quad (5.15b)$$

Although collision course, α_{QC} , is a necessary required condition for Quadrotor in order to be able to follow the helipad, however, it is not guarantee the collision and alignment of the Quadrotor with the center of the helipad. In order to fulfill collision, the Quadrotor should also move faster than helipad, which means the magnitude of the V_Q should be adequately bigger than V_P . For that purpose, V_Q could be chosen to be proportional with the planer distance between Quadrotor and elevated center of helipad, R .

$$\begin{aligned} V_Q &\propto R \\ V_Q &= V_P + k \cdot R \end{aligned} \quad (5.16)$$

5.2 Adaption of the CLOS System for Navigation Control Design

In the last section, CLOS system has been introduced and equations (5.15a), (5.15b), and (5.16) have been derived for navigation of the Quadrotor. Although these equation directly could be applied for maneuvering of the Quadrotor in the collision path, however, in this section CLOS system would be adapted for a systematic navigation control system design.

Assuming that V_P , α_P , and σ , could be measured by the aid of appropriate sensors and are available at any time, the aim is to develop a control system for navigation of the Quadrotor based on CLOS system. Relative kinematic of the Quadrotor and helipad have been presented by equations (5.2a), (5.2b), which could be rewritten as follow.

$$\begin{aligned}\dot{R} &= V_P \cdot \cos(\alpha_P - \sigma) - V_Q \cdot \cos(\alpha_Q - \sigma) \\ &= V_P \cos \alpha_P \cos \sigma + V_P \sin \alpha_P \sin \sigma - V_Q \cos \alpha_Q \cos \sigma - V_Q \sin \alpha_Q \sin \sigma\end{aligned}\quad (5.17a)$$

$$\begin{aligned}\dot{\sigma} &= \frac{1}{R} (V_P \cdot \sin(\alpha_P - \sigma) - V_Q \cdot \sin(\alpha_Q - \sigma)) \\ &= \frac{1}{R} (V_P \sin \alpha_P \cos \sigma - V_P \cos \alpha_P \sin \sigma - V_Q \sin \alpha_Q \cos \sigma + V_Q \cos \alpha_Q \sin \sigma)\end{aligned}\quad (5.17b)$$

Denoting $V_{Px} = V_P \cos \alpha_P$, $V_{Py} = V_P \sin \alpha_P$, $V_{Qx} = V_Q \cos \alpha_Q$, $V_{Qy} = V_Q \sin \alpha_Q$, equations (5.17a), and (5.17b) could be simplified as follow.

$$\dot{R} = V_{Px} \cos \sigma + V_{Py} \sin \sigma - V_{Qx} \cos \sigma - V_{Qy} \sin \sigma \quad (5.18a)$$

$$\dot{\sigma} = \frac{1}{R} (V_{Py} \cos \sigma - V_{Px} \sin \sigma - V_{Qy} \cos \sigma + V_{Qx} \sin \sigma) \quad (5.18b)$$

Choosing $x_1 = R$, and $x_2 = \sigma$ as state variables and $u_1 = V_{Qx}$, and $u_2 = V_{Qy}$ as the control inputs, nonlinear equations (5.18a), (5.18b) could be represented in state space mode, with $d_1 = V_{Px}$, and $d_2 = V_{Py}$ as two measurable disturbances for the model.

$$\dot{x}_1 = d_1 \cdot \cos x_2 + d_2 \cdot \sin x_2 - u_1 \cdot \cos x_2 - u_2 \cdot \sin x_2 \quad (5.19a)$$

$$\dot{x}_2 = \frac{1}{x_1} \left(d_2 \cdot \cos x_2 - d_1 \cdot \sin x_2 - u_2 \cdot \cos x_2 + u_1 \cdot \sin x_2 \right) \quad (5.19b)$$

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{bmatrix} -\cos x_2 & -\sin x_2 \\ \frac{\sin x_2}{x_1} & -\frac{\cos x_2}{x_1} \end{bmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \begin{bmatrix} \cos x_2 & \sin x_2 \\ -\frac{\sin x_2}{x_1} & \frac{\cos x_2}{x_1} \end{bmatrix} \cdot \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} \quad (5.20)$$

$$\dot{x} = B(x)u + D(x)d \quad (5.21a)$$

$$B(x) = \begin{bmatrix} -\cos x_2 & -\sin x_2 \\ \frac{\sin x_2}{x_1} & -\frac{\cos x_2}{x_1} \end{bmatrix} \quad (5.21b)$$

$$D(x) = \begin{bmatrix} \cos x_2 & \sin x_2 \\ -\frac{\sin x_2}{x_1} & \frac{\cos x_2}{x_1} \end{bmatrix} \quad (5.21c)$$

Control inputs, $u_1 = V_{Qx}$, and $u_2 = V_{Qy}$, are responsible for the navigation of the Quadrotor and should be adjusted based on CLOS system to maneuver Quadrotor to cruise toward and align with the helipad.

The model represented in (5.21a) is a nonlinear system and new control input, u^* , could be chosen in a way to linearize the system. The resultant linear system would be as follow.

$$\dot{x} = u^* \quad (5.22)$$

$$u_1^* = k_1 \cdot (x_{1,d} - x_1) \quad (5.23a)$$

$$u_2^* = k_2 \cdot (x_{2,d} - x_2) \quad (5.23b)$$

In equations (5.23a) and (5.23b), $x_{1,d} = R_d$ and $x_{2,d} = \sigma_d$ are the desired state variables. In order to align Quadrotor with the helipad, the desired

planer range is zero, $x_{1,d} = R_d = 0$. In addition, Quadrotor could be aligned with the center of the helipad in any arbitrary orientation and for simplification it could be chosen also to be zero, $x_{2,d} = \sigma_d = 0$. Therefore, the new control inputs would be as follow.

$$u_1^* = -k_1 \cdot x_1 \quad (5.24a)$$

$$u_2^* = -k_2 \cdot x_2 \quad (5.24b)$$

in which k_1, k_2 should be adjusted based on CLOS system.

Combining (5.21a) and (5.22) yields

$$u^* = B(x)u + D(x)d \quad (5.25)$$

from which the control inputs of the system could be calculated as follow.

$$u = B^{-1}(x) \cdot (u^* - D(x)d) \quad (5.26)$$

$B^{-1}(x)$ is the inverse matrix of $B(x)$ and could be derived from (5.21b).

$$B^{-1}(x) = \begin{bmatrix} -\cos x_2 & x_1 \sin x_2 \\ -\sin x_2 & -x_1 \cos x_2 \end{bmatrix} \quad (5.27)$$

The control inputs could be derived by substituting equations (5.27), (5.24a), (5.24b), and (5.21c) in the equation (5.26).

$$\begin{aligned}
\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} &= \begin{bmatrix} -\cos x_2 & x_1 \sin x_2 \\ -\sin x_2 & -x_1 \cos x_2 \end{bmatrix} \\
&\quad \cdot \left(\begin{bmatrix} -k_1 x_1 \\ -k_2 x_2 \end{bmatrix} - \begin{bmatrix} \cos x_2 & \sin x_2 \\ -\frac{\sin x_2}{x_1} & \frac{\cos x_2}{x_1} \end{bmatrix} \cdot \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \right) \\
&= \begin{bmatrix} -\cos x_2 & x_1 \sin x_2 \\ -\sin x_2 & -x_1 \cos x_2 \end{bmatrix} \cdot \begin{bmatrix} -k_1 x_1 - d_1 \cos x_2 - d_2 \sin x_2 \\ -k_2 x_2 + \frac{d_1 \sin x_2}{x_1} - \frac{d_2 \cos x_2}{x_1} \end{bmatrix}
\end{aligned} \tag{5.28}$$

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} d_1 + x_1(k_1 \cos x_2 - k_2 x_2 \sin x_2) \\ d_2 + x_1(k_1 \sin x_2 + k_2 x_2 \cos x_2) \end{bmatrix} \tag{5.29}$$

$$\begin{bmatrix} V_{Qx} \\ V_{Qy} \end{bmatrix} = \begin{bmatrix} V_{Px} + R(k_1 \cos \sigma - k_2 \sigma \sin \sigma) \\ V_{Py} + R(k_1 \sin \sigma + k_2 \sigma \cos \sigma) \end{bmatrix} \tag{5.30}$$

5.3 Adjusting k_1 , k_2 Based on CLOS System

The necessary conditions for alignment of the Quadrotor with helipad, had been represented by equations (5.13) and (5.14).

The direction of Quadrotor velocity vector, α_Q , could be calculated directly from equation (5.30), with adjustable control parameters k_1 and k_2 .

$$\alpha_Q = \arctan \frac{V_{Qy}}{V_{Qx}} = \arctan \frac{V_{Py} + R(k_1 \sin \sigma + k_2 \sigma \cos \sigma)}{V_{Px} + R(k_1 \cos \sigma - k_2 \sigma \sin \sigma)} \tag{5.31}$$

In collision path, α_Q should be equal to α_{QC} .

$$\begin{aligned}
\alpha_Q &= \alpha_{QC} \\
\arctan \frac{V_{Py} + R(k_1 \sin \sigma + k_2 \sigma \cos \sigma)}{V_{Px} + R(k_1 \cos \sigma - k_2 \sigma \sin \sigma)} &= \alpha_{QC} \\
&= \sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right]
\end{aligned} \tag{5.32}$$

In section (5.2) an intermediate control input, u^* , had been chosen to linearize the system. The resultant linearized system would be as follow.

$$\begin{aligned}
\dot{x} &= u^* \\
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} u_1^* \\ u_2^* \end{bmatrix} \\
&= \begin{bmatrix} -k_1 \cdot x_1 \\ -k_2 \cdot x_2 \end{bmatrix}
\end{aligned} \tag{5.33}$$

Replacing the state variables with the real system parameters in (5.33) yields

$$\begin{bmatrix} \dot{R} \\ \dot{\sigma} \end{bmatrix} = \begin{bmatrix} -k_1 \cdot R \\ -k_2 \cdot \sigma \end{bmatrix} \tag{5.34}$$

Inserting the collision condition in (5.34), it could be concluded that k_2 is zero.

$$\left. \begin{array}{l} \dot{\sigma} = 0 \\ \dot{\sigma} = -k_2 \cdot \sigma \end{array} \right\} \Rightarrow k_2 = 0 \tag{5.35}$$

k_1 could be calculated from equation (5.32), by inserting $k_2 = 0$.

$$\begin{aligned}
& \arctan \frac{V_{Py} + R(k_1 \sin \sigma + k_2 \sigma \cos \sigma)}{V_{Px} + R(k_1 \cos \sigma - k_2 \sigma \sin \sigma)} = \alpha_{QC} \\
& \xrightarrow{k_2 \rightarrow 0} \arctan \frac{V_{Py} + Rk_1 \sin \sigma}{V_{Px} + Rk_1 \cos \sigma} = \alpha_{QC}
\end{aligned} \tag{5.36}$$

$$\begin{aligned}
\tan \alpha_{QC} &= \frac{V_{Py} + Rk_1 \sin \sigma}{V_{Px} + Rk_1 \cos \sigma} \\
\Rightarrow k_1 &= \frac{V_{Px} \tan \alpha_{QC} - V_{Py}}{R \sin \sigma - R \cos \sigma \tan \alpha_{QC}}
\end{aligned} \tag{5.37}$$

in which α_{QC} could be substituted from equation (5.14).

$$k_1 = \frac{V_{Px} \cdot \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \right) - V_{Py}}{R \sin \sigma - R \cos \sigma \cdot \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \right)} \tag{5.38}$$

Substituting resultant k_1 and k_2 from equations (5.38) and (5.35) respectively, into the control input of the navigation control system, represented by the equation (5.30), the velocity vector of the Quadrotor could be calculated as follow.

$$\begin{aligned}
V_{Qx} &= V_{Px} + R(k_1 \cos \sigma - k_2 \sigma \sin \sigma) \\
&= V_{Px} + R \cos \sigma \frac{V_{Px} \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \sin(\alpha_P - \sigma) \right] \right) - V_{Py}}{R \sin \sigma - R \cos \sigma \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \sin(\alpha_P - \sigma) \right] \right)} \\
&= \frac{V_{Py} \cos \sigma - V_{Px} \sin \sigma}{\cos \sigma \cdot \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \sin(\alpha_P - \sigma) \right] \right) - \sin \sigma}
\end{aligned} \tag{5.39}$$

$$\begin{aligned}
V_{Qy} &= V_{Py} + R(k_1 \sin \sigma + k_2 \sigma \cos \sigma) \\
&= V_{Py} + R \sin \sigma \frac{V_{Px} \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \sin(\alpha_P - \sigma) \right] \right) - V_{Py}}{R \sin \sigma - R \cos \sigma \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \sin(\alpha_P - \sigma) \right] \right)} \\
&= \frac{\left(V_{Py} \cos \sigma - V_{Px} \sin \sigma \right) \cdot \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \sin(\alpha_P - \sigma) \right] \right)}{\cos \sigma \cdot \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \sin(\alpha_P - \sigma) \right] \right) - \sin \sigma} \quad (5.40)
\end{aligned}$$

It is interesting to mention that the final control inputs of the navigation system, V_{Qx} and V_{Qy} , are independent from the planer range, R . However, to optimize navigation controller, the magnitude of the Quadrotor velocity, $|V_Q|$, should be proportional with the planer range. For instance, it could be chosen as follow.

$$|V_Q| = V_P + 2 \cdot R \quad (5.41)$$

In addition the simulation results reveals that if $|V_P| > \frac{1}{4} \cdot |V_Q|$, the Quadrotor will not be able to align itself with the helipad.

5.4 Special Case Study

In special case of stationary helipad, in which $V_P = 0$, from equation (5.38), k_1 would be undefined.

$$\begin{aligned}
k_1 &= \frac{V_{Px} \cdot \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \right) - V_{Py}}{R \sin \sigma - R \cos \sigma \cdot \tan \left(\sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \right)} \\
\stackrel{V_P=0}{\Rightarrow} k_1 &= \frac{0 \cdot \tan(\sigma + 0) - 0}{R \sin \sigma - R \cos \sigma \cdot \tan(\sigma + 0)} = \frac{0 - 0}{R \sin \sigma - R \sin \sigma} = \frac{0}{0} \quad (5.42)
\end{aligned}$$

and from equations (5.39) and (5.40), both V_{Qx} and V_{Qy} also would be $\frac{0}{0}$ and undefined.

To cope with this problem, for calculation of α_{QC} equation (5.14) should be applied.

$$\begin{aligned}\alpha_{QC} &= \sigma + \arcsin \left[\frac{V_P}{V_Q} \cdot \sin(\alpha_P - \sigma) \right] \\ \xrightarrow{V_P=0} \alpha_{QC} &= \sigma\end{aligned}\tag{5.43}$$

Therefore, in the case of stationary helipad, the Quadrotor should move on the course of line-of-sight angle toward the helipad. The resultant Quadrotor velocity vector would be as follow.

$$\begin{bmatrix} V_{Qx} \\ V_{Qy} \end{bmatrix} = \begin{bmatrix} V_Q \cos \sigma \\ V_Q \sin \sigma \end{bmatrix}\tag{5.44}$$

Chapter 6

Altitude Controller

For altitude control, equation (4.27) at page 18 could be applied, which is the control input, \tilde{u}_3 , for the Quadrotor system.

$$\tilde{u}_3 = K_z \cdot (x_{6d} - x_6) = K_z \cdot (\dot{z}_d - \dot{z}) \quad (6.1)$$

In this equation \dot{z} is the real-time vertical linear speed of the Quadrotor which could be measured by the aids of an appropriate sensor and will be discussed in chapter (9). From the other hand, \dot{z}_d is the vertical component of the desired linear velocity of the Quadrotor which should be designed in a way to keep the Quadrotor to hover and cruise in a desired altitude which depends on the real-time navigation control system requirements. For that reason a proportional controller could be defined as follow.

$$\dot{z}_d = K \cdot (z_d - z) \quad (6.2)$$

in which \dot{z}_d is the vertical component of the desired lateral velocity of the Quadrotor, K is the proportional control factor, z_d is the desired altitude for the Quadrotor, and z is the real-time altitude of the Quadrotor in any given time which could be measured by the aids of an appropriate sensor and will be discussed in chapter (9). The choice of z_d also depends on the states of a behavior-based control system and will be discussed in chapter (8).

It is quite clear that the attitude and altitude of the Quadrotor influence each other and hence the altitude of the Quadrotor is a function of its

attitude. The simulation results will show this fact which will be discussed in chapter (12).

Chapter 7

Onboard Mobile Helipad Model

In this thesis work, the navigation control system should be developed in a way to control Quadrotor to track and land on an onboard mobile helipad. The helipad could be a buoys or ship floating and moving on the water, or a terrestrial mobile robot moving on the ground and therefore, its height above the water or ground could be assumed constant as the helipad has no movement in the z direction. Therefore a two dimensional consideration of the helipad movement in a *xy reference coordinate system* would be enough for formulation and modeling of the helipad.

For that reason, a sinusoidal motion along the $y = x$ line has been chosen and formulated in order to model the mobile helipad.

$$x(t) = x_0 + v_0 \cdot t \quad (7.1)$$

$$y(t) = x(t) + A \sin(f \cdot x(t)) \quad (7.2)$$

$$z(t) = \text{cte.} \quad (7.3)$$

in which, x_0 is the x-component of the initial position of the onboard mobile helipad in the reference coordinate system. This choice will lead the helipad to have a y-component of the initial position equal to $y_0 = x_0 + A \sin(f \cdot x_0)$. In addition, v_0 is the initial velocity, A is the amplitude of the oscillatory component of the path, and f is the frequency representing the rate of changes in path direction of the mobile helipad.

Since $z(t)$ assumed to be constant, therefore, a state space model with two state variables, x_1 and x_2 which are defined by set of equations (7.4)

and (7.5), would be adequately represents the chosen dynamic of the helipad. The state variables and underlying differential equations of the system are as follow.

$$x_1 = x(t) \quad (7.4)$$

$$x_2 = y(t) \quad (7.5)$$

$$\dot{x}_1 = \dot{x} \quad (7.6)$$

$$\dot{x}_2 = \dot{y} = \dot{x} + A \cdot f \cdot \dot{x} \cdot \cos x_1 \quad (7.7)$$

$$\dot{x}_1 = v_{p_0} \quad (7.8)$$

$$\dot{x}_2 = v_{p_0} + A \cdot f \cdot v_{p_0} \cdot \cos x_1 \quad (7.9)$$

Helipad is moving with the constant velocity v_{p_0} and its motion course could be calculated as follow.

$$\alpha_p = \arctan \left(\frac{dy}{dx} \right) \quad (7.10)$$

With dx and dy from equations (7.8) and (7.9), equation (7.10) could be simplified as follow.

$$\begin{aligned} \alpha_p &= \arctan \left(\frac{v_{p_0} + A \cdot f \cdot v_{p_0} \cdot \cos x_1}{v_{p_0}} \right) \\ &= \arctan (1 + A \cdot f \cdot \cos x_1) \end{aligned} \quad (7.11)$$

Chapter 8

Cruise, Align, and Land Control

In order that Quadrotor could be landed on an onboard mobile helipad, two control tasks should be executed in parallel; a real time, low level attitude control task, for stabilization of the Quadrotor which is a time critical task, and a high level navigation control task, consists of cruise, align, and land modes, which is a longterm and less time critical task. Here, for longterm control task, a behavior-based control system has been proposed. At the beginning, the Quadrotor should cruise from any initial position in an optimized safe height above the ground toward the onboard mobile helipad by tracking its motion path. As soon as the helipad become in the field of view of Quadrotr, the Quadrotor should descent and align itself with the center of the helipad and as soon as the alignment reaches to a satisfactory level of accuracy, the Quadrotor will land on the helipad. In this chapter, a three state behavior-based control system for longterm navigation control task will be presented.

8.1 Behavior-Based Finite State Model for Control Task Coordination

The overall navigation and landing task is best described by a simple finite state machine with three states of *cruise*, *align*, and *land* mode. For state transition of the behavior-based control system, an appropriately chosen imaginary 3D semi-conical-semi-cylindrical geometric shape has been defined above and around the helipad. This state transition geometric shape has been depicted in the figure (8.1) in which, the upper conical section is chosen to be the limitation for *align* mode transition, while the lower cylindrical section is chosen to be the limitation for *land* mode transition.

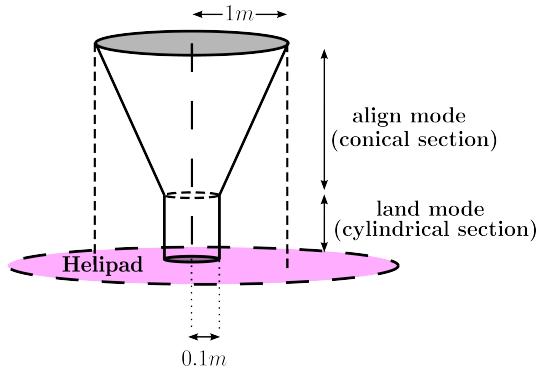


Figure 8.1: 3D imaginary state transition geometric shape on the helipad for behavior-based navigation control system transition.

Initially the Quadrotor is in the *cruise* mode in which the Quadrotor cruises on an optimized safe altitude above the ground, here 2m, in order to track the helipad and enters inside the state transition geometric shape. As soon as the Quadrotor approaches close enough to the helipad and enters inside the upper conical section of the imaginary state transition geometric shape, the behavior-based navigation control system will transit to the *align* mode in which the controller will maneuver the Quadrotor in a way to decent and align with the center of the helipad inside the imaginary conical section. During alignment procedure, if the Quadrotor exits the conical section, the behavior-based navigation control system will transits back to the *cruise* mode and the controller will maneuver the Quadrotor to ascent back to the safe altitude while tracking the helipad in parallel.

Since the helipad is mobile, exact alignment of the Quadrotor with the center of the helipad is very difficult and might never be happen. Therefore, the lower cylindrical section of the state transition geometric shape has been defined as the limitation for *land* mode transition.

Considering the size of the Quadrotor, the radius of the upper circular plane base of the conical section of the imaginary state transition geometric shape has been chosen to be equal to 1m. Also, the radius of the lower intersect of the cylindrical section has been chosen to be 0.1m, as 10cm considered adequately accurate for alignment of mass center of the Quadrotor with center of the helipad in this work.

As discussed in the chapter 5, for navigation of the Quadrotor based on the command to line-of-sight system, as depicted in the figure 5.1 at page 22, the planer distance between Quadrotor and elevated helipad has been applied. In order to detect the pose of the Quadrotor relative to the state transition geometric shape, it is necessary to find the relation between the

altitude and range inside the geometric shape.

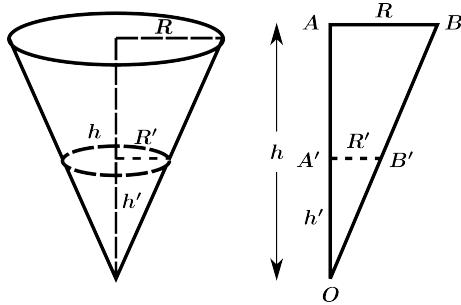


Figure 8.2: Relation between range and altitude inside the imaginary 3D geometric shape.

Referring to figure (8.2) the range and altitude inside the chosen geometric shape could be formulated as follow.

$$\begin{aligned}
 \triangle OAB &\sim \triangle B'A'B \\
 \frac{\overrightarrow{OB}}{\overrightarrow{B'B}} &= \frac{\overrightarrow{OA}}{\overrightarrow{B'A'}} \Rightarrow \frac{h}{h'} = \frac{R}{R'} \\
 \Rightarrow R' &= R \cdot \frac{h}{h'}
 \end{aligned} \tag{8.1}$$

In equation (8.1), h is the altitude of the Quadrotor, R is the planer range, which is the distance between the center of mass of the Quadrotor and the center of the elevated helipad, and $Range$ is the distance between the center of mass of the Quadrotor and the center of the helipad on the ground.

As discussed in the chapter 5, the onboard mobile helipad has a constant altitude above the ground with some disturbances due to vibration of the helipad during its motion. Therefore, the alignment of the Quadrotor and the helipad could be considered as a two dimensional problem in which $z - component$ could be neglected. Therefore, in the above mentioned formulas, R has been used instead of $Range$. This concept has been depicted once again at figure (8.3).

The resulting behavior-based finite state model for control task coordination, consisting of three states, has been depicted in the figure 8.4. Three states of *cruise*, *align*, and *land* mode, have been depicted in figures 8.5, 8.6,

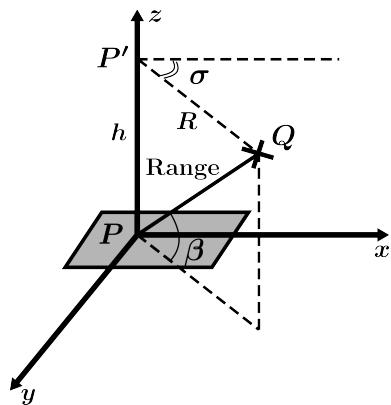


Figure 8.3: Location of the Quadrotor relative to the helipad depends on R and h .

and 8.7.

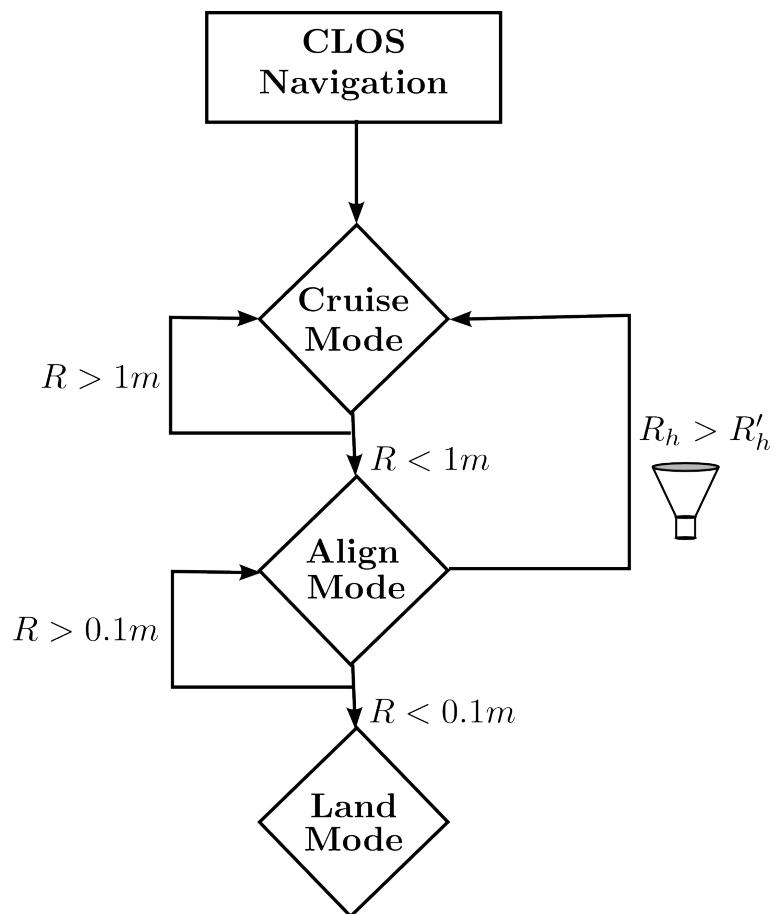


Figure 8.4: State transition diagram for landing task.

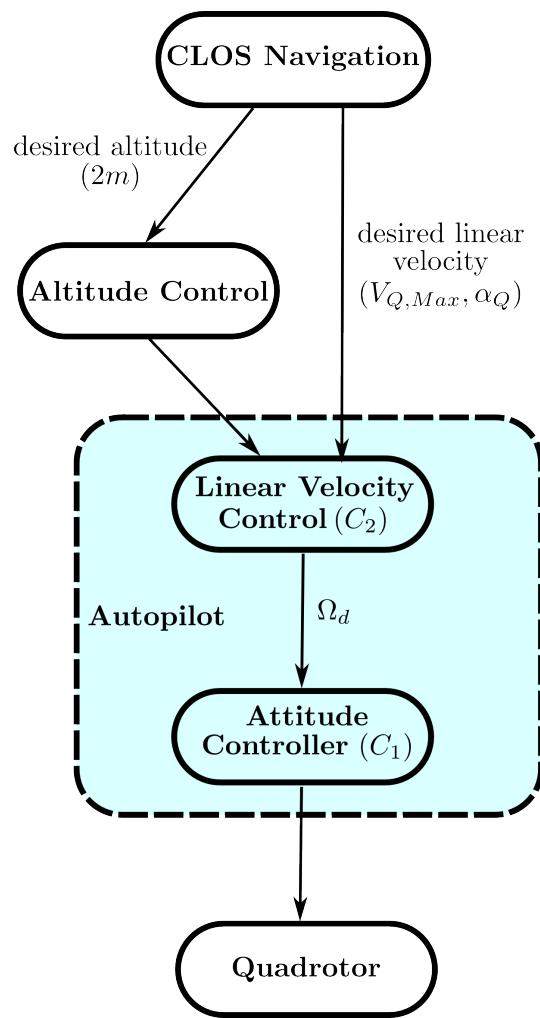


Figure 8.5: Behavior-based cruise control system.

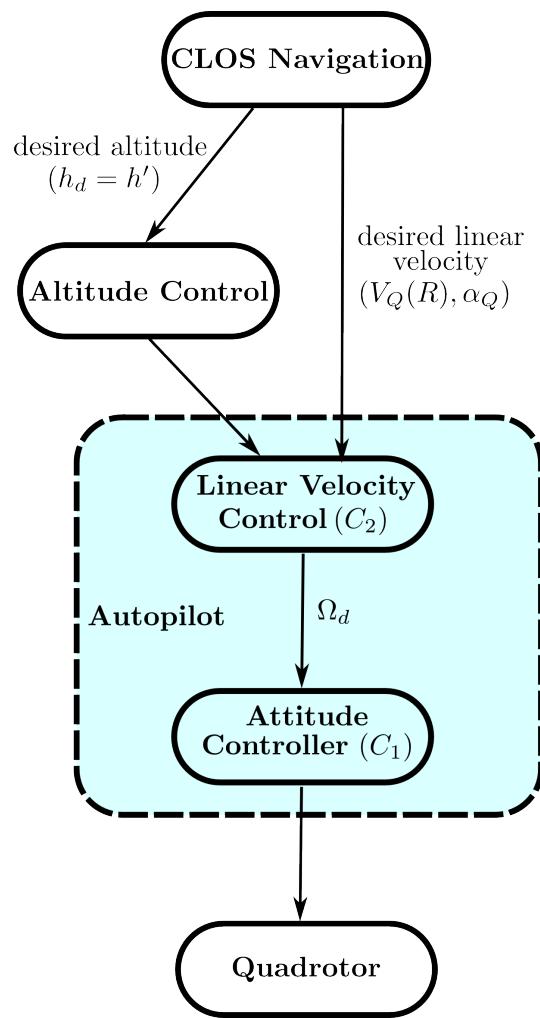


Figure 8.6: Behavior-based align control system.

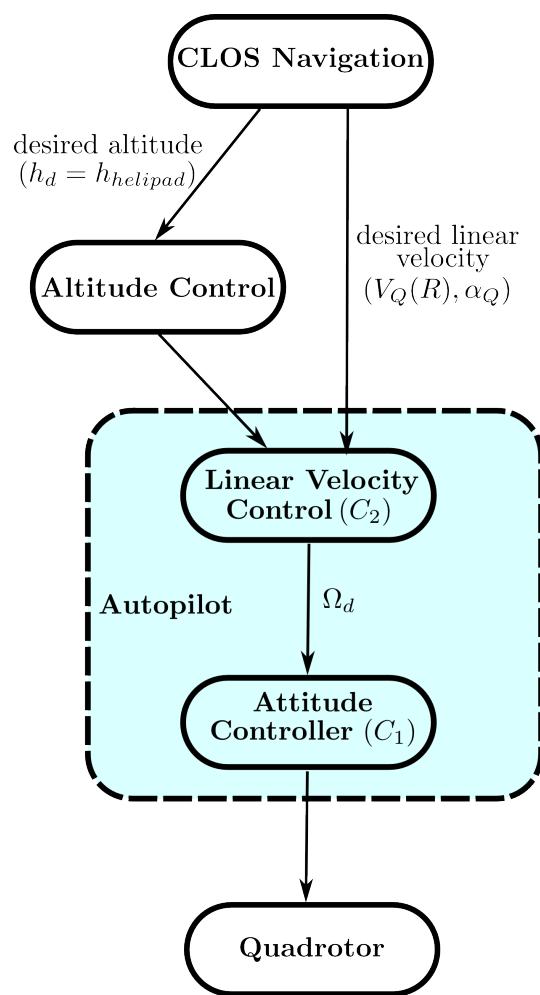


Figure 8.7: Behavior-based land control system.

Chapter 9

Sensor Fusion

9.1 Introduction

In order to realize proposed control algorithms for attitude stabilization and navigation of the Quadrotor based on command to line-of-sight angle, it is essential to be able to measure underlying control parameters of the system by appropriately chosen sensors. In simulation, the Quadrotor and helipad models provides the exact information about the position and orientation of Quadrotor and helipad which leads to a very accurate control results in simulation environment. However, realization and evaluation of the control algorithms in real world implementation will depends highly on the availability and accuracy of measurements of the position and orientation of the Quadrotor and helipad.

In order to propose and choose any sensor for such application, it is necessary to cope with Quadrotor limitations such as small size, maximum payload capacity, and energy resource limitations. In the other words, any sensor choice should be able to measure desired control parameters as accurate as possible, for as minimum size, energy consumption, weight, and cost as possible. Although the main purpose of this thesis work was to develop control algorithms for autonomous navigation and landing of the Quadrotor on a mobile helipad and evaluation of the efficiency of the proposed algorithms in simulation, however, from the beginning, special attention has been paid to realistic choice of the control parameters according to their measurability and availability of the sensor technologies. This chapter attempts to introduce some rough ideas about the choice of sensors for implementation of proposed control algorithms in future works. It is obvious that evaluation of the performance of any proposed method for measurement of variables, require separate research works.

9.2 Control Parameters in CLOS-Based Navigation

In this thesis work, the proposed control algorithm consists of a low-level, real-time attitude controller and a high-level longterm navigation and landing controller based on the command to line-of-sight angle. Therefore, the control parameters are the attitude of the Quadrotor, (ϕ, θ, ψ) , planer distance between Quadrotor and helipad, R , and the line-of-sight angle, σ . While attitude of the Quadrotor could be measured directly by the help of an appropriate gyroscope sensor, which will be discussed in the next section, however, measurement of the R and σ might be indirect which requires more deep consideration.

Figure 9.1 shows the relative position of the Quadrotor and helipad in local NE^1 reference coordinate system.

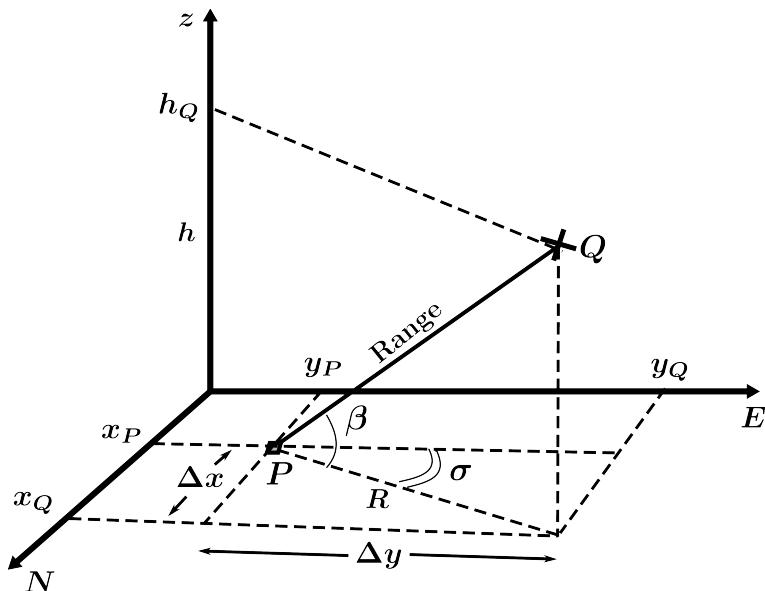


Figure 9.1: Quadrotor and helipad in reference coordinate system.

Any sensor combination which able to measure absolute or relative position of the Quadrotor and helipad in a reference coordinate system, will provides necessary information about R and σ . With information about positions, considering figure (9.1), R and line-of-sight angle could be calculated as follow.

¹North-East

$$R = \sqrt{(x_P - x_Q)^2 + (y_P - y_Q)^2} \quad (9.1)$$

$$\sigma^* = \arctan\left(\frac{|y_P - y_Q|}{|x_P - x_Q|}\right) \quad (9.2)$$

However, the line-of-sight angle contains information about the relative position of the Quadrotor and helipad, which ignored in equation (9.2). The Quadrotor might be located in one of the eight possible sections relative to helipad position. However, since the helipad is on the surface of the ground, and in other words, the Quadrotor always located above the helipad, therefore, eliminating four underground sections, the Quadrotor might be located in one of four possible sections above the helipad. Figure (9.2) shows these four possibilities.

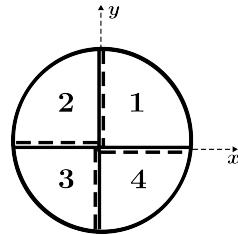


Figure 9.2: Relative position of the Quadrotor and helipad; assuming the helipad on the center and Quadrotor on the circumference of the circle

Considering these four possible orientation of the Quadrotor relative to the helipad, the line-of-sight angle could be recalculated as follow.

$$\sigma = \begin{cases} \sigma^* & \text{Section 1: if } \Delta x > 0, \Delta y \geq 0 \\ \pi - \sigma^* & \text{Section 2: if } \Delta x \leq 0, \Delta y > 0 \\ \pi + \sigma^* & \text{Section 3: if } \Delta x < 0, \Delta y \leq 0 \\ 2\pi - \sigma^* & \text{Section 4: if } \Delta x \geq 0, \Delta y < 0 \end{cases} \quad (9.3)$$

In the following sections, some examples of sensors, which could be suggested for measurement of the required control parameters of this thesis work, are briefly introduced.

9.3 Attitude Sensor

An electronic miniature gyroscope could be used for measuring of the attitude of the Quadrotor. A good example, is MTi from Xsens motion technologies. Figure 9.3 shows this miniature sensor in the local *NE* reference coordinate system. It could provides drift-free 3D orientation as well as calibrated 3D acceleration, 3D rate of turn (rate gyro) and 3D earth-magnetic field data. More information about the characteristics of this sensor is available in [23].

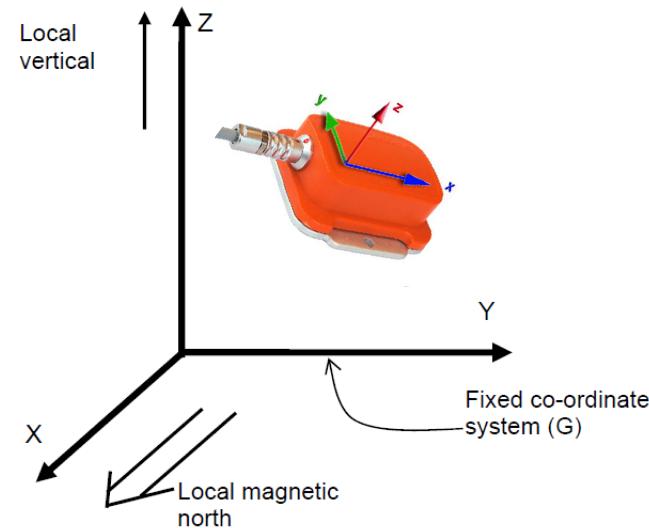


Figure 9.3: MTi in *NE* Reference Coordinate System (Adapted from [23]).

The main drawback of MTi as a sophisticated and miniature attitude sensing device, is its high cost; indeed its capabilities are far more than the needs of this work. In [16] a combination of a low-cost inertial sensor and a pair of CMOS cameras used instead of expensive inertial/GPS sensing system for an autonomous helicopter stabilization.

9.4 Altitude Measurement

In this thesis work, the relative altitude of the Quadrotor above local ground surface is required. GPS is not suitable for this purpose, since it provides absolute altitude in a reference coordinate system, missing the information

about local ground surface elevations.

In [8] a combination of barometric pressure and optic flow measurements are used for estimation of the height above ground. In [48] an *Elementary Motion Detector (EMD)*, has been applied as an optic flow (OF²) measuring sensor and since OF is a constant value, the error between desired OF and measured OF has been used as an estimation of height above ground.

The optic flow concept is applicable only for moving objects and this could be mentioned as the main drawback of this method. In other words, this method could only be used for altitude estimation in the cruise mode, because in align and land mode the Quadrotor might be in hovering position. However, in most application, in align and land mode, exact information about the position of the Quadrotor relative to the helipad is required to be able to control the Quadrotor and therefore, the altitude information could be extracted indirectly from navigation sensors.

Another method has been introduced in [24] in which an ultrasonic range detector has been mounted on the Quadrotor which estimates altitude based on the reflected ultrasonic signal in the Quadrotor. However, materials such as grass, water, or soil have different reflection factors for the ultrasonic signal and the reflected ultrasonic signal is highly affected by the material on the surface of the ground beneath the Quadrotor; moreover, ultrasonic signal is highly noisy which make the estimation of the altitude more difficult.

9.5 Differential GPS

The GPS is a well known and widely used navigation system; however, since in the next sections, a method for navigation will be suggested, which is mainly adapted from GPS system, therefore, the following information about the basic functionalities of the GPS system are cited from [26]. All the information between double quotation marks are cited from [26], unless otherwise declared.

“The Navstar Global Positioning System (GPS) is a space-based radio-positioning and time transfer system. GPS provides accurate position, velocity, and time (PVT) information to an unlimited number of suitably equipped ground, sea, air and space users.”

“GPS comprises three major system segments, Space, Control, and User (see figure (9.4))”

²OF = $\frac{V_x}{h} = \frac{\text{Ground Velocity}}{\text{Height}}$ (Adapted from [48])

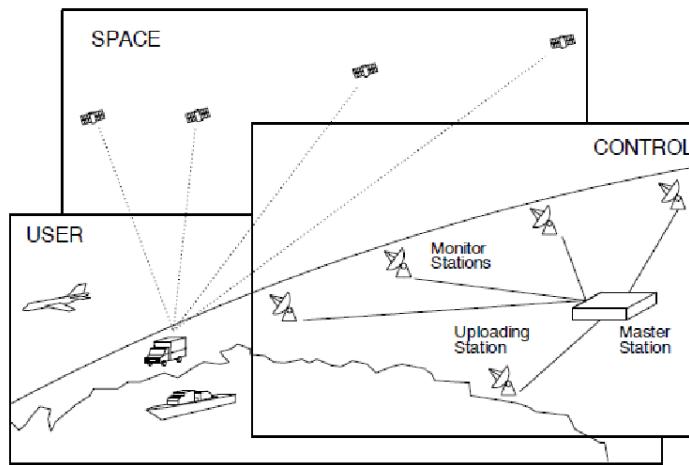


Figure 9.4: The Navstar Global Positioning System consists of three fundamental segments: Space, Control, and User. (Adapted from [38].)

“The absolute three-dimensional location of any GPS receiver is determined through simple trilateration techniques based on time of flight for uniquely coded spread-spectrum radio signals transmitted by the satellites. Precisely measured signal propagation times are converted to pseudoranges representing the line-of-sight distances between the receiver and a number of reference satellites in known orbital positions. The measured distances have to be adjusted for receiver clock offset, hence the term pseudoranges. Knowing the exact distance from the ground receiver to three satellites theoretically allows for calculation of receiver latitude, longitude, and altitude”

“Although conceptually very simple, this design philosophy introduces at least four obvious technical challenges [38]:

- Time synchronization between individual satellites and GPS receivers.
- Precise real-time location of satellite position.
- Accurate measurement of signal propagation time.
- Sufficient signal-to-noise ratio for reliable operation in the presence of interference and possible jamming.”

There are potential error sources for measured pseudoranges in GPS based navigation system, which are listed in the following table.

Error Source	Standard Deviation [m]
Satellite position	3
Ionospheric refraction	5
Tropospheric refraction	2
Multipath reflection	5
Selective availability	30

Table 9.1: Potential error sources for measured pseudoranges (Adapted from [38])

“With the exception of multi-path effects, all of the error sources listed in Table 9.1, can be essentially eliminated through use of a practice known as differential GPS (DGPS). The concept is based on the premise that a second GPS receiver in fairly close proximity (i.e., within 10 km) to the first will experience basically the same error effects when viewing the same reference satellites. If this second receiver is fixed at a precisely surveyed location, its calculated solution can be compared to the known position to generate a composite error vector representative of prevailing conditions in that immediate locale. This differential correction can then be passed to the first receiver to null out the unwanted effects, effectively reducing position error for commercial systems to well under 10 meters [38].”

“Typical DGPS accuracies are around 4 to 6 meters, with better performance seen as the distance between the mobile receivers and the fixed reference station is decreased [38].”

In overall, the main drawback of DGPS navigation system is the fact that it is not suitable for indoor applications and also the inaccuracy of the available commercial versions make them unsuitable for *align and land mode*. DGPS navigation system could be implemented in outdoor *cruise mode*; for *align and land mode* or indoor applications a more accurate navigation system is required. From the other hand, as discussed before, DGPS provides absolute altitude estimation which is not suitable for altitude control task. Finally, it worth to mention that in [49] a GPS based localization with an accuracy of $1 - 2[m]$ has been introduced.

9.6 On-board Camera

One of the most popular methods for localization and navigation of UAV, especially in autonomous landing phase, is a vision based system, in which a simple and light camera, most often by the help of gimbals, mount on the UAV. In the landing phase the camera searches for a predefined marker attached to the landing target and an image processing algorithm extracts the relative position and orientation of the target and UAV.

However, this method has some limitations and disadvantages; first, the target should be in the field of view of the camera. Moreover, extracting image information out of a mobile target is a very difficult task, if not impossible, and in most of the research works vision based sensor has been applied as a complementary sensor along with GPS or other methods for detecting mobile objects. In addition The illumination and weather conditions are other limitations for the vision based systems.

In chapter 2, some of the related research works has been introduced.

9.7 Time Of Flight Camera

Another method for localization of the relative position and orientation of the Quadrotor and helipad, could be the time of flight cameras. In [4] a typical time of flight camera, called PMD³, has been introduced. The key component is an array or line sensor which can measure the distance to the target pixelwise in parallel without scanning. The sensors consist of smart pixels, called the Photonic Mixer Device (PMD) which enables fast optical sensing and demodulation of incoherent light signals in one component [4]. However, since this kind of camera is relatively heavy and consumes high energy, it is not suitable to be mounted on the Quadrotor; instead it could be mounted on the helipad in ground and as soon as the Quadrotor approaches to the field of view of this camera, it could detect the position and orientation of the Quadrotor relative to the helipad and these information could be sent to the Quadrotor by the help of a radio communication module.

9.8 Intensity Sensor

Another idea for localization of the Quadrotor relative to the helipad is to use intensity sensor. The main idea is to mount a signal source, such as infrared, on the helipad and an intensity sensor on the Quadrotor and let the intensity sensor to search and maneuver the Quadrotor toward the signal source. This is a rough idea which requires a research study to be evaluated

³Photonic Mixer Device

and realized.

9.9 Ultrasonic Based Localization System

Another idea for localization, is to develop a local system similar to the universal GPS. The main idea is to mount several ultrasonic sources in well defined positions on the helipad, similar to the satellites network in the GPS system. In addition, a radio signal source will be mounted on the helipad for synchronization and triggering purpose. All ultrasonic sources are synchronized and triggered by the radio signal source circuit and transmit their signals simultaneously with different frequencies.

An ultrasonic receiver and a radio signal receiver are mounted on the Quadrotor. The radio signal travels with the speed of light and the receiver in the Quadrotor will receive this signal almost simultaneously. However, the ultrasonic signal travels with the speed of sound and since the sources on the helipad are mounted in different positions, in any given time they have different distances with the Quadrotor and the receiver in the Quadrotor will receive them separately with some delay which is proportional to the distance from their sources on the helipad, similar to the pseudoranges in GPS system.

As soon as the radio signal received by the Quadrotor, a timer will be actuated and will keep the time delay of ultrasonic signals. Since the ultrasonic signals have different frequencies, by reverse Fourier transformation they could be separated and detected and hence pseudoranges could be measured. In the next step, with trilateration similar to the GPS system, the position and orientation of the Quadrotor relative to the helipad could be calculated.

Figure 9.5 and 9.6 show this idea.

As depicted in the figure 9.5, four ultrasonic sensors are mounted in the well defined positions on the helipad, $US_1(0, 0, 0)$, $US_2(a, b, 0)$, $US_3(a, -b, 0)$, and $US_4(-c, 0, 0)$, with unknown position of the Quadrotor in the helipad reference coordinate system, (x, y, z) . The reference coordinate system is located at the center of the helipad with x-axis toward the local North, y-axis toward local East and z-axis in vertical upward direction. t_1 , t_2 , t_3 , and t_4 are the measurable times of flight of the ultrasonic signals from helipad to the receiver, triggered by the radio signal and separated by reverse Fourier transformation.

Figure 9.6 shows the concept of pseudoranges. The underlying trilateration for calculation of the position and orientation of the Quadrotor relative to the helipad, are as follow.

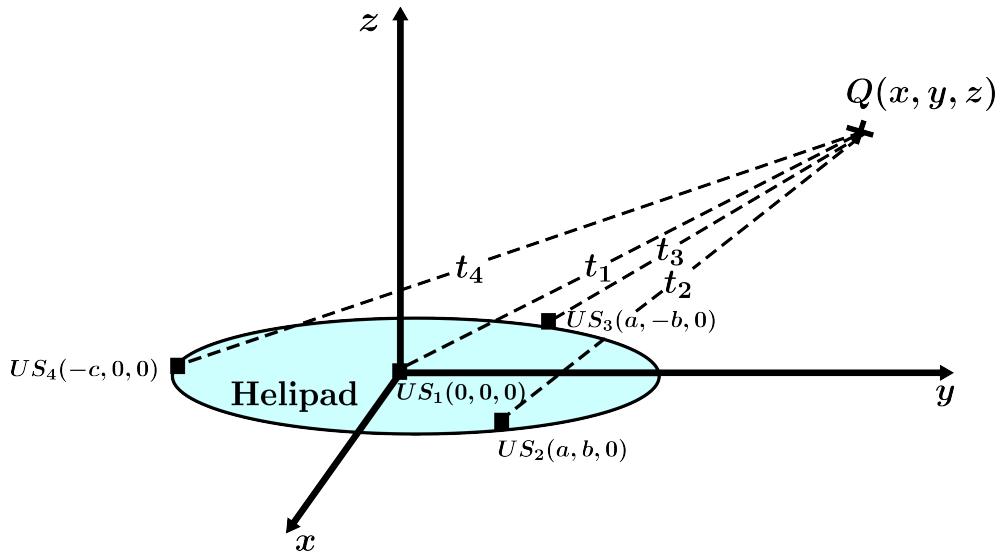


Figure 9.5: A localization system based on ultrasonic signal, adapted from GPS system.

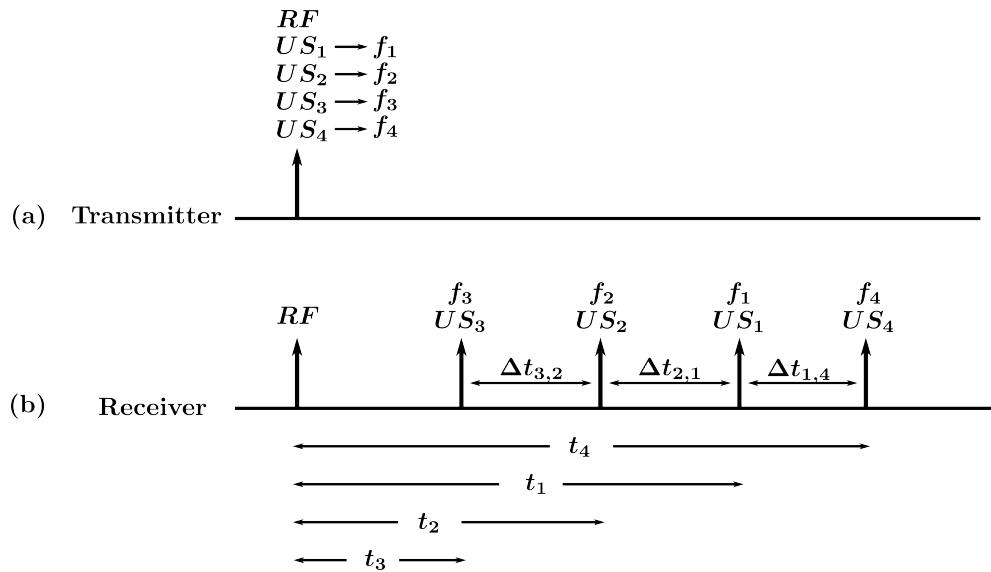


Figure 9.6: Measurement of pseudoranges; (a) radio signals and all ultrasonic signals with different frequencies transmitted simultaneously form helipad, (b) on Quadrotor, radio signal will received simultaneously, while ultrasonic signals will be received with some delay proportional to the distance (pseudoranges)

$$D_1 = D_{Q,US_1} = t_1 \cdot V_s \quad (9.4)$$

$$D_2 = D_{Q,US_2} = t_2 \cdot V_s = (t_1 - \Delta t_{2,1}) \cdot V_s \quad (9.5)$$

$$D_3 = D_{Q,US_3} = t_3 \cdot V_s = (t_1 - (\Delta t_{2,1} + \Delta t_{3,2})) \cdot V_s \quad (9.6)$$

$$D_4 = D_{Q,US_4} = t_4 \cdot V_s = (t_1 + \Delta t_{1,4}) \cdot V_s \quad (9.7)$$

Defining $\Delta_2 = \Delta t_{2,1} \cdot V_s$, $\Delta_3 = (\Delta t_{2,1} + \Delta t_{3,2}) \cdot V_s$, and $\Delta_4 = \Delta t_{1,4} \cdot V_s$ and substituting in the equations will give:

$$D_1 = d \quad (9.8)$$

$$D_2 = d - \Delta t_{2,1} \cdot V_s = d - \Delta_2 \quad (9.9)$$

$$D_3 = d - (\Delta t_{2,1} + \Delta t_{3,2}) \cdot V_s = d - \Delta_3 \quad (9.10)$$

$$D_4 = d + \Delta t_{1,4} \cdot V_s = d + \Delta_4 \quad (9.11)$$

$$D_1 = d = \sqrt{x^2 + y^2 + z^2} \quad (9.12)$$

$$D_2 = d - \Delta_2 = \sqrt{(x - a)^2 + (y - b)^2 + z^2} \quad (9.13)$$

$$D_3 = d - \Delta_3 = \sqrt{(x - a)^2 + (y + b)^2 + z^2} \quad (9.14)$$

$$D_4 = d + \Delta_4 = \sqrt{(x + c)^2 + y^2 + z^2} \quad (9.15)$$

In equations (9.12), (9.12), (9.12), and (9.12), the distance between Quadrotor and center of the helipad, d , and also the position of the Quadrotor, (x, y, z) , are unknown variables which could be calculated from four equations. Notice that a , b , and c are the position of the ultrasonic sources and are known parameters.

However, this is merely a rough suggestion and the realization and evaluation of the efficiency of the proposed method requires careful research study. A possible drawback for such constellation might be the wind effect as disturbance for ultrasonic signals. In addition, Fourier transformation requires lots of calculation power; therefore, it might be better that such calculation done in a central on ground computation unit and the calculated position and orientation information could be sent back to the Quadrotor.

Chapter 10

Modeling of the Wind as a Disturbance to the System

In [30], Etele comprehensively studied the available methods for quantifying a wind gust in an attempt to use this quantification for prediction of wind effect on UAV stability and to provide a realistic understanding of the environmental issues associated with UAV operations in urban and mountainous environments. Also, In [50] and [45], a method for target path tracking of a UAV in wind has been proposed.

Wind has a very significant nonlinear effect on the navigation of small UAVs and it could strongly affects the inertial orientation and rates of the vehicle. Similar operating conditions apply to some large high altitude UAVs that may be exposed to jet-stream effects.

In this work, it has been assumed that in a two dimension NE-coordinate system ¹, the wind could adequately be represented by a velocity vector, with a magnitude equal to the wind velocity, V_w , and a direction equal to the wind direction, ψ_w . The strength of the wind, which represented by its velocity, is a variable quantity with a high rate of variation; however, its direction during a finite time interval could be assumed constant. For instance, during a time interval of few hours, the wind almost belows in the North-East direction with the angle of 30° from the East axis reference, but its velocity could varies dramatically during this time interval. Fig. (10.1) depicts a vector in the NE coordinate system representing the wind.

Although a more detailed study of the wind effect on the navigation control system requires a more comprehensive and accurate model for the wind, however, above mentioned method would be enough for a basic un-

¹North-East reference coordinate system, in which, if we stand toward the local North, our right hand should be in the direction of local East

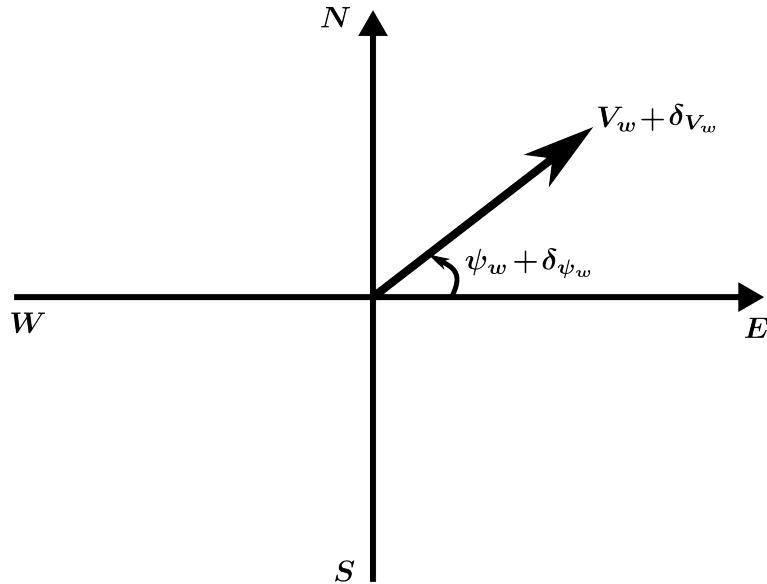


Figure 10.1: Representation of the wind in the local North-East coordinate system.

derstanding of the potential wind effect.

In such constellation, the wind could be represented by equations (10.1) and (10.2) which could be implemented in the simulation model.

$$V_{wx} = V_{wE} = V_w \cdot \cos \psi_w \quad (10.1)$$

$$V_{wy} = V_{wN} = V_w \cdot \sin \psi_w \quad (10.2)$$

Assuming the initial velocity of the wind to be V_{w0} and its initial direction to be ψ_{w0} , the wind effect could be modeled as follow.

$$V_{wx} = V_{wE} = (V_{w0} + \delta_v) \cdot \cos(\psi_w + \delta_\psi) \quad (10.3)$$

$$V_{wy} = V_{wN} = (V_{w0} + \delta_v) \cdot \sin(\psi_w + \delta_\psi) \quad (10.4)$$

in which δ_v is the variation in the wind strength, and δ_ψ is variation in the wind direction. δ_ψ has a very small value in the range of few degrees, as the wind direction is almost constant, while δ_v has a bigger value.

The wind effect acts as disturbances to the Quadrotor model and could be implemented in the simulation model by addition of the wind velocity vector to the first two sentences of differential equations (3.9) at page 12.

$$\dot{x} = \dot{x}_1 = x_4 + V_{wx} \quad (10.5)$$

$$\dot{y} = \dot{x}_2 = x_5 + V_{wy} \quad (10.6)$$

Replacing equations (10.5) and (10.6) in the differential equation sets (3.9), the new differential equations of the Quadrotor, including the wind components as disturbance to the system will be as follow.

$$\dot{\mathbf{X}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = \begin{cases} \dot{x}_1 = x_4 + V_{wx} \\ \dot{x}_2 = x_5 + V_{wy} \\ \dot{x}_3 = x_6 \\ \dot{x}_4 = -(cosx_7sinx_8cosx_9 + sinx_7sinx_9) \cdot \frac{u_1}{m} \\ \dot{x}_5 = -(cosx_7sinx_8sinx_9 + sinx_7cosx_9) \cdot \frac{u_1}{m} \\ \dot{x}_6 = g - (cosx_7cosx_8) \cdot \frac{u_1}{m} \\ \dot{x}_7 = x_{10} \\ \dot{x}_8 = x_{11} \\ \dot{x}_9 = x_{12} \\ \dot{x}_{10} = x_{11}x_{12}I_1 - \frac{I_R}{I_x}x_{11}g(u) + \frac{L}{I_x}u_2 \\ \dot{x}_{11} = x_{10}x_{12}I_2 - \frac{I_R}{I_y}x_{10}g(u) + \frac{L}{I_y}u_3 \\ \dot{x}_{12} = x_{10}x_{11}I_3 + \frac{L}{I_z}u_4 \end{cases} \quad (10.7)$$

Since the wind as disturbance implemented in the differential equations of the Quadrotor, it would affects both attitude and linear speed of the Quadrotor.

Chapter 11

Simulation Model

In the previous chapters, a behavior-based navigation control system for landing of the Quadrotor on a mobile helipad has been developed. It is necessary to evaluate the efficiency of the proposed control system by simulation. Because of the high flexibilities and capabilities of MATLAB and Simulink, this powerful toolbox has been chosen for simulation and evaluation of the overall system. For that reason, a Simulink model of the overall scenario, consisting of several individual models for Quadrotor, autopilot, navigation control system, and helipad, have been developed. In the following sections, the approach for development of these individual models have been described; the source codes of the relative *S-Functions* and *m-files* are attached to the appendices.

11.1 Overall System Model

The overall Simulink model consists of several individual blocks which includes Quadrotor, autopilot, onboard mobile helipad, wind, CLOS navigation control system, and altitude control system model. Information about the real time altitude of the Quadrotor, planer range, and line-of-sight angle have been extracted from the appropriate blocks and calculated within two separate blocks, which labeled as altitude and range sensors. A land terminator block also included to terminate the simulation after that the Quadrotor landed on the helipad. This block is necessary to prevent simulator to resume the navigation task by transiting the navigation control system back to the align or even cruise mode after the time in which the Quadrotor lands on the onboard mobile helipad for the first time. Without this block, the Quadrotor will stop at the first landing position, while the helipad will continue to go away from initial land position and therefore, the simulator will resume the navigation by transiting the control system back to the align mode and an undesirable bouncing effect will be appear

in the simulation results in which the Quadrotor lands several times on the helipad.

The overall block diagram of the Simulink model has been depicted in figure (11.1).

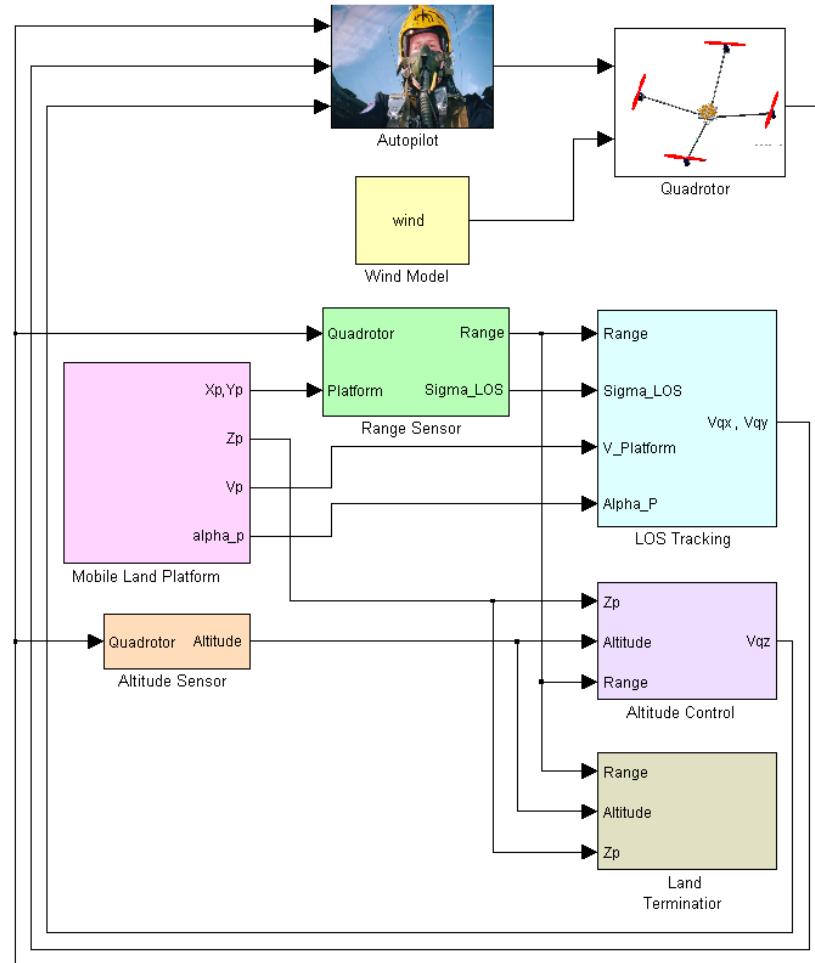


Figure 11.1: The model of the overall scenario in Simulink

11.2 Quadrotor Model

The underlying differential equations representing the Quadrotor system has been discussed extensively in chapter 3. In addition a more comprehensive differential equations including the wind velocity vector as disturbance effect for the Quadrotor system, has been introduced in equation (10.7) at page

60 in chapter 10.

The Quadrotor block in Simulink model consists of a S-Function with 6 inputs and 12 outputs. Inputs consists of control inputs of the Quadrotor, u_1, u_2, u_3, u_4 , and also the disturbance components of the wind, V_{wx} and V_{wy} . The outputs are 12 states of the Quadrotor, $x_1 = x, x_2 = y, x_3 = z, x_4 = \dot{x}, x_5 = \dot{y}, x_6 = \dot{z}, x_7 = \phi, x_8 = \theta, x_9 = \psi, x_{10} = \dot{\phi}, x_{11} = \dot{\theta}$, and $x_{12} = \dot{\psi}$.

The source code for the S-Function is given at appendix A.

11.3 Autopilot Model

A step by step autopilot system development has been described extensively in chapter 4. The essential equations for development of a Simulink model for autopilot are equations (3.7), (4.27) with appropriately chosen coefficients K_x, K_y , and K_z , equation sets (4.32) to calculate desired parameters, u_1, ϕ_d, θ_d , equations (4.21), (4.23), and (4.25) for calculating a_2, a_3, a_4 , and finally equations (4.14), (4.15), and (4.16) to calculate control inputs u_2, u_3 , and u_4 .

The autopilot block in Simulink model consists of a m-file function with 15 inputs and 6 outputs. Inputs consists of 12 states of Quadrotor, 2 desired linear velocities, V_{dx} and V_{dy} from CLOS navigation control system, and vertical component of the desired linear velocity, V_{dz} , from altitude controller. The outputs are 4 control inputs of the Quadrotor, u_1, u_2, u_3 , and u_4 , and also the desired roll and pitch angles which are needed to be saved in workspace of the Matlab and will be used to evaluate the attitude control system.

The source code for corresponding m-file is given at appendix B.

11.4 CLOS Navigation Control System Model

The CLOS navigation control system model has been described extensively in chapter 5. The underlying equations for development of a Simulink model for CLOS navigation control system are equations (5.39) and (5.40) and also equation set (5.44) for considering the special case of stationary helipad. In equations (5.39), and (5.40) especial attention should be paid in order to prevent the denominator of both equations to be zero.

The line-of-sight tracking block in Simulink model consists of a m-file with 4 inputs and 2 outputs. Inputs are range, R , line-of-sight angle, σ , and

helipad speed and direction, V_p , and α_p . The outputs are x and y components of the desired linear velocities of the Quadrotor, V_{Qx} , and V_{Qy} . The source code of the m-file is given at appendix C.

11.5 Altitude Control Model

The altitude control system has been described in details in chapter 6. The underlying equations for development of a Simulink model for altitude controller is the equation (6.2). In addition, depending on the longterm controller state, which is one of the *cruise*, *align*, or *land* mode and also depending on the position of the Quadrotor relative to the onboard mobile helipad, the desired altitude of the Quadrotor above the ground is either a predefined safe altitude or it could be derived out of cone equation (8.1).

The altitude control block in Simulink model consists of a m-file with 3 inputs and 1 output. Inputs are range, R , altitude of the Quadrotor, and altitude of the helipad above the ground. The output is the vertical component of the desired linear velocity of the Quadrotor.

The source code of m-file is given at appendix D.

11.6 On-board Mobile Helipad Model

The onboard mobile helipad model has been described in details in chapter 7. The underlying equations for development of a Simulink model for altitude control system are equations (7.8) and (7.9). Since the altitude of the helipad above the ground is constant, with some disturbances due to trembling of the helipad during its motion, therefore, in Simulink model a random value generator with a value limited to a factor of 1cm, could be implemented to represent the disturbances in the altitude of the onboard mobile helipad.

The platform block in Simulink model consists of a S-Function which only generate 5 outputs without any input variable. These outputs are the planer position of the mobile onboard mobile helipad in a reference NE-coordinate system, x_p and y_p , the real time altitude of the helipad above the ground, and also, the velocity, V_p , and direction of movement of the helipad, α_p .

The source code of S-Function is given at appendix E.

11.7 Wind Model

A typical wind model has been introduced in chapter 10. The underlying equations for development of a Simulink model based on such model are equations (10.3) and (10.4).

The wind block in Simulink model consists of a S-Function which only generate 2 outputs, V_{wx} and V_{wy} .

The source code of S-Function is given at appendix F.

11.8 Range and Altitude Sensor Model

Calculation of the range, R , and line-of-sight angle, σ , based on the information on the position of the Quadrotor and mobile onboard mobile helipad in a reference coordinate system are given by equation (9.1) and (9.3).

The altitude sensor block in Simulink model extracts the third state out of the Quadrotor states, which is the real time altitude of the Quadrotor, $x_3 = z$. The Range sensor block in Simulink model consists of a m-file for calculation of the range and line-of-sight angle based on the equations (9.1) and (9.3). This block has 4 inputs which are the real time position of the Quadrotor and helipad in the NE reference coordinate system, (x_Q, y_Q) and (x_p, y_p) , and it generates 2 outputs, which are range, R , and line-of-sight angle, σ .

The source code for this m-file is given at appendix G.

11.9 Land Terminator Block

As discussed before, the reason for including a land terminator in the overall system is to prevent simulator to run infinitely and to make Simulink to stop after the Quadrotor lands for the first time on the helipad.

This block consists of several sub-blocks which have been depicted in the figure 11.2. In this work, the minimum acceptable alignment accuracy has been defined to be $0.1cm$, which is the diameter of the semi-cylindrical section of the state transition geometric shape, which had been depicted in figure 8.1 at page 40.

The land terminator block has been depicted in the figure 11.2.

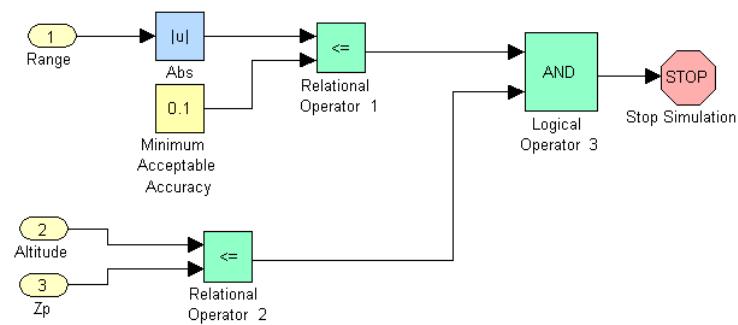


Figure 11.2: Land terminator block in Simulink

Chapter 12

Simulation Results

In this chapter the efficiency of the proposed control algorithms will be evaluated by the help of simulation. The navigation control system should be optimized in order to be able to maneuver the Quadrotor to land autonomously on the helipad in different situations; however, since the helipad is a moving object, this control task will depend also on the characteristics of the motional path of the helipad such as speed and rate of changes of path direction. In other words, if the speed and/or rate of changes in path direction of the mobile helipad are bigger than a threshold value, the Quadrotor will not be able to track and align itself with the helipad.

In the following sections, simulation results for three different case have been discussed. At the first experiment, due to high rate of changes in the path direction of the helipad, the Quadrotor fails to land on the mobile helipad. In the second experiment, the Quadrotor successfully lands on the mobile helipad which is moving with the same speed as previous experiment, but slightly reduced rate of changes in path direction. In the last experiment, the special case of stationary helipad has been studied.

12.1 Experiment 1: High Rate of Changes in Path Direction of Helipad

In this experiment, the Quadrotor initially located at the origin of the reference coordinate system, $(x_{Q0}, y_{Q0}, z_{Q0}) = (0, 0, 0)$, with initial orientation of $(\phi_{Q0}, \theta_{Q0}, \psi_{Q0}) = (0, 0, 0)$. The mobile helipad initially located at the position of $(x_{p0}, y_{p0}, z_{p0}) = (5m, 5m, 0.1m)$; the height of helipad above the ground assumed to be constant, 10cm, while it is moving in a planer sinusoidal path which characterized as follow.

$$y_p(x(t)) = x_p(t) + 2 \sin(2 \cdot x_p(t)) \quad (12.1)$$

$$x_p(t) = 0.5t \quad (12.2)$$

From (12.2) it could be concluded that the speed of helipad is $0.5m/s$, and from (12.1) it could be concluded that the motion is on a sinusoidal path, with a rate of changes in direction equal to $f = 2Hz$, which is the frequency of the sinusoidal part.

In addition, in simulation, it has been assumed that wind blows with a maximum speed of $0.2m/s$ in the direction of the local East to West, which could be formulated as follow.

$$V_w = 0.2 \cdot \sin(t) \quad (12.3)$$

$$\psi_w = 180^\circ \quad (12.4)$$

In Simulink the simulation time has been set to $25s$. The simulation results have been depicted in the figures 12.1, 12.2, 12.3, and 12.4.

Figure 12.1 shows the performance of the navigation control system from a top view in two dimensional NE-coordinate system, in which the Quadrotor takes off from its initial position and follows the sinusoidal path of the helipad. In this figure, diagram (a) reveals that the Quadrotor path coincide with the path of the helipad and from this perspective, the navigation control system has a good performance; however, since the rate of changes in path direction of the helipad is too big, as depicted in diagram (b), the behavior-based navigation control system couldn't transit to land mode and at the end of simulation time, the Quadrotor lagged at $0.54m$ behind the helipad.

Figure 12.2 shows the altitude and planer range diagrams to evaluate the performance of the navigation control system in align and land modes. Diagram (d) reveals that the behavior-based control system transits to the align mode several times as Quadrotor enters inside the state transition geometric shape, which indicated by the green dotted boxes on the diagram (d); however, since the rate of changes in path direction of helipad is too big, the control system quickly transits back to the cruise mode, as the Quadrotor exits the state transition geometric shape. This fact could be revealed more

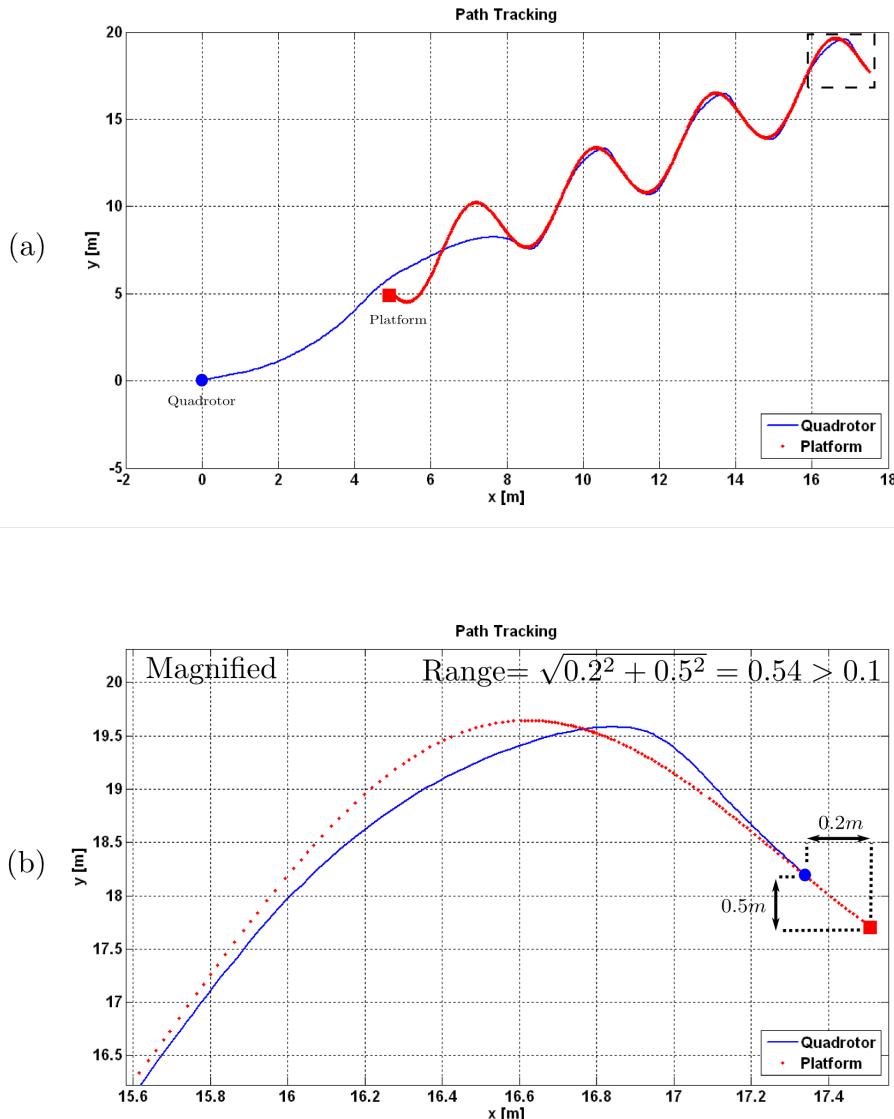


Figure 12.1: Performance of the navigation control system for a big rate of changes in path direction, *frequency* = 2Hz; (a) top view of scenario during simulation time of 25s, (b) magnification of the last few seconds of path tracking

clearly from diagram (c) in which the altitude start to fluctuate as soon as Quadrrotor enters and exits the state transition geometric shape.

Figures 12.3 and 12.4 depict the performance of the autopilot in roll and pitch angles control. In diagram (c) of both figures, the dotted red curva-

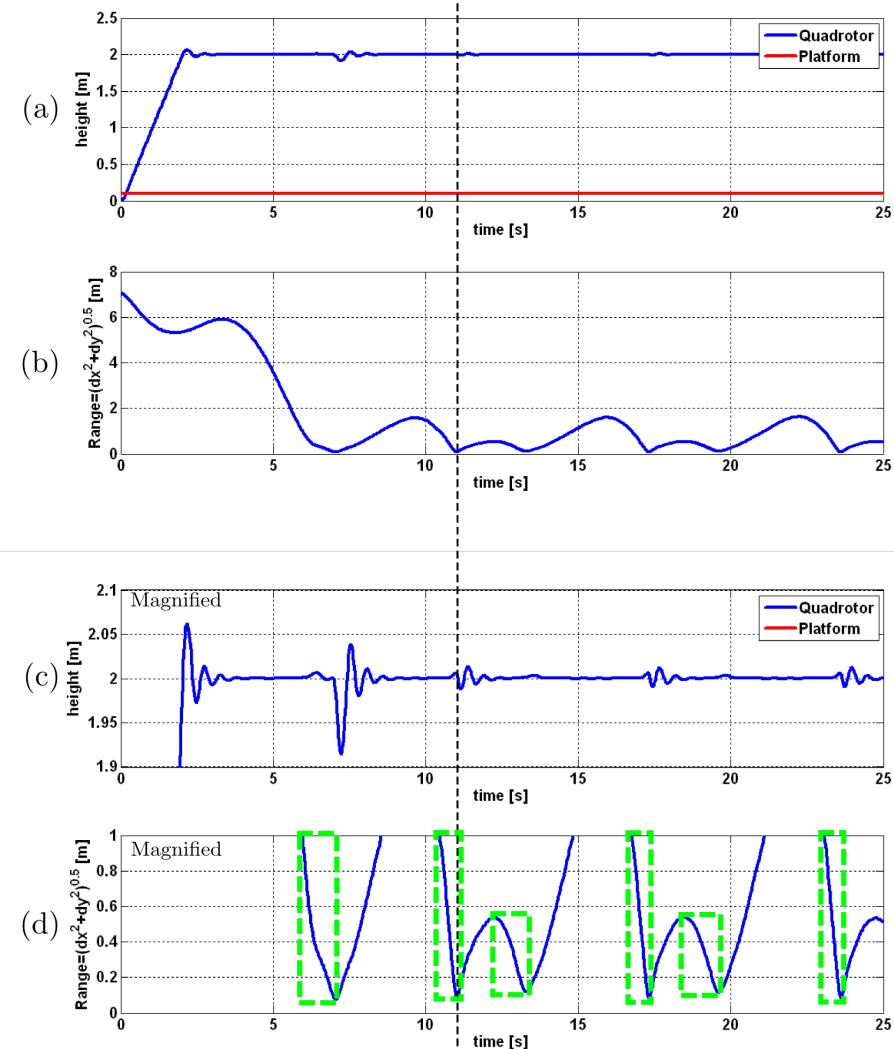


Figure 12.2: Performance of the behavior-based navigation control system in align and land mode for a big rate of changes in path direction, $frequency = 2Hz$; (a) altitude of the Quadrotor during 25s of simulation, (b) planer range between Quadrotor and helipad during 25s of simulation, (c) altitude fluctuation caused by alternative transitions of the control system to align mode and back to cruise mode, (d) alternative entering and exiting of Quadrotor inside and outside of the alignment geometric shapes, indicated by the green dotted boxes

tures show the desired roll and pitch angles, which are the output of the block C_2 of the autopilot as discussed in chapter 4, and the blue curvatures show the real-time roll and pitch angles, which could be measured by the

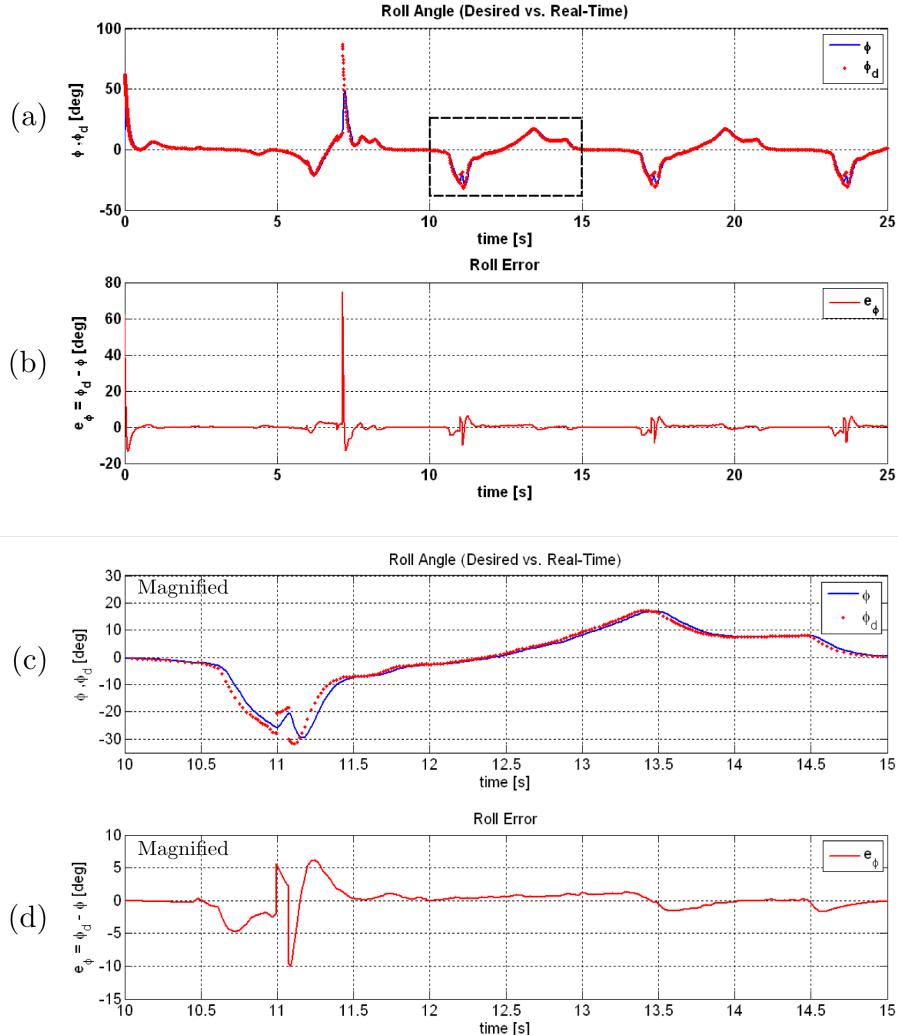


Figure 12.3: Attitude controller performance for a big rate of changes in path direction, (a) roll angle control performance, (b) roll angle error, (c) magnification of the roll angle control, (d) magnification of the roll angle error

aids of an appropriate sensor in the Quadrotor. The diagram (d) of both figures, show the roll and pitch errors. These diagrams reveal that the errors are proportional with the magnitude of the desired angles. In other words, for bigger desired angles, in which the Quadrotor bears bigger turns, the errors are bigger. In addition, it could be concluded that the autopilot has relatively fast and good enough performance, as it is able to follow the control commands almost simultaneously.

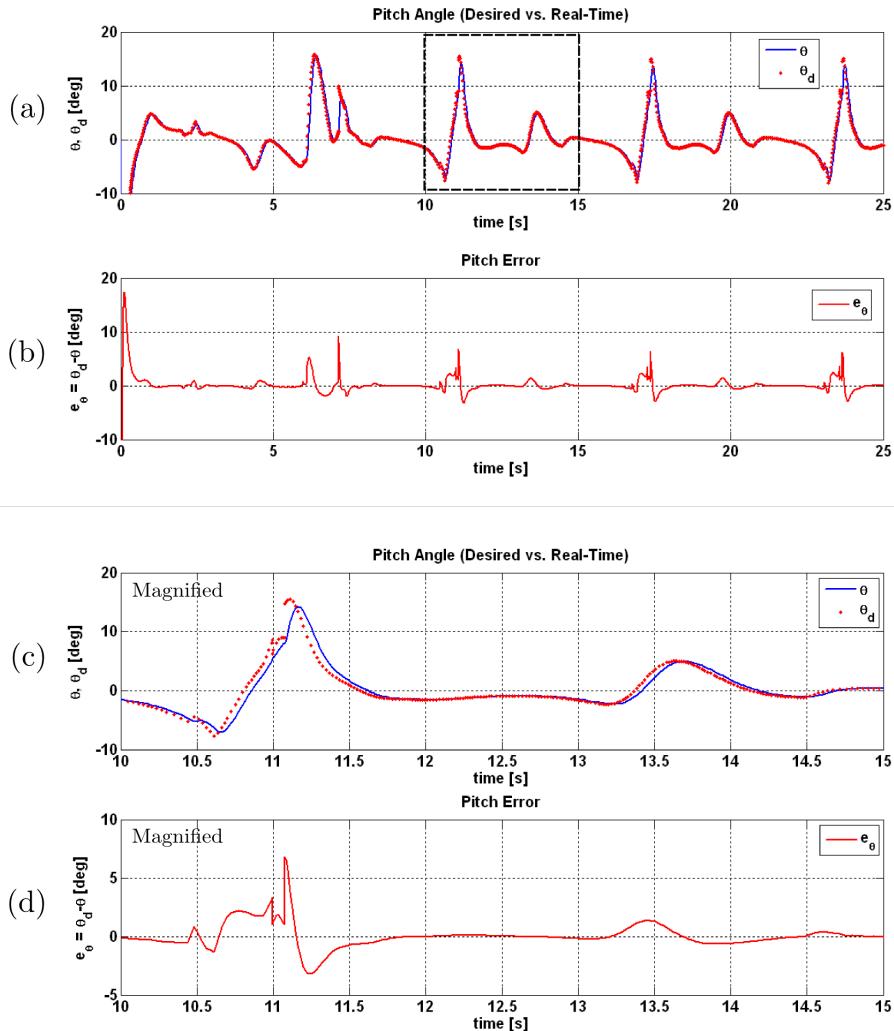


Figure 12.4: Attitude controller performance for a big rate of changes in path direction, (a) pitch angle control performance, (b) pitch angle error, (c) magnification of the pitch angle control, (d) magnification of the pitch angle error

12.2 Experiment 2: Reduced Rate of Changes in Path Direction of Helipad

This experiment is similar to the previous one with reduced rate of changes in path direction for helipad; the Quadrotor initially located at the origin of the reference coordinate system, $(x_{Q0}, y_{Q0}, z_{Q0}) = (0, 0, 0)$, with initial orientation of $(\phi_{Q0}, \theta_{Q0}, \psi_{Q0}) = (0, 0, 0)$ and the mobile helipad initially

located at the position of $(x_{p0}, y_{p0}, z_{p0}) = (5m, 5m, 0.1m)$; the height of helipad above the ground also assumed to be constant, $10cm$, while it is moving in a planer sinusoidal path which characterized as follow.

$$y_p(x(t)) = x(t) + 2 \sin(1 \cdot x(t)) \quad (12.5)$$

$$x_p(t) = 0.5t \quad (12.6)$$

From (12.6) it could be concluded that the speed of helipad is $0.5m/s$, as before, and from (12.5) it could be concluded that the motion is on a sinusoidal path. However, in this experiment the rate of changes in path direction is reduced to $f = 1Hz$.

The wind condition is similar to the previous experiment. It has been assumed that the wind blows with a maximum speed of $0.2m/s$ in the direction of the local East to West. The formulation will be similar to 12.3 and 12.4.

The simulation time also has been set to $25s$. The new simulation results have been depicted in the figures 12.5, 12.6, 12.7, and 12.8.

Figure 12.5 shows the performance of the navigation control system from top view in a two dimensional NE coordinate system, in which the Quadrotor takes off from its initial position and follows the sinusoidal path of the helipad. In this figure, once again diagram (a) reveals that the Quadrotor path coincide with the path of the helipad; from the other hand, as depicted in diagram (b), as soon as the Quadrotor enters inside the state transition geometric shape, it could be aligned with the helipad and therefore, the behavior-based navigation control system transits to the land mode and the Quadrotor lands on the helipad with a very good accuracy of $5cm$.

Figure 12.6 shows the altitude and planer range diagrams to evaluate the performance of the navigation control system in align and land modes. As shown in diagram (c), as soon as the Quadrotors enters inside the state transition geometric shape, since the rate of changes in direction of the helipad path is reduced, it could be aligned with the helipad and the navigation control system transits to the land mode. Diagram (a) shows that the Quadrotor landed on the helipad.

Figures 12.7 and 12.8 depict the performance of the autopilot in roll and pitch angles control. In diagrams (a) of both of the figures, the dotted red curvatures show the desired roll and pitch angles, which are the output of the block C_2 of autopilot as discussed in chapter 4, and blue curvatures show

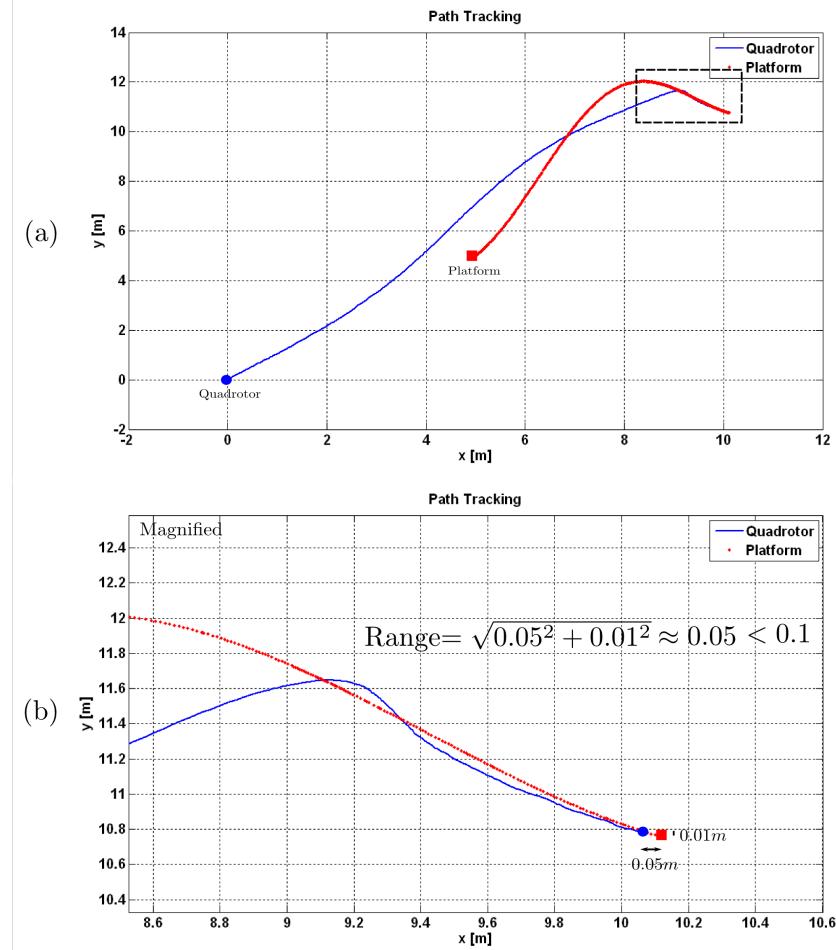


Figure 12.5: Performance of the navigation control system for a reduced rate of changes in path direction, $frequency = 1Hz$; (a) top view of scenario during simulation time, (b) magnification of the last few seconds of path tracking

the real-time roll and pitch angles, which could be measured by aids of an appropriate sensor on the Quadrotor. The diagram (b) in both figures, show the control error which is the difference between desired and real-time values. As before, it could be concluded that the errors are proportional with the magnitude of the desired angles and errors increase for bigger desired angles.

Moreover, according to diagrams (a) in both figures, the desired angles for the first few seconds, between $t \approx 1s$ and $t \approx 7s$, are zero, which implies that after initial acceleration of the Quadrotor toward the helipad, it cruises toward the helipad without any further acceleration, which is principally

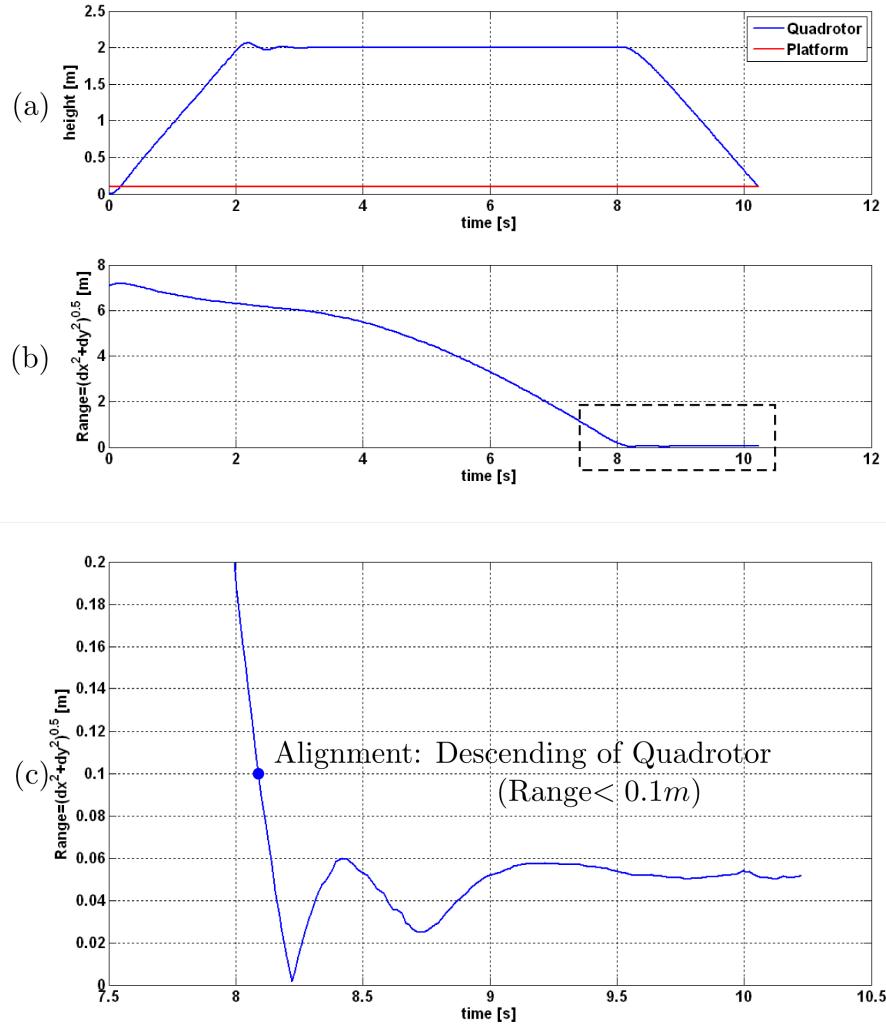


Figure 12.6: Performance of the behavior-based navigation control system in align and land modes for a reduced rate of changes in path direction, *frequency* = 1Hz; (a) altitude of the Quadrotor during simulation time, (b) planer range between Quadrotor and helipad during simulation time, (c) control transition to land mode

impossible because of the deceleration effects of the air friction and Quadrotor aerodynamic. However, in this work, these effects were neglected in Quadrotor model for simplification of the simulation. In future works, the Quadrotor model could be improved by means of including the aerodynamic and frictional effects, to achieve a more realistic simulation results.

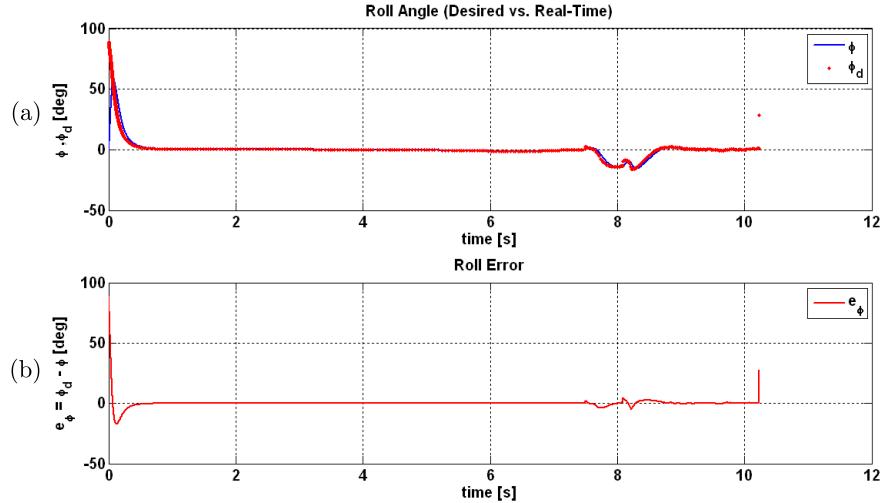


Figure 12.7: Attitude control system performance for a reduced rate of changes in path direction, (a) roll angle control system, (b) roll angle error.

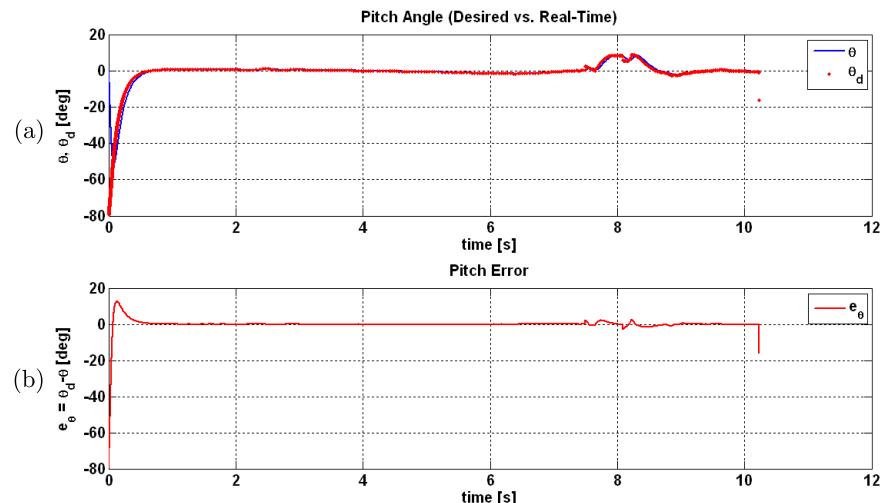


Figure 12.8: Attitude control system performance for a reduced rate of changes in path direction, (a) pitch angle control system, (b) pitch angle error.

12.3 Experiment 3: Stationary Helipad

In the last experiment, the performance of the control system in the special case of stationary platform has been studied. Similar to the previous experiments, the Quadrotor initially located at the origin of the refer-

ence coordinate system, $(x_{Q0}, y_{Q0}, z_{Q0}) = (0, 0, 0)$, with initial orientation of $(\phi_{Q0}, \theta_{Q0}, \psi_{Q0}) = (0, 0, 0)$ and the helipad initially located at the position $(x_{p0}, y_{p0}, z_{p0}) = (5m, 5m, 0.1m)$, with a fixed height above the ground equal to 10cm and it has been assumed to be stationary.

$$y_p(x(t)) = 0 \quad (12.7)$$

$$x_p(t) = 0 \quad (12.8)$$

The wind condition is similar to the previous experiments. It has been assumed that the wind blows with the maximum speed of $0.2m/s$ in the direction of the local East to West. The formulation will be similar to 12.3 and 12.4.

The simulation time also has been set to 25s. The new simulation results have been depicted in the figures 12.9, and 12.10.

Figure 12.9 shows the performance of the navigation control system from a two dimensional top view of the scenario in NE-coordinate system, in which the Quadrotor takes off from its initial position and cruises toward the stationary helipad. The simulation results proves the efficiency of the control system also in the case of stationary helipad. As depicted in the diagram (b) of figure 12.9, it is interesting to mention that the alignment between Quadrotor and helipad in this experiment was quite accurate as final planer distance between them is only $R = 2cm$.

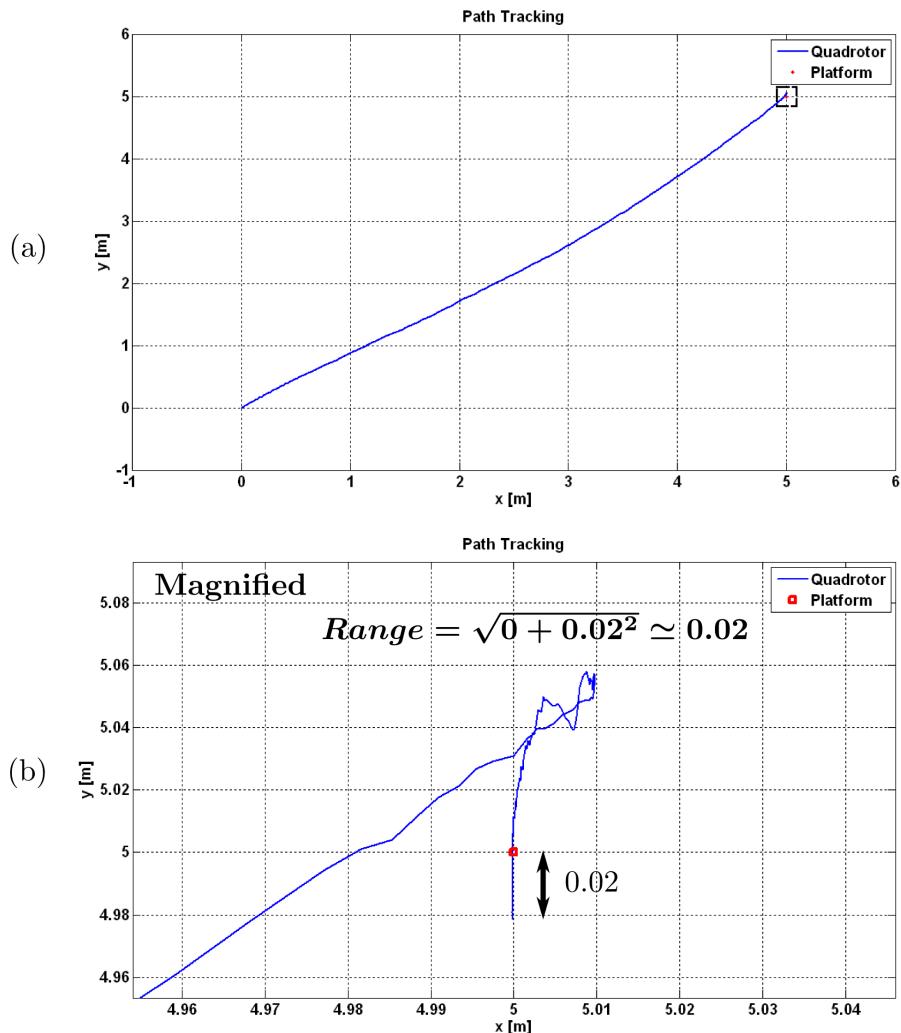


Figure 12.9: Performance of the controller in cruise mode for a stationary helipad; (a) top view of path tracking diagram within 25s, (b) magnification of the last few seconds of path tracking

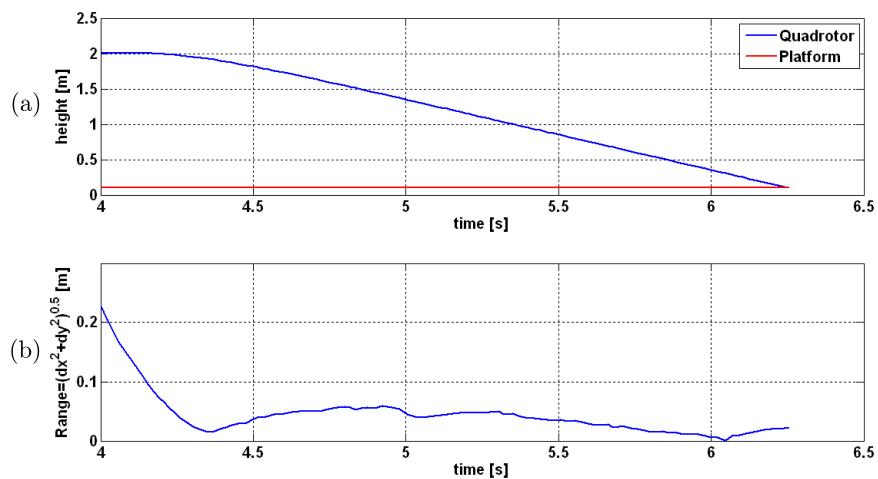


Figure 12.10: Performance of the navigation control system in align and land mode for a stationary helipad; (a) altitude of the Quadrotor during simulation time, (b) planer range between Quadrotor and helipad during simulation time.

Chapter 13

Summary and Future Works

In this thesis work a behavior-based navigation control system for autonomous landing of a Quadrotor on a mobile helipad have been developed which is mainly adapted form command to line-of-sight (CLOS) system. This navigation control system consists of three states of *cruise*, *align*, and *land* mode and the state transition has been performed depending to the position of the Quadrotor relative to an appropriately defined state transition geometric shape above the helipad. In addition, for attitude stabilization of the Quadrotor, an autopilot has been implemented which is mainly adapted from feedback linearization method.

The performance of the proposed behavior-based navigation control system for a simplified Quadrotor and wind model has been evaluated through three simulation with different helipad characteristics. In the first simulation, the haeipad was moving with a high rate of changes in path direction and therefore, although the navigation control system was successful to track the mobile helipad, it fails to keep the Quadrotor inside the state transition geometric shape and the Quadrotor was not able to land on the helipad. In the second simulation, by reducing the rate of changes in the path direction of the helipad, the control system was able to keep the Quadrotor inside the state transition geometric shape and therefore, Quadrotor was able to land on the mobile helipad. In the last simulation, the control system proved to have a good performance in the case of stationary helipad, in which the Quadrotor was successfully landed on the stationary helipad with an enhanced alignment accuracy of $2cm$.

Simulation results reveals that autopilot has good performance for attitude control in all cases, as the Quadrotor is able to follow the desired roll and pitch angles with a reasonably short response time and small error. However, due to the simplifications and neglects of the aerodynamic and frictional effects in the Quadrotor model, the performance of the Quadrotor

in the cruise mode was not realistic, as it continues to cruise with initially acceleration without any further enforcement.

In future works, the Quadrotor model should be enhanced and aerodynamic and fictional effects should be included in the model. In addition, a more enhanced and detailed model for the wind effect should be applied. In addition, in this work the obstacles and coordination of multi Quadrotors and collision avoidance were not in the focus of the research, which could be studied in future works. Finally, the performance of the overall navigation control system should be realized and evaluated experimentally.

Appendix A

S-Function for Quadrotor

```
%This is a S-Function for the Quadrotor state space model
%with 12 states.
function [sys, x0] = quadrotor(t,x,u,flag)

    g = 9.81;           %Gravity of the Earth
    Ix = 4.85e-3;      %Inertia
    Iy = 4.85e-3;      %Inertia
    Iz = 8.81e-3;      %Inertia
    L = 0.2;            %Lever's length
    m = 0.5;            %Quadrotor mass
    q = pi/180;          %Degree to radian converting factor
    Jr = 3.36e-5;       %Rotor inertia
    b = 2.92e-6;         %Thrust factor
    d = 1.12e-7;         %Drag factor

    xi=1;                %Initial x position of Quadrotor
    yi=2;                %Initial y position of Quadrotor
    zi=0;                %Initial z position of Quadrotor
    vxi=0;               %Initial x velocity of Quadrotor
    vyi=0;               %Initial y velocity of Quadrotor
    vzi=0;               %Initial z velocity of Quadrotor
    phii=0*q;             %Initial phi of Quadrotor
    thetai=0*q;            %Initial theta of Quadrotor
    psii=0*q;              %Initial psi of Quadrotor

    T =[b b b b;0 b 0 -b; b 0 -b 0; d -d d -d];
    %Transformation matrix
%=====
if abs(flag) == 0        % Initial conditions:
```

```

% sys = [#states, # discrete states, #outputs, #inputs,
%          roots, ];
sys = [12, 0, 12, 6, 0, 0];
x0 = zeros(12,1);

x0(1)=xi;           %Initial x position of Quadrotor
x0(2)=yi;           %Initial y position of Quadrotor
x0(3)=zi;           %Initial z position of Quadrotor
x0(4) = vxi;         %Initial x velocity of Quadrotor
x0(5) = vyi;         %Initial y velocity of Quadrotor
x0(6) = vzi;         %Initial z velocity of Quadrotor
x0(7) = phii;        %Initial phi of Quadrotor
x0(8) = thetai;       %Initial theta of Quadrotor
x0(9) = psii;        %Initial psi of Quadrotor
x0(10) = 0;           %phidot_0
x0(11) = 0;           %thetadot_0
x0(12) = 0;           %psidot_0

end

%=====
if abs(flag) == 1      % Differential Equations
                         % Representing System

    uu = u;             %Control inputs
    os = inv(T)*u(1:4,1); %os=[w1^2 w2^2 w3^2 w4^2]', and
                          %wi=rotational speed of ith motor
    os = (os + abs(os))/2;%In order to avoid negative wi.
                          %Negative wi means rotation of the
                          %blades in opposite direction; but
                          %however, it is principally impossible
                          %and results in errors in MATLAB.

    omega = sqrt(os(1))-sqrt(os(2))+sqrt(os(3))-sqrt(os(4));
                         %g(u)

    sys(1)=x(4)+u(5);   %Addition of wind component in x
                         %direction as disturbance to the
                         %system.
    sys(2)=x(5)+u(6);   %Addition of wind component in y
                         %direction as disturbance to the
                         %system.
    sys(3)=x(6);

```

```

sys(4) = -(cos(x(7))*sin(x(8))*cos(x(9))
         + sin(x(7))*sin(x(9)))/m*uu(1);
sys(5) = -(cos(x(7))*sin(x(8))*sin(x(9))
         - sin(x(7))*cos(x(9)))/m*uu(1);
sys(6) = g - (cos(x(7))*cos(x(8))/m*uu(1));
sys(7) = x(10);
sys(8) = x(11);
sys(9) = x(12);
sys(10) = (x(11)*x(12)*(Iy-Iz)/Ix) - (Jr/Ix*x(11)*omega)
           + (L/Ix*uu(2));
sys(11) = (x(10)*x(12)*(Iz-Ix)/Iy) - (Jr/Iy*x(10)*omega)
           + (L/Iy*uu(3));
sys(12) = (x(10)*x(11)*(Ix-Iy)/Iz) + (1/Iz*uu(4));

end; % if abs(flag) == 1

%=====
if abs(flag) == 3
    %Generating Outputs, sys(i)=x(i) are the states

    sys(1) = x(1);
    sys(2) = x(2);
    sys(3) = x(3);
    sys(4) = x(4);
    sys(5) = x(5);
    sys(6) = x(6);
    sys(7) = x(7);
    sys(8) = x(8);
    sys(9) = x(9);
    sys(10) = x(10);
    sys(11) = x(11);
    sys(12) = x(12);

end; % if abs(flag) == 3

```

Appendix B

M-File for Autopilot

```
%This is a m-file for the Autopilot

function u = autopilot(x)

    g = 9.81;          %Gravity of the Earth
    Ix = 4.85e-3;      %Inertia
    Iy = 4.85e-3;      %Inertia
    Iz = 8.81e-3;      %Inertia
    L = 0.2;           %Lever's length
    m = 0.5;           %Quadrotor mass
    q = pi/180;         %Degree to radian converting factor
    Jr = 3.36e-5;       %Rotor inertia
    b = 2.92e-6;        %Thrust factor
    d = 1.12e-7;        %Drag factor

    xd = zeros(12,1);
    %Initializing desired states
    xd(1)=0;           % x
    xd(2)=0;           % y
    xd(3)=0;           % z
    xd(4)=x(13);       %Desired lateral velocity (vdx)
    xd(5)=x(14);       %Desired lateral velocity (vdy)
    xd(6)=x(15);       %Desired lateral velocity (vdz)
    xd(7) = 0;          % phi
    xd(8) = 0;          % theta
    xd(9) = 0;          % psi
    xd(10) = 0;         % phidot
    xd(11) = 0;         % thetadot
    xd(12) = 0;         % psidot
```

```

I1 = (Iy-Iz)/Ix;      %Definition
I2 = (Iz-Ix)/Iy;      %Definition
I3 = (Ix-Iy)/Iz;      %Definition

%u tilde, new control inputs
%defined in a way to linearize M2
ut1 = 5*(xd(4) - x(4));
ut2 = 5*(xd(5) - x(5));
ut3 = 5*(xd(6) - x(6));

u(1) = m*(g - ut3); %u1, control input

xd(7) = ut2/(g-ut3);    %Desired phi

%Saturation block
%to limit phi, roll angle, between -90° and +90°
if xd(7)<=-90*q      %-90° limitation for phi
    xd(7)=-90*q;
elseif xd(7)>=90*q   %+90° limitation phi
    xd(7)=90*q;
end

xd(8) = - ut1/(g - ut3);    %Desired theta

%Saturation block
%to limit theta, pitch angle, between -90° and +90°
if xd(8)<=-90*q      %-90° limitation for theta
    xd(8)=-90*q;
elseif xd(8)>=90*q   %+90° limitation for theta
    xd(8)=90*q;
end

K2 = -80;
a2 = (0.5*K2)^2*Ix/L;
K3 = -80;
a3 = (0.5*K3)^2*Iy/L;
K4 = -80;
a4 = (0.5*K4)^2*Iz;

u(2) = (Ix/L)*(K2*x(10) - I1*x(11)*x(12))
      + a2*(xd(7) - x(7));
      %u2, control input
u(3) = (Iy/L)*(K3*x(11) - x(10)*x(12)*I2)

```

```
+ a3*(xd(8) - x(8));
%u3, control input
u(4) = Iz*(K4*x(12) - x(10)*x(11)*I3)
+ a4*(xd(9)- x(9));
%u4, control input
u(5)=xd(7); %Desired roll angle
u(6)=xd(8); %Desired pitch angle
```

Appendix C

M-File for Tracking Model

```
%This is a m-file for search or track control  
%based on constant line-of-sight angle method.  
  
function u = track(x)  
  
R=x(1); %Planner range  
sigma=x(2); %Line-Of-Sight angle  
vp=x(3); %Platform speed  
alpha_p=x(4); %Platform motion direction  
  
vpx=vp*cos(alpha_p); %x component of platform motion  
vpy=vp*sin(alpha_p); %y component of platform motion  
  
%If planner distance between Quadrotor and mobile platform  
%is bigger than 1m, set the Quadrotor speed to max  
%(Vq,max=2m/s)  
if R>1  
    vq=2;  
else  
    %If planner distance between Quadrotor and mobile platform  
    %is less than 1m, set the Quadrotor speed to be proportional  
    %with the range  
    vq=2*R+0.5;  
end  
  
%Alpha_Q in collision path  
beta=sigma+asin(vp/vq*sin(alpha_p-sigma));
```

```
denominator=cos(sigma)*tan(beta)-sin(sigma);  
  
%To avoid devision by 0 in simulation  
if (denominator==0)  
    denominator=0.01;  
end  
  
if vp==0  
    vq=2*R;  
    vqx=vq*cos(beta);  
    vqy=vq*sin(beta);  
else  
    vqx=(vpy*cos(sigma)-vpx*sin(sigma))/denominator;  
    vqy=(vpy*cos(sigma)*tan(beta)  
          -vpx*sin(sigma)*tan(beta))/denominator;  
end  
  
vq=sqrt(vqx^2+vqy^2);  
if vq>2      %Maximum speed of Quadrotor(V_Q,max=2m/s)  
    vqx=vqx*2/vq;  
    vqy=vqy*2/vq;  
end  
  
%Rate of changes in range  
Rdot=vp*cos(alpha_p-sigma)-vq;  
  
%Outputs  
u(1)=vqx;  
u(2)=vqy;  
u(3)=Rdot;
```

Appendix D

M-File for Altitude Controller

```
%This is a m-file for altitude control

function u = height(x)

sfh=2;          %Safe altitude above ground during
                 %search mode
lbr=1;          %Mobile landing platform radius
mala=0.1;        %Minimum acceptable accuracy for
                 %land termination
zcp=x(1);        %Real time altitude of the mobile
                 %landing platform above ground
zcq=x(2);        %Real time altitude of the quadrotor
                 %above ground
Range=x(3);       %Planer range

zprime=Range*sfh/lbr;
                 %Altitude inside the landing cone volume

if (mala<Range<lbr && zcq>zprime)
                 %If Range is inside the landing cone
                 %volume, quadrotor will start to descent
                 %(desired altitude = altitude inside
                 %the landing cone volume )
zdot=(zprime-zcq)*30;

elseif(Range<=mala && zcq>zcp)
                 %If Range is inside the min acceptable
                 %land accuracy limit, full land will
```

```
%start (desired altitude = altitude of
%the mobile platform )
zdot=(zcp-zcq)*30;

else
    zdot=(sfh-zcq)*30;
    %In search mode, in which range is outside
    %of the landing cone volume, keep quadrotor
    %to fly in predefined safe altitude above
    %ground
end

%Saturate the max vertical component of lateral speed
%(system limitation)
if zdot>=1
    zdot=1;
elseif zdot<=-1
    zdot=-1;
end

u(1)=zdot; %Output of the Simulink block (vertical
            %component of lateral speed)
```

Appendix E

S-Function for On-board Mobile Helipad

```
%This is a S-Function for the platform.

function [sys, x0] = platform(t,x,u,flag)

xpi=0; %Initial position of platform (x)
ypi=0; %Initial position of platform (y)
zpi=0; %Initial position of platform (z)
v=0.2; %Platform speed
f=1; %Frequency of platform sinusoidal movement
A=1; %Amplitude of platform sinusoidal movement
rp = random('Poisson',1:1,1,1);
%A random value (to simulate vibration of platform
%during its motion)

%=====
if abs(flag) == 0 %Initial conditions

% sys = [#states, # discrete states, #outputs,
% #inputs, roots, ];
sys = [2, 0, 5, 0, 0, 0];

x0 = zeros(2,1);

x0(1)=xpi; %x
x0(2)=ypi; %y

end
=====
```

```
if abs(flag) == 1
    % Differential Equations Representing System
    % x=v.t+x0 ==> dx=v
    % y=x+A.sinpx ==> dy=dx+A.p.dx.cospx

    sys(1)=v;                      % dx
    sys(2)=v+A*p*v*cos(p*x(1)); % dy

end; % if abs(flag) == 1

%=====
if abs(flag) == 3           %Outputs

    sys(1) = x(1);          %x
    sys(2) = x(2);          %y (movement on
                           % (x,x+sinx) path)
    sys(3) = zpi+0.01*rp; %Real-time platform altitude
                           %assumption: vibration of
                           %platform in the range of
                           %1cm = 0.01m

    sys(4)=v;                %Platform velocity
    sys(5)=atan(1+A*p*cos(p*x(1)));
                           %Platfrom motion direction
                           %alpha_platform=atan(dy/dx)
                           %dy/dx=1+cosx

end; % if abs(flag) == 3
```

Appendix F

S-Function for Wind Model

```
%This is a S-Function for wind model.

function [sys, x0] = wind(t,x,u,flag)

q=pi/180;          %Degree to radian converting factor
psi_w=210*q;      %Wind direction (South-West direction)
v_w=0.2*sin(t);  %Wind speed

%=====
if abs(flag) == 0    % Initial conditions :

    % sys = [#states, # discrete states, #outputs,
    %           #inputs, roots, ];
    sys = [2, 0, 2, 0, 0, 0];

    x0 = zeros(2,1);
    x0(1)=0;
    x0(2)=0;
end
%=====
if abs(flag) == 1    % Differential Equations

    sys(1)=0;
    sys(2)=0;

end; % if abs(flag) == 1

%=====
if abs(flag) == 3      %Outputs
```

```
%x component of wind (V_w,x)
sys(1) = (v_w+0.1*random('Poisson',1:1,1,1))
          *sin(psi_w+q*random('Poisson',1:1,1,1));

%y component of wind (V_w,y)
sys(2) = (v_w+0.1*random('Poisson',1:1,1,1))
          *cos(psi_w+q*random('Poisson',1:1,1,1));

end; % if abs(flag) == 3
```

Appendix G

M-File for Range Sensor

```
%This is a m-file for the range sensor

function u = range(x)

    xcp=x(1);    %Real time platform position (x)
    ycp=x(2);    %Real time platform position (y)
    xcq=x(3);    %Real time quadrotor position (x)
    ycq=x(4);    %Real time quadrotor position (y)

    deltax=xcp-xcq;
    deltay=ycp-ycq;

    R=sqrt(deltax.^2+deltay.^2);
        %Planner range (on xy plane)

    sigma=atan(abs(deltay)/abs(deltax));

    if (deltax<=0 && deltay>0)      %2nd quarter
        sigma=pi-sigma;
    elseif(deltax<0 && deltay<=0)   %3rd quarter
        sigma=pi+sigma;
    elseif (deltax>=0 && deltay<0)  %4th quarter
        sigma=2*pi-sigma;
    end

    u(1)=R;      %Range
    u(2)=sigma; %Line-Of-Sight angle
```

Bibliography

- [1] Saripalli S. Sukhatme G. S., and , Montgomery J. F. *An Experimental Study of the Autonomous Helicopter Landing Problem*. In Proceedings of International Symposium on Experimental Robotics (ISER), 2002, Sant'Angelo d'Ischia, Italy
- [2] Slotine J. J. E., and LI W. *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, New Jersey, pp. 207-275
- [3] Chen M., and Huzmezan M. *A Simulation Model and $H\infty$ Loop shaping Control of a Quad Rotor Unmanned Air Vehicle*. Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada
- [4] Ringbeck T., Hagebeuker B. *A 3D Time of Flight Camera for Object Detection*. Optical 3-D Measurement Techniques, 09-12 July, 2007, ETH, Zürich
- [5] Tayebi A., and McGilvray S. *Attitude stabilization of a four-rotor aerial robot*. 43rd IEEE Conference on Decision and Control December 14-17, 2004, Atlantis, Paradise Island, Bahamas
- [6] McGilvray S. J. *Attitude Stabilization of a Quadrotor Aircraft*. Masters Thesis Report, June 2004, Lakehead University, Thunder Bay, Ontario, Canada
- [7] Merz T., Duranti S., and Conte G. *Autonomous Landing of an Unmanned Helicopter based on Vision and Inertial Sensing*. Department of Computer and Information Science Linköping University, SE-58183 Linköping, Sweden
- [8] Barber D. B., Griffiths S. R., McLain T. W., and Beard R. W. *Autonomous Landing of Miniature Aerial Vehicles*. Brigham Young University Provo, UT
- [9] Bosse M., Karl W. C., Castañon D., and DeBitetto P. *A Vision Augmented Navigation System*. IEEE 1998, 0-7803-4269-0/97

- [10] Sharp C. S., Shakernia O., and Sastry S. S. *A Vision System for Landing an Unmanned Aerial Vehicle*. Department of Electrical Engineering & Computer Science University of California Berkeley, Berkeley
- [11] Bouabdallah S., Murrieri P., Siegwart R. *Design and Control of an Indoor Micro Quadrotor*. Proceedings IEEE International Conference on Robotics and Automation, 2004, Vol.5, pp. 4393-4398
- [12] In Joong Ha, and Chong S. *Design of a CLOS Guidance Law Via Feed-back Linearization*. IEEE Transactions on Aerospace and Electronic Systems, Vol. 28, No. 1, January 1992
- [13] Hoffmann G. M., Waslander S. L., and Tomlin C. J. *Distributed Cooperative Search using Information-Theoretic Costs for Particle Filters, with Quadrotor Applications*. American Institute of Aeronautics and Astronautics
- [14] Tournier G. P., Valenti M., Howz J. P., and Feronx E. *Estimation and Control of a Quadrotor Vehicle Using Monocular Vision and Moiré Patterns*. AIAA Guidance, Navigation, and Control Conference and Exhibit, 21-24 August 2006, Keystone, Colorado
- [15] Shakernia O., Ma Y., John Koo T., and Sastry S. *Landing an Unmanned Air Vehicle: Vision Based Motion Estimation and Nonlinear Control*. Department of Electrical Engineering & Computer Science University of California Berkeley, Berkeley
- [16] Roberts J. M., Corke P. I., and Buskey G. *Low-Cost Flight Control System for a Small Autonomous Helicopter*. Proc. 2002 Australasian Conference on Robotic and Automation, Auckland, 27-29 November 2002
- [17] Dabney J. B. *Mastering SIMULINK 4*. Microsoft
- [18] Bishop R. H. *The Mechatronics Handbook-The Electrical Engineering Handbook Series*. Crain T. P. Chapter 28, *Kalman Filters as Dynamic System State Observer*. NASA Johnson Space Center, CRC Press
- [19] Pounds P., Mahony R., Corke P. *Modelling and Control of a Quad-Rotor Robot*. Australian National University, Canberra, Australia
- [20] Bresciani T. *Modelling, Identification and Control of a Quadrotor Helicopter*. Masters Thesis Report, Department of Automatic Control, Lund University, October 2008
- [21] Martínez V. M. *Modelling of the Flight Dynamics of a Quadrotor Helicopter*.asters Thesis Report, 2007, School of Engineering, Cranfield University

- [22] Dorf R. C., and Bishop R. H. *Modern Control Systems*. Prentice Hall, Upper Saddle River, NJ, pp. 631-679
- [23] unknown writer *MTi and MTx User Manual and Technical Documentation*. Document MT0100P, Revision G, March 2, 2006, Xsens Technologies B.V., The Netherlands
- [24] Waslander S.L., Hoffmann G.M., Jung S. J., and Tomlin C.J. *Multi-agent quadrotor testbed control design: integral sliding mode vs. reinforcement learning*. IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS 2005), 2005, pp. 3712-3717, ISBN 0-7803-8912-3
- [25] Albertos P., and Sala A. *Multivariable Control Systems: An Engineering Approach*. Springer, pp. 267-269
- [26] unknown writer *NAVSTAR GPS User Equipment Introduction*. unknown publisher, September 1996
- [27] Voos H. *Nonlinear and Neural Network-based Control of a Small Four-Rotor Aerial Robot*. IEEE/ASME international conference on advanced intelligent mechatronics, Zurich, 2007, pp. 1-6, ISBN 978-1-4244-1263-1
- [28] Voos H. *Nonlinear Control of a Quadrotor Micro-UAV using Feedback-Linearization*. IEEE International Conference on Mechatronics (ICM 2009), 14-17 April, 2009, Málaga, Spain
- [29] Voos H. *Nonlinear State-Dependent Riccati Equation Control of a Quadrotor UAV*. IEEE International Symposium on Intelligent Control, 4-6 Oct. 2006, pp. 2547-2552
- [30] Etele J. *Overview of Wind Gust Modelling with Application to Autonomous Low-Level UAV Control*. Defence R&D Canada, Ottawa, CONTRACT REPORT DRDC Ottawa CR 2006-221, November 2006, Mechanical and Aerospace Engineering Department, Carleton University, Canada
- [31] Bouabdallah S., Noth A., and Siegwart R. *PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor*. Autonomous Systems Laboratory, Swiss Federal Institute of Technology, Lausanne, Switzerland
- [32] Harbick K., Montgomery J. F., and Sukhatme G. S. *Planner Spline Trajectory Following for an Autonomous Helicopter*. University of Southern California
- [33] Hoffmann G. M., Huang H., Waslander S. L., and Tomlin C. J. *Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment*.

AIAA Guidance, Navigation and Control Conference and Exhibit,
Hilton Head, South Carolina, 20-23 August 2007

- [34] Hoffmann G. M., Waslander S. L., and Tomlin C. J. *Quadrotor Helicopter Trajectory Tracking Control*. American Institute of Aeronautics and Astronautics
- [35] Danko T. W., Kellas A., and Oh P. Y. *Robotic Rotorcraft and Perch-and-Stare: Sensing Landing Zones and Handling Obscurants*. Drexel University, Philadelphia PA
- [36] Mokhtari A., Benallegue A., and Daachi B. *Robust Feedback Linearization and GH ∞ Controller for a Quadrotor Unmanned Aerial Vehicle*. Journal of ELECTRICAL ENGINEERING, vol. 57, no. 1, 2006, pp. 20-27
- [37] Voos H. *UAV “See And Avoid” With Nonlinear Filtering and Non-Cooperative Avoidance*. Proceeding Robotics and Applications and Telematics, August 29-31, 2007, Würzburg, Germany
- [38] Everett H. R. *Sensors for Mobile Robots: Theory and Application*. A K Peters, Ltd., 1995, Natick, Massachusetts, pp. 139-168, pp. 395-423
- [39] Hoffmann G., Rajnarayan D. G., Waslander S. L., Dostal D., Jang J. S., and Tomlin C. J. *The Stanford Testbed of Autonomous Rotorcraft for Multi Agent Control (STARMAC)*. Digital Avionics Systems Conference 2004, Vol. 2, ISBN 0-7803-8539
- [40] Yu Z., Nonami K., Shin J., and Celestino D. *3D Vision Based Landing Control of a Small Scale Autonomous Helicopter*. International Advanced Robotic Systems, Unmanned Aerial Vehicle Lab., Electronics and Mechanical Engineering, Chiba University
- [41] Pounds P., Mahony R., and Gresham J. *Towards Dynamically-Favourable Quad-Rotor Aerial Robots*. Australian National University, Canberra, Australia
- [42] Bouabdallah S., Becker M., Perrot V. de, and Siegwart R. *Toward Obstacle Avoidance on Quadrotors*. Proceedings of the XII International Symposium on Dynamic Problems of Mechanics (DINAME 2007), ABCM, Ilhabela, SP, Brazil, February 26 - March 2, 2007
- [43] Wise R. A., and Rysdyk R. T. *UAV Coordination for Autonomous Target Tracking*. Autonomous Flight Systems Laboratory, University of Washington, Seattle
- [44] Rysdyk R. *UAV Path Following for Constant Line-Of-Sight*. American Institute of Aeronautics and Astronautics

- [45] Rysdyk R. *UAV Path Following for Target Observation in Wind*. University of Washington, Seattle
- [46] Saripalli S., Montgomery J. F. , and Sukhatme G. S. *Vision-based Autonomous Landing of an Unmanned Aerial Vehicle*. In Proceedings Of IEEE International Conference on Robotics and Automation, May 2002, Washington D.C., USA, pp. 2799-2804
- [47] Saripalli S., Montgomery J. F. , and Sukhatme G. S. *Visually-Guided Landing of an Unmanned Aerial Vehicle*. IEEE Transactions on Robotics and Automation, Vol 19, No 3, pp 371-381, Jun 2003
- [48] Ruffier F., and Franceschini N. *Visually Guided Micro-Aerial Vehicle: Automatic Take off, Terrain Following, Landing and Wind Reaction*. Proceeding of the 2004 IEEE International Conference on Robotics and Automation, April 2004, New Orleans, LA
- [49] Stoleru R., He T., Stankovic J. A. *Walking GPS: A Practical Solution for Localization in Manually Deployed Wireless Sensor Networks*. Department of Computer Science, University of Virginia
- [50] Osborne J., and Rysdyky R. *Waypoint Guidance for Small UAVs in Wind*. University of Washington, Seattle, WA