

Linköping Studies in Science and Technology

Thesis No. 938

Fuzzy Control for an Unmanned Helicopter

by

Bourhane Kadmiry



INSTITUTE OF TECHNOLOGY
LINKÖPINGS UNIVERSITET

Submitted to the School of Engineering at Linköping University in partial fulfilment of the requirements for degree of Licentiate of Engineering.

Department of Computer and Information Science
Linköpings universitet
SE-581 83 Linköping, Sweden

Linköping 2002

Tryckt av Unitytryck, Linköping, Sverige, 2002

ISBN 91-7373-313-X

ISSN 0280-7971

Fuzzy Control for an Unmanned Helicopter

by

Bourhane Kadmiry

March 2002

ISBN 91-7373-313-X

Linköping Studies in Science and Technology

Thesis No. 938

ISSN 0280-7971

LiU-Tek-Lic-2002:11

ABSTRACT

The overall objective of the Wallenberg Laboratory for Information Technology and Autonomous Systems (WITAS) at Linköping University is the development of an intelligent command and control system, containing vision sensors, which supports the operation of an unmanned air vehicle (UAV) in both semi- and full-autonomy modes. One of the UAV platforms of choice is the APID-MK3 unmanned helicopter, by Scandicraft Systems AB. The intended operational environment is over widely varying geographical terrain with traffic networks and vehicle interaction of variable complexity, speed, and density.

The present version of APID-MK3 is capable of autonomous take-off, landing, and hovering as well as of autonomously executing pre-defined, point-to-point flight where the latter is executed at low-speed. This is enough for performing missions like site mapping and surveillance, and communications, but for the above mentioned operational environment higher speeds are desired. In this context, the goal of this thesis is to explore the possibilities for achieving stable “aggressive” manoeuvrability at high-speeds, and test a variety of control solutions in the APID-MK3 simulation environment.

The objective of achieving “aggressive” manoeuvrability concerns the design of attitude/velocity/position controllers which act on much larger ranges of the body attitude angles, by utilizing the full range of the rotor attitude angles. In this context, a flight controller should achieve tracking of curvilinear trajectories at relatively high speeds in a robust, w.r.t. external disturbances, manner. Take-off and landing are not considered here since APID-MK3 has already have dedicated control modules that realize these flight modes.

With this goal in mind, we present the design of two different types of flight controllers: a fuzzy controller and a gradient descent method based controller. Common to both are model based design, the use of nonlinear control approaches, and an inner- and outer-loop control scheme. The performance of these controllers is tested in simulation using the nonlinear model of APID-MK3.

This work was supported by a research grant provided by the Knut and Alice Wallenberg Foundation in Sweden.

Department of Computer and Information Science
Linköpings universitet
SE-581 83 Linköping, Sweden

Linköping Studies in Science and Technology
Dissertation No. 938

Fuzzy Control for an Unmanned Helicopter

By
Bourhane Kadmiry

Submitted to the School of Engineering at Linköping University in partial
fulfilment of the requirements for the degree of Licenciate of Engineering

Department of Computer and Information Science
Linköpings Universitet
SE-581 83 Linköping, Sweden

Linköping 2002

Abstract

The overall objective of the Wallenberg Laboratory for Information Technology and Autonomous Systems (WITAS) at Linköping University is the development of an intelligent command and control system, containing vision sensors, which supports the operation of a unmanned air vehicle (UAV) in both semi- and full-autonomy modes. One of the UAV platforms of choice is the APID-MK3 unmanned helicopter, by Scandicraft Systems AB. The intended operational environment is over widely varying geographical terrain with traffic networks and vehicle interaction of variable complexity, speed, and density.

The present version of APID MK-3 is capable of autonomous take-off, landing, and hovering as well as of autonomously executing pre-defined, point-to-point flight where the latter is executed at low-speed. This is enough for performing missions like site mapping and surveillance, and communications, but for the above mentioned operational environment higher speeds are desired. In this context, the goal of this thesis is to explore the possibilities for achieving stable “aggressive” manoeuvrability at high-speeds, and test a variety of control solutions in the APID-MK3 simulation environment.

The objective of achieving “aggressive” manoeuvrability concerns the design of a attitude/velocity/ position controllers which act on much larger ranges of the body attitude angles, by utilizing the full range of the rotor attitude angles. In this context, a flight controller should achieve tracking of curvilinear trajectories at relatively high speeds in a robust, w.r.t. external disturbances, manner. Take-off and landing are not considered here since APID MK-3 has already have dedicated control modules that realize these flight modes.

With this goal in mind, we present the design of two different types of flight controllers: a fuzzy controller and a gradient descent method based controller. Common to both are model based design the use of nonlinear control approaches and an inner- and outer-loop control scheme. The performance of these controllers is tested in simulation using the nonlinear model of APID-MK3.

This work was supported by a research grant provided by the Knut and alice Wallenberg Foundation in Sweden.

Acknowledgments

I would like to thank Dimiter Driankov, my supervisor, for his many suggestions and constant support during this research. He is a wonderful person and his support makes research like this possible.

I would like to thank Rainer Palm, at SIEMENS AG Munich-Germany, and Pontus Bergstein, at Örebro university, for our good collaboration. They shared with me their knowledge and provided me useful references and friendly encouragement.

I am also thankful to Thomas Högström, at Scandicraft AB, who helped with understanding the APID-MK3 model and supplied additional information to the platform model; and to Erik Skarman, at SAAB industry, for his experienced advises during our early work on the WITAS platform.

The WITAS project at the Department of Computer Science has been a stimulating place for my work. Thanks to Erik Sandewall for his guidance and interest in my work; and Patrick Doherty for giving me the opportunity to do research in such an inspiring environment.

I had the pleasure of meeting Malik Guallab, at LAAS, Toulouse university, who expressed his interest in my work and supplied me with help and support, directing me to some recent work in the field, which gave me a better perspective on my own results. I would like also to mention that my graduate studies in Sweden are done jointly with the (LAAS) Research laboratory. My thanks go the 'team' there for their welcoming me into their group.

Thanks to the Knut and Alice Wallenberg Foundation in Sweden, which has supported this work by a research grant.

Finally, I am grateful to my parents for their patience and love. Without them this work would never have come into existence.

Linköping, Sweden
April 20, 2002

Bourhane KADMIRY

To 'my' Susanne

Table of Contents

Abstract	iii
Acknowledgments	v
Table of Contents	vii
List of Figures	ix
1 Introduction	1
1.1 Motivation	1
1.1.1 Why aggressive manoeuvrability	2
1.1.2 Why fuzzy gain scheduling ?	3
1.1.3 Why not conventional gain scheduling ?	5
1.2 Related Work	6
1.2.1 The Berkeley Aerorobot Team	6
1.2.2 The Georgia Tech Aerial Robotics Mission	7
1.2.3 The MIT Backstepping Controller	7
1.2.4 The Compiegne University Controller	8
1.2.5 The Fuzzy Unmanned Helicopter	8
1.3 The Purpose of the Thesis	9
1.4 Publications	10
1.5 Contributions	10
1.6 Outline of the thesis	11
2 The helicopter model	13
2.1 Helicopter basic concepts	14
2.2 A general model	15
2.2.1 General kinematics and dynamics	15
2.2.2 Nature of the model	18
2.3 Equations of motion	19
2.3.1 Translational motion	19
2.3.2 Rotational motion	24
2.4 Mathematical model of APID-MK3	27
2.4.1 APID-MK3 general model	27
2.4.2 Longitudinal motion model	28
2.4.3 Lateral motion model	29

2.4.4	Vertical motion model	30
2.4.5	Roll model	31
2.4.6	Pitch model	32
2.4.7	Yaw model	33
2.5	Comparison with other helicopter models	34
2.5.1	The Berkeley model	34
2.5.2	The CMU model	38
2.6	Summary	39
3	Flight Controller Design	41
3.1	Introduction	41
3.2	The control scheme	42
3.3	Fuzzy Gain-Scheduled Control	45
3.3.1	Structure of the Takagi-Sugeno model	46
3.3.2	TS models for dynamical systems	47
3.3.3	Obtaining TS fuzzy models	48
3.3.4	Takagi-Sugeno controllers	52
3.4	Mamdani-type controllers	56
3.4.1	The fuzzy rule base	56
3.4.2	The Mamdani PD-controller	58
3.5	The Fuzzy Flight Controller	59
3.5.1	The FGS controller for the inner-loop	59
3.5.2	Linearization of the inner-loop model	63
3.5.3	Mamdani fuzzy controller for the outer-loop	67
3.5.4	Related work	70
3.6	The Gradient-Descent Flight Controller	71
3.6.1	The open loop model	72
3.6.2	The inner loop attitude controller	72
3.6.3	The outer loop velocity controller	73
3.6.4	The outer-loop position control	75
3.7	Summary	76
4	Simulation results	77
4.1	Introduction	77
4.2	Simulation with the fuzzy flight controller	78
4.2.1	Robustness	78
4.2.2	Attitude control robustness	78
4.2.3	Altitude control robustness	81
4.2.4	Aggressive flying	87
4.3	Simulation results for GDM controller	94
4.3.1	Horizontal velocity control	95
4.3.2	Vertical motion control	97
4.3.3	Position control	98
4.4	Summary	99
5	Summary and future work	101

List of Figures

3.1	Overall flight control scheme	42
3.2	Control scheme for the fuzzy gain scheduling method	43
3.3	Control scheme for the gradient descent method	44
3.4	Real function (solid), 3-mf approximation (dashed), 5-mf approximation(dotted)	50
3.5	Membership functions to approximate $\sin(x)$	51
3.6	Closed loop with an affine term as a “measurable” disturbance	56
3.7	Servo-actuator diagram including the Bell-Hiller mixer	61
3.8	Bode diagram and step response for the servo-actuator	61
3.9	Boundaries of inputs, and outputs for the servo-actuators	62
3.10	Membership functions $F_1^{(\cdot)}$ and $F_2^{(\cdot)}$	65
3.11	Fuzzy gain scheduler for the inner-loop	67
3.12	Rule for longitudinal speed with membership functions	68
3.13	Rule for lateral speed with membership functions	69
3.14	Rule for heading with membership functions	70
3.15	The Mamdani controller	70
4.1	Exp.1: Attitude set-point regulation	79
4.2	Input signals and main rotor force (left), and attitude angles (right) comparisons	80
4.3	Filter for the noise on the input signals for the attitude angles	80
4.4	Exp.2: Attitude tracking	81
4.5	Input signals and main rotor force (left), and attitude angles (right) comparisons	82
4.6	Exp.3: altitude control with constant wind and body mass	83
4.7	Input signals and main rotor force comparisons	83
4.8	Altitude-tracking with constant wind and body mass	84
4.9	Wind model and wind force components	85
4.10	Wind disturbance signal	85
4.11	Exp.4: Altitude-tracking with varying wind speed	86
4.12	Exp.5: Altitude-tracking with decreasing body mass	86
4.13	Exp.6: Altitude-tracking with wind and body mass changes	87
4.14	Exp.7: \dot{x} (left) and \dot{y} (right) set-point regulation	88
4.15	Exp.8: Sharp turns	89
4.16	Exp.9: Smooth turns	90
4.17	Exp.10: Rectangular pattern	91

4.18	Exp.11: Circular pattern	91
4.19	Fuzzy compensator scheme	92
4.20	Drift effect of wind changes on horizontal velocities	92
4.21	Wind action on the attitude angles: roll (left) and pitch (right)	93
4.22	Wind mapping for the roll (left) and pitch (right)	93
4.23	Wind mapping compensator scheme	94
4.24	Wind action with and without compensation	95
4.25	Exp.12: Low \dot{y} and high \dot{x} set-point regulation	95
4.26	Exp.13: Velocity tracking with strong and weak wind	96
4.27	Exp.14: \dot{y} and \dot{x} set-point change with strong and weak wind	97
4.28	Exp.15: Altitude set-point regulation and trajectory tracking.	98
4.29	Exp.16: Hovering control with strong and weak wind	98
4.30	Exp.17: Positioning with strong and weak wind	99

Chapter 1

Introduction

The central problem addressed in this thesis is the design of a control system that achieves *stable “aggressive” manoeuvrability for an unmanned helicopter*.

While almost all existing work in this area uses various modifications of *feedback linearization* we employ a gain-scheduling approach based on the use of Takagi-Sugeno fuzzy models [1], i.e., *fuzzy gain-scheduling*. However, here, we differ significantly from the conventional two-step gain-scheduling by proposing a one-step design – simultaneous synthesis of linear controllers and a gain scheduler with guaranteed global stability and robustness properties.

The experimental results showing the feasibility of the proposed fuzzy gain-scheduling approach are obtained via simulation using a mathematical model of the APID-MK3 unmanned helicopter, by Scandicraft Systems AB (www.scandicraft.se).

1.1 Motivation

Basically, there are two types of UAV autonomy: *functional* and *tactical*. The first type of autonomy addresses the execution of basic flight modes such as “take off”, “landing”, “cruise flight” as well as more aggressive flight patterns. Here, the major concern is twofold: 1) use and reliability of proprioceptive sensors (compass, GPS, gyros, etc.) to monitor the internal state of the UAV; and 2) robust and stable position/velocity control based on inputs from the UAV’s proprioceptive sensors. Thus, this concern is related to the air-worthiness of an UAV in unmanned flight and unmanned landing/take-off. The subject of this thesis is this type of tactical autonomy, and in particular, increasing the “aggressiveness” with which it is performed. We assume here the availability of reliable –within certain noise characteristics– proprioceptive sensors and focus on the

robustness and stability of attitude/altitude/velocity control via the use of fuzzy gain-scheduling.

The tactical type of autonomy addresses mission execution in a safe and reliable manner. Typical mission examples include “track ground vehicle”, “follow coast line”, “deliver load” and autonomy requires making as few assumptions as possible about the environment encountered during mission execution; and that execution should be sensitive to the environment, and adapt to the contingencies encountered. A typical example of a safety-related UAV behavior during mission execution is “sense and avoid”: it makes sure that collisions with elevated ground formations do not occur. Thus a major concern in achieving mission autonomy is the use of exteroceptive sensors, like a camera or a laser range finder, to acquire information about the state of the environment as it is at the moment and based on this information to react instantly to it by adopting a behavior that complies with this state alone. Therefore the success of the mission and the safety of an UAV is not dependent on delays in communication with a ground-station operator during which the possibility of a communication failure cannot be neglected. This type of autonomy is not of concern here. However one necessary condition for achieving it is for example, the ability to fly at varying speed and fast acceleration/deceleration capability.

1.1.1 Why aggressive manoeuvrability

The work reported in this thesis is a contribution to the overall objective of the *Wallenberg Laboratory for Information Technology and Autonomous Systems* (WITAS, www.ida.liu.se/ext/witas) at Linköping University: the development of an intelligent, deliberative/reactive command and control system, containing active-vision sensors, which supports the operation of a unmanned air vehicle (UAV) in both semi- and full-autonomy modes.

One of the UAV platforms of choice is the APID MK-III unmanned helicopter. The intended operational environment for APID MK-III is over widely varying geographical terrain with traffic networks and vehicle interaction of variable complexity, speed, and density. The present version of APID MK-III is capable of unmanned take-off, landing, hovering, and motion along linear trajectories with a constant low-speed. This is enough for performing missions like site mapping and surveillance, and electronic warfare and communications where the predominant flight modes used are hovering at predefined points and slowly moving, along a predefined straight-line, from one hovering point to another.

Other type of missions, e.g., tracking a ground vehicle, require the execution of curvilinear trajectories with a varying speed profile. However, the current control system for APID MK-III does not utilize large

ranges of the rotor attitude angles. As a consequence this produces lower rate-of-change of the body attitude angles. Consequently, the control is done on rather small ranges for these and this restricts the magnitude of the curvature of the trajectory which these angles can follow at a given relatively high speed. Furthermore, control within small ranges for the body attitude angles implies small acceleration rate – a shortcoming when a ground object is capable of accelerating at higher rates. Last but not least, the ability to decelerate fast is necessary for safe navigation when sudden unknown terrain elevations are encountered and have to be avoided as fast as possible.

In this context, the objective of achieving “aggressive” manoeuvrability concerns the design of a attitude/velocity/position controllers which act on much larger ranges of the body attitude angles, i.e., $-\pi/4 \leq \phi \leq +\pi/4$, $-\pi/4 \leq \theta \leq +\pi/4$, $-\pi \leq \psi \leq +\pi$, by utilizing the full range of the rotor attitude angles. The latter are approximated to the interval $[-0.25, +0.25]$ rad. These controllers should achieve robust and stable tracking of trajectories with varying curvature magnitude at relatively high speed. Take-off and landing are not considered here since APID MK-III has already have dedicated control modules that realize these flight modes.

It has to be noted here that the above interpretation of “aggressive manoeuvrability” agrees with the one given in [2], that is “ the ability to track *fast trajectories*”. A principally different reading of “aggressive manoeuvrability” is provided in [3] in: “the finite time transition between two *trim trajectories*”. Trim trajectories are defined as those trajectories along which the velocities in body axes (the twist) and the control input are constant.

1.1.2 Why fuzzy gain scheduling ?

A study of the relevant literature on unmanned helicopter control reveals very few well-documented case when a nonlinear model of an unmanned helicopter is deployed for the controller design. In all other cases the design is based on linear models and the linear control techniques used are μ -synthesis [4], H_∞ [5], or Linear Quadratic Gaussian LQG [6]. Examples include: a linear robust controller implemented on the Yamaha R-50 at Carnegie Mellon University. The controller consists of one MIMO loop for attitude stabilization and four separate SISO loops for velocity and position control. The speed of motion achieved is 4 m/s. Recent flight test can be found at www-2.cs.cmu.edu/marcol/research/flight_tests/html/flight_tests.html;

2) MIMO linear controller, based on μ -synthesis, is implemented on the Yamaha R-50 at the University of California at Berkeley for control during hover and way-point navigation. Recent flight test at speeds up to

6m/s can be found at <http://robotics.eecs.berkeley.edu/bear/>; 3) MIMO H_∞ and LQG hovering controllers [7] are implemented on the Caltech's Kyosho EP Concept electric model helicopter. It is important to emphasize on the fact that all linear designs are implemented and tested on the real platforms, while almost all nonlinear designs only are evaluated in simulation with the exception of the Georgia Tech controller implemented on the Yamaha-50 platform.

The predominant nonlinear controller designs are based on the notion of feedback linearization [8] of the original nonlinear helicopter model. The idea here is to transform the nonlinear dynamics into a linear form by using state feedback, with input-state linearization corresponding to complete linearization, and input-output linearization to partial linearization. It is the latter type of feedback linearization that is normally used for controller design in the case of unmanned helicopters.

Input-output linearization means the generation of a linear differential relation between the output and a new input. By means of this the dynamics of the original nonlinear system is decomposed into external (input-output) part and internal (unobservable) part. Since the external part consists of a linear relation between the output and the new input it is easy to design the input so that the output behaves as desired. Then the question is whether the internal part will also behave well, i.e., whether the internal states will remain bounded. The answer to this question is provided by studying the so-called zero-dynamics of the internal part, i.e., the dynamics when the control input is such that the output is maintained at zero. If an input-output linearized system has stable zero dynamics it is called minimum phase, and if it has unstable dynamics then it is a non-minimum phase. The control law for a minimum phase system can simply be obtained by model inversion. However this type of control law cannot be applied to non-minimum phase systems since they are not invertible. Thus the major focus in all reported controller designs for unmanned helicopters that are based on input-output linearization is: the generation of such input-output relation for the original nonlinear system so that the internal dynamics of the input-output linearized system is either minimum phase or it has no internal (zero-) dynamics. An input-output linearized system with no internal dynamics can be obtained as follows [9]: when performing successive differentiations of the selected output, to simply neglect the terms containing the input and keep differentiating the output a number of times equal to the system's order, so that there is "approximately" no internal dynamics. Of course this approach is only meaningful if the input coefficients at the intermediate steps are "small", i.e., the system is "weakly non-minimum phase", i.e., "fast" right-half plane zeros are neglected.

Controller designs based on input-output linearization have a number of important limitations amongst which the most important one, in the context of control of unmanned helicopters, is that no robustness is guaranteed in the presence of parameter uncertainties, unmodeled dynamics, or external disturbances [8]. In this context, the dynamic output fuzzy gain-scheduling controller [10] designed within the H_∞ framework and presented in this thesis allows to: 1) shape the closed loop transient dynamics so that it conforms to performance specifications; and 2) design a robust controller that rejects the influence of bounded model uncertainties and external disturbances. Yet another principal difference between fuzzy gain-scheduling and input-output linearization is that: fuzzy gain-scheduling design is a technique for transforming the original nonlinear system into another nonlinear system while input-output linearization transforms the original nonlinear system into a (fully or partially) linear system.

1.1.3 Why not conventional gain scheduling ?

The design of gain scheduled controllers [11] has, for a very long time, followed a two-step approach: first, the nonlinear model under control is linearized at a number of different operating points. These operating points may be different velocities, angles of attack, and altitudes. As a result one obtains a grid of working points according to the previously mentioned parameters and a linear model for each point in the grid. Then a linear controller is designed for each of the linear models in this set. When the flight conditions (altitude, velocity, angle of attack) change, the general control strategy should determine the working point in the grid to which these new conditions (approximately) correspond. The control action is performed by the linear controller which corresponds to this working point. Second, for points in the grid that do not have a corresponding linear controller a so-called *gain-scheduler* is designed via interpolation of the linear controllers in their neighborhood. The gain-scheduler is then used to perform the change from one linear controller to another that is, to control the system during the transition from one flight condition to another. This is called *gain-scheduling*. The weak part here is that each linear controller is only effective in a small neighborhood of its corresponding grid point (flight condition). Therefore one needs to verify that the change from one controller to another is smooth enough and doesn't cause instabilities. The most common approach is to leave this evaluation for the simulation stage.

In contrast to the above, fuzzy gain scheduling [12] is a one-step approach to the design of gain-schedulers: it provides for the simultaneous synthesis of linear controllers and a gain scheduler with guaranteed global

stability and robustness properties, thus avoiding the need for extensive simulation. It uses an approximation of the original nonlinear model in terms of a Takagi-Sugeno fuzzy system where the latter is a convex nonlinear combination of a set of linear models, hence the similarity with conventional gain-scheduling.

1.2 Related Work

In recent years, the design and implementation of control algorithms for unmanned helicopters has been the object of quite a number of studies. This is due to the recognized need for maneuverable autonomous air vehicles, for both military and civil applications. While slower and less efficient than airplanes, helicopters are capable of vertical take-off and landing, hover, and in general are more maneuverable in tight spaces than airplanes. As a consequence, helicopters are one of the best platforms for operations in urban or otherwise cluttered environments. However, in many respects the dynamics of a helicopter are more complicated than those of a fixed wing aircraft: a helicopter is inherently unstable at hover, and the flight characteristics change dramatically over the entire flight envelope.

In order to provide a proper framework within which the contributions of this thesis can be meaningfully evaluated we will only consider here studies that report in significant level of detail:

1. control algorithms that make use of nonlinear models and control techniques and their performance is evaluated either on the real platform or in simulation based on a mathematical model close enough to the real platform.

1.2.1 The Berkeley Aerorobot Team

The controller design [13] is based on a nonlinear model of the Ursa-Minor unmanned helicopter and its performance has so far been evaluated in simulation. In particular, the design is based on *approximate input-output linearization* of the original nonlinear helicopter model. “Approximate” here means that the input-output linearization is performed on the original nonlinear model only after the coupling effect between rolling moment and lateral force on one side and pitching moment and longitudinal force on the other are neglected. Then by choosing positions and heading as outputs and applying a dynamic decoupling algorithm [14] a linearized helicopter model which does not contain any unobservable zero-dynamics, and hence is minimum phase, is obtained. Finally, a tracking control law for this linearized model is designed and applied to the original nonlinear helicopter model so that a bounded tracking error is guaranteed. However, as noticed

in [15], the so-obtained tracking control law is sensitive to model disparities such as changes in the payload or the thrust-torque model, or external disturbances such as side-winds. Though, as mentioned in [10], the tracking error can be reduced by placing the poles further away from the origin in the left-half plane, this comes with a price – higher control input magnitude which may not be physically feasible. As already mentioned, this type of robustness limitations is inherent to all controller designs based on input-output linearization.

1.2.2 The Georgia Tech Aerial Robotics Mission

The design of an attitude controller is done on approximate linear model of the rotational dynamics and implemented on the Yamaha R-50 unmanned helicopter (see www.ae.gatech.edu/research/controls/labs/uavrf/). The controller design [16] uses linear model inversion. However, since the linear model is only approximation of the real dynamics of the helicopter it is subject to modeling errors arising from flight conditions and inaccurate modeling. Hence, an adaptive unit in the form of a neural network, is used to cancel the inversion errors using feedback and a stable update law based on Lyapunov stability theory. This same structure is used in all three channels, roll, pitch, and yaw. The adaptive unit also adjusts to changing atmospheric conditions and dynamics. Thus, the controller can be used at different points in the flight envelope without tuning.

1.2.3 The MIT Backstepping Controller

The backstepping controller design [17] is performed on the Berkeley UAV model and experimental results based on simulation are performed.

The major motivation is avoiding artificial singularities due to attitude dynamics representation via the use of Euler angles. These singularities arise when maneuvers like loops, barrel rolls, and split-ups are executed.

The helicopter model is approximated in the same manner as in [13]. Thus the approximated model is feedback linearizable or has differential flatness. On the basis of this model a non-trivial extension of backstepping ideas [18] is proposed. In its basic form backstepping is carried out on a chain of integrators (integrator backstepping). In this particular case backstepping is done on the group of rotations in the 3-D space rather in 2-D space. The design procedure avoids the introduction of artificial singularities through over parameterization of the outputs: full specification of the reference attitude is required. The so obtained backstepping controller is capable of tracking trim trajectories, e.g, climbing turn, and transitions between trim trajectories, e.g., a transition to inverted flight. The latter is the authors' interpretation of aggressive

maneuvering.

1.2.4 The Compiegne University Controller

The controller design [19] is done on a model valid for slow maneuvers (e.g., take-off and landing) close to hover and the control task is to track a trajectory given in position coordinates. The model describes a model helicopter (mass = 16kg) used in the unmanned helicopter project at Compiegne University of Technology, France. However the controller performance is evaluated in simulation.

The major effort is the derivation of a helicopter model in block pure feedback form so that backstepping or input-output linearization techniques can be used. In order to achieve this the helicopter is considered as a rigid body consisting of two parts: the helicopter airframe and additional load associated with the sensing and computer systems. The additional load is then distributed in such a way so that the moments of inertia of the helicopter around the first two principal axis corresponding to pitch and roll are equal. Using the tail rotor input to put to zero the rotation around the third principal axis, the reduced rigid body dynamics are simplified. Due to a specific structure of the inertia matrix, diagonal with the first two entries equal, this only requires an input sufficient to cancel the torques due to rotor drag and fully decouples the rotational dynamics of the system. The reduced dynamics obtained after this allows to define a point that acts as a center of oscillation for the airframe. The coordinates of this center of oscillation are not differentially flat outputs due to the presence of the parasitic torques associated with the rotor drag. However, taking these coordinates as the position of the airframe, the reduced dynamics can be rewritten in block pure feedback form with four integrations corresponding first to the translational dynamics and then to the rotational dynamics. From here a control law for almost exact tracking of the center of oscillations can be derived using backstepping or input-output linearization.

1.2.5 The Fuzzy Unmanned Helicopter

The work by Sugeno [20] reports a hierarchical, Mamdani-type of a controller for the unmanned helicopter Yamaha R-50 by Yamaha Motors. The lower layer contains a number of Mamdani-type control modules: longitudinal (pitch control), lateral (roll control), collective (vertical control), rudder (yaw control), and coupling compensation modules. Furthermore, within each such module there is a number of sub-modules only some of which correspond directly to our Mamdani-type controller from Sect. 4. These are as follows:

- Longitudinal: this module includes a \dot{x} Mamdani-type controller. The \dot{x} controller infers a desired

pitch angle using a velocity-error and its derivative and is identical to the one used by us;

- Lateral: this module includes a \dot{y} Mamdani-type controller. The \dot{y} controller infers a desired roll angle using a velocity-error and its derivative and is identical to the one used by us;
- Collective: this module includes a \dot{z} Mamdani-type controller. The \dot{z} controller infers a control value for the main collective using altitude, velocity-error and its derivative;
- Rudder: this module, given a desired heading, infers a control input for the tail collective using yaw angle error and its rate of change;
- Coupling compensation: the use of this module is twofold: i) it takes into account cross-couplings between longitudinal/lateral and vertical motion; ii) it takes into account cross-couplings between yaw and roll during a turn.

In Section 3.5.4, we make a detailed comparison between our outer-loop Mamdani-type controller and the one proposed by Sugeno, and point out important differences.

1.3 The Purpose of the Thesis

The purpose of this thesis is three-fold.

First, we aim at showing that the current limited number of flight modes that APID-MK3 is capable of can be extended to include “aggressive manoeuvrability” capabilities defined in terms of: 1) tracking curvilinear trajectories at high speed; and 2) fast acceleration/deceleration.

Second, we aim at showing the feasibility of controller design that is directly based on the nonlinear unmanned helicopter model and at the same time its stability can be guaranteed in a formal way. Furthermore, this design should be preferably done in a modern robust control framework, say H_∞ , which can be used for limiting the effect of model uncertainties and external disturbances.

Third, we aim at identifying the major limitations of fuzzy gain scheduling for the control of multiple-input multiple-output (MIMO) models of nonlinear dynamic systems by using it on a very realistic MIMO model of an intrinsically unstable unmanned helicopter. In addition, we illustrate how a fuzzy gain-scheduler can be used for performing aggressive flying.

We are completely aware that the “realism” of the results reported in this thesis w.r.t. the above aims is limited by the fact that all the work is performed in simulation. However, since the mathematical model used

in simulation is close enough, from control point of view, to the real APID-MK3 system we have all reasons to believe in the realism of these results. We can also mention here that for example, the reported results on work in advanced controller design for “aggressive” manoeuvrability in the BEAR project are also only tested and evaluated in simulation.

1.4 Publications

Parts of this thesis have been presented at international conferences. These are as follows:

1. Autonomous helicopter control using fuzzy gain scheduling; Kadmiry, B.; Bergsten, P.; Driankov, D. In: Proc. of the IEEE Int. Conf. on Robotic and Automation(ICRA), 3: pp. 2980–2985, May 2001, Seoul, Korea.
2. Autonomous Helicopter Control Using Gradient Descent Optimization Method; Kadmiry, B.; Palm R; Driankov, D. In: Proc. of the Asian Conf. on Robotic and Automation (ACRA), pp: 193–198; June 2001, Singapore.
3. Fuzzy control of an autonomous helicopter; Kadmiry, B.; Driankov, D. In: Proc. of the 9th IEEE Int. Fuzzy Systems Association (IFSA/NAFIPS) World Congress, 5: pp. 2797–2802; July 2001, Vancouver-Canada.
4. Autonomous Helicopter Control Using Linguistic and Model-based Fuzzy Control; Kadmiry, B.; Driankov, D. In: Proc. of the IEEE Int. Symposium on Intelligent Control (CCA / ISIC), pp: 348–352; Sept. 2001, Mexico-city-Mexico.

The last publication was awarded a second best student paper award.

1.5 Contributions

The major contributions in this thesis are as follows:

1. deriving the simplified version of the APID-MK3 model and comparing it to a general VTOL model, and the Berkeley and CMU unmanned helicopter models;
2. developing two novel nonlinear control methods for the design of a stable and robust -with respect to external disturbances- flight controllers for achieving aggressive manoeuvrability; and

3. performing extensive simulation in order to verify the previously mentioned properties of the flight controllers.

1.6 Outline of the thesis

The thesis is structured as follows. Chapter 2 is devoted to the mathematical model of APID-MK3. It presents this mathematical model and the assumptions under which it is derived. It also compares it to a general VTOL model and to the mathematical models developed in Berkeley and Carnegie-Mellon for similar unmanned platforms. In Chapter 3, we present the design of two different flight controllers. The fuzzy flight controller uses a combination of fuzzy gain scheduling (FGS) and heuristic fuzzy control in an inner- outer- control loops scheme. The gradient descent method (GDM) uses a combination of a gradient descent optimization and linear control. Both controllers are intended to realize aggressive flying subject to external disturbances. Chapter 4 presents results from extensive simulation and aimed at showing the robustness of the flight controllers and their capability to realize aggressive flying. The robustness properties are verified in simulations of the inner-loop controllers while aggressive flying is simulated via the use of the full inner- outer- control loops. In chapter 5, we summarize the thesis and present directions for future work.

Chapter 2

The helicopter model

The major aim of this chapter is to introduce the reader to the mathematical model of APID-MK3. The quality of the control algorithms, described in later chapters, as well as the validity of the simulation results depend heavily on how much the mathematical model reflects the real APID-MK3 platform. For this purpose, it is important to highlight the assumptions under which this model is derived, and also the simplifications made in order to facilitate the design of the flight controllers proposed here.

Yet another aim is to present comparisons between the APID-MK3 mathematical model, and the mathematical models for similar platforms in order –one more time– to assess the model’s quality. The other mathematical models considered here describe platforms that have actually been used in a unmanned flight mode.

In Section 2.1 we introduce in an informal manner the basic concepts related to the motion and control of a VTOL aircraft. Then in Section 2.2, we introduce the reader to the mathematical representation of the kinematics and dynamics of a general VTOL. Then in Section 2.3, the general kinematics and dynamics are further developed in order to obtain the mathematical models of motion that can be used for the purpose of control. Section 2.4 presents the mathematical model of APID-MK3 and compares it, where possible, to the general equations of motion developed in the previous sections. Finally, in Section 2.5, we compare the APID-MK3 mathematical model with models developed in Berkeley and Carnegie-Mellon for similar unmanned platforms.

2.1 Helicopter basic concepts

Helicopters are vertical take off and landing aircraft. They use rotating blades in order to create forces necessary to lift and move the helicopter body (VTOL). The helicopter has 6 D.O.F. which contribute to maintain the aircraft in normal flight position.

- Longitudinal motion: motion along the x-axis, described by the position x and the velocity \dot{x} ;
- Lateral motion: motion along the y-axis and described by the position y and the velocity \dot{y} ;
- Heave: vertical motion along the z-axis and described by the position z and the velocity \dot{z} ;
- Roll: obtained by rotation around the x-axis and described by the Euler angle ϕ and its rate $\dot{\phi}$;
- Pitch: obtained by rotation around the y-axis and described by the Euler angle θ and its rate $\dot{\theta}$; and
- Yaw: obtained by rotation around the z-axis and described by the Euler angle ψ and its rate $\dot{\psi}$.

The positions (x, y, z) and their time derivatives $(\dot{x}, \dot{y}, \dot{z})$ determine the helicopter's translational motion (longitudinal, lateral and heave motions) along the x-, y- and z-axis and are described in the inertial frame (\mathcal{R}_I). The angles (ϕ, θ, ψ) and their time derivatives $(\dot{\phi}, \dot{\theta}, \dot{\psi})$ determine the helicopter's attitude, defined as the orientation of the helicopter body frame (\mathcal{R}_B) w.r.t. the vehicle-carried vertical frame (\mathcal{R}_V). The latter frame, whose origin is the C.O.G. of the helicopter body, is oriented in the same way as the inertial frame (\mathcal{R}_I). Thus, we will call (\mathcal{R}_V) the inertial frame.

To perform the different types of motion, the helicopter depends on the main and tail rotor's forces and moments which relate to cabin aerodynamics $\vec{\mathcal{F}}_A$, main rotor aerodynamics $\vec{\mathcal{F}}_M$ and tail rotor aerodynamics $\vec{\mathcal{F}}_T$, as well as vertical $\vec{\mathcal{F}}_V$ and horizontal $\vec{\mathcal{F}}_H$ stabilizers. The previous defined forces are controlled as follows:

- main rotor collective pitch or "collective", increases the main rotor force ($\vec{\mathcal{F}}_M$), which gives the helicopter the possibility to perform ascend/descend (vertical) motion;
- cyclic pitch or "longitudinal cyclic", directs the main rotor force along the x-axis, which induces a longitudinal motion by inclining the main rotor force in the x-direction;
- cyclic roll or "lateral cyclic", directs on the main rotor force along the y-axis, which induces a lateral motion by inclining the main rotor force in the y-direction; and

- tail rotor collective pitch or “tail”, acts on the tail rotor force ($\vec{\mathcal{F}}_T$) to turn the helicopter around its main rotor axis (azimuthal turn).

Due to the construction characteristics of a helicopter, some of these control commands result in undesired motion, distinguishing the existence of cross-couplings. We list in the following some of these:

- the inclination of the main rotor force ($\vec{\mathcal{F}}_M$) from its vertical position reduces the lift force to the benefit of the trust force. Thus, the result is a loss of altitude. The lift force is the vertical component of the main rotor force while the trust is the horizontal component;
- due to transverse airflow ($\vec{\mathcal{F}}_W$), additional forces induce a tendency of the helicopter to pitch when in longitudinal motion, and roll when in lateral motion;
- the tail rotor force ($\vec{\mathcal{F}}_T$) may cause a lateral motion and a rolling moment -this induces a drift in the helicopter’s horizontal and vertical motions.

The cross-couplings mentioned above should be foreseen when designing a motion controller, it should be able to counterbalance the undesired motions caused by the cross-couplings.

The pilot action only refers to the action of the main and tail rotor force on the motion of the helicopter. In order to model this motion we have to consider the totality of forces and moments that apply to the helicopter body. These forces are mainly expressed in terms of wind action, gravity force, aerodynamics, and main and tail rotor force.

2.2 A general model

2.2.1 General kinematics and dynamics

In this section we will describe the kinematics and dynamics of a general VTOL. These two components are necessary to determine the motion of the helicopter in terms of position, velocity, and attitude through the knowledge of forces and moments.

Kinematics

The kinematic equations relate the descriptions of velocities and angular speeds (the rates of roll, pitch and yaw). These descriptions are done in both inertial (\mathcal{R}_I) and body (\mathcal{R}_B) frame as follows:

$$\begin{aligned} \dot{\vec{\eta}} = J(\Theta)\vec{\mathcal{V}} \text{ with } \vec{\Theta} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \text{ and } J(\Theta) = \begin{bmatrix} \mathcal{R}_{IB} & 0_3 \\ 0_3 & \mathcal{R}_\Omega \end{bmatrix} & \quad (2.1) \\ \dot{\vec{\eta}} = \begin{bmatrix} \vec{V}_I \\ \vec{\Omega}_I \end{bmatrix} \text{ with, } \vec{V}_I = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \text{ and } \vec{\Omega}_I = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \\ \vec{\mathcal{V}} = \begin{bmatrix} \vec{V}_B \\ \vec{\Omega}_B \end{bmatrix} \text{ with, } \vec{V}_B = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \text{ and } \vec{\Omega}_B = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \end{aligned}$$

where

- $\vec{\eta}$ is a vector of state-variables expressed in terms of position (x, y, z) and attitude (ϕ, θ, ψ) . $\dot{\vec{\eta}}$ is its derivative w.r.t. the inertial frame;
- $\vec{\mathcal{V}}$ is the vector of state-variables expressed in terms of velocity (u, v, w) and attitude rate (p, q, r) . $\vec{\mathcal{V}}$ is the time derivative of $\vec{\eta}$ described w.r.t. to the body frame; and
- $J(\Theta)$ is the operator which transforms the state-variables from body frame to the inertial frame.
- \mathcal{R}_{IB} and \mathcal{R}_Ω are the transformation matrices for both the translation and rotation vectors respectively, between inertial and body frames.

The advantage of this parameterization is that it allows for a direct measurement of the angular speeds (p, q, r) and their accelerations $(\dot{p}, \dot{q}, \dot{r})$. This is done by using on-board sensors such as magnetic compass, inclinometers, and gyros. The velocities and positions are given by accelerometers, GPS, and altimeter.

Helicopter dynamics

The dynamics of the helicopter body are related to the forces acting on it. These forces generate accelerations according to the general rules of forces and moments defined as follows:

$$\begin{aligned} \dot{\vec{V}} &= \lambda \sum \vec{\mathcal{T}} && \text{with} && (2.2) \\ \dot{\vec{V}} &= \begin{bmatrix} \dot{V}_B \\ \dot{\vec{\Omega}}_B \end{bmatrix} && \text{with} && \dot{V}_B = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} && \text{and} && \dot{\vec{\Omega}}_B = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}, \\ \sum \vec{\mathcal{T}} &= \begin{bmatrix} \vec{\mathcal{F}} \\ \vec{\mathcal{M}} \end{bmatrix} && \text{with} && \vec{\mathcal{F}} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} && \text{and} && \vec{\mathcal{M}} = \begin{bmatrix} R \\ M \\ N \end{bmatrix}. \end{aligned}$$

where

- $\dot{\vec{V}}$ represents the state-variables in terms of the time derivative of the vector speed $\dot{V}_B = (\dot{u}, \dot{v}, \dot{w})$ and angular rates $\dot{\vec{\Omega}}_B = (\dot{p}, \dot{q}, \dot{r})$ both derived and represented in the body frame;
- $\sum \vec{\mathcal{T}}$ is a sum of forces (X, Y, Z) and moments (R, M, N) describing system inputs that produce motion; and
- λ is an intrinsic constant related to the helicopter characteristics (inertia) in terms of mass (m) and angular moments (\mathcal{I}) .

The parameterization described above is frame dependent. The state-variables are provided by the sensors.

Relation between kinematics and dynamics

The model including both kinematics and dynamics is given by a sum of the forces, and a sum of the moments, w.r.t. the body frame. Thus the helicopter's translational and angular motion equations are given as follows:

$$\begin{aligned} V_I &= \mathcal{R}_{IB} V_B; && \Omega_I &= \mathcal{R}_{\Omega} \Omega_B \\ \vec{\mathcal{F}}|_B &= \frac{d}{dt}|_I (m \vec{V}_B) && \text{and} && \vec{\mathcal{M}}|_B &= \frac{d}{dt}|_I (\mathcal{I} \vec{\Omega}_B) \end{aligned} \quad (2.3)$$

where $\vec{\mathcal{F}}|_B$ is the sum of forces and $\vec{\mathcal{M}}|_B$ is the sum of moments acting on the helicopter body, both described in the body frame. m is the mass of the helicopter including fuel and sensory platform, and \mathcal{I} the moment of

inertia of the helicopter body. Eq. (2.3) can be rewritten as

$$\begin{aligned}
V_I|_I &= \mathcal{R}_{IB}V_B|_I; & \Omega_I|_I &= \mathcal{R}_\Omega\Omega_B|_I \\
\dot{V}_B|_I &= \dot{V}_B|_B + \Omega_B \times mV_B; & \dot{\Omega}_B|_I &= \dot{\Omega}_B|_B + \Omega_B \times m\Omega_B = \dot{\Omega}_B|_B \\
\vec{\mathcal{F}}|_B &= m\dot{V}_B|_I & \text{and} & \vec{\mathcal{M}}|_B &= \mathcal{I}\dot{\Omega}_B|_I
\end{aligned} \tag{2.4}$$

where $\dot{V}_B|_I$ is the derivative of the velocity “defined” in the body frame and “derived” in the inertial frame.

2.2.2 Nature of the model

Basically, there exist two flight modes:

- The hover mode: steady positioning of the helicopter at a certain position.
- The free flight mode: consisting of the fore/aft (longitudinal), side-ward (lateral) and up/downward (vertical) motions.

The mathematical model that can be used to stabilize a VTOL in a hover mode is obtained from Eq. (2.4) by neglecting the Coriolis $\Omega_B \times mV_B$ and gyroscopic $\Omega_B \times \mathcal{I}\Omega_B$ components, because the attitude rates (p, q, r) and the translational speed (u, v, w) are very small in the hover mode.

$$\begin{aligned}
\vec{\mathcal{F}}|_B &= m\dot{V}_B|_I = m\dot{V}_B|_B; & \text{and} \\
\vec{\mathcal{M}}|_B &= \mathcal{I}\dot{\Omega}_B|_I = \mathcal{I}\dot{\Omega}_B|_B
\end{aligned} \tag{2.5}$$

As it can be seen from above, only the translational acceleration $\dot{V}_B|_B = (\dot{u}, \dot{v}, \dot{w})$ and rotational acceleration $\dot{\Omega}_B|_B = (\dot{p}, \dot{q}, \dot{r})$ w.r.t. the body frame are considered.

In order to be able to perform free flight, then the full model in Eq. (2.4) should be considered. The APID-MK3 model to be presented in later sections corresponds to the one given by Eq. (2.5). That is, it can be used for stabilization in hover mode. However, we use Eq. (2.5) also to design motion controllers for free flight. Then the question is: Is this correct, since Eq. (2.5) lacks the Coriolis and gyroscopic components? The answer is yes, because when Eq. (2.5) is transformed from the body to the inertial frame, then the previously mentioned components are recovered as illustrated below. From the hover model we have that

$$m\dot{V}_B|_B = \vec{\mathcal{F}}|_B;$$

the equivalent expression in the inertial frame will be

$$m\dot{V}_I|_I = \vec{\mathcal{F}}|_I; \quad (2.6)$$

where, from the use of the kinematic relation (see Eq. (2.1))

$$V_I|_I = \mathcal{R}_{IB}V_B|_I \implies \dot{V}_I|_I = \dot{\mathcal{R}}_{IB}V_B|_I + \mathcal{R}_{IB}\dot{V}_B|_I \quad (2.7)$$

Furthermore, the derivative of the transformation matrix $\dot{\mathcal{R}}_{IB}$ is given as the operator

$$\dot{\mathcal{R}}_{IB}(\cdot) = \mathcal{R}_{IB}\Omega_B \times (\cdot) \quad (2.8)$$

where (\cdot) is any vector. When $\dot{\mathcal{R}}_{IB}(\cdot)$ is injected in Eq. (2.7), this gives

$$\dot{V}_I|_I = \mathcal{R}_{IB}\Omega_B \times V_B|_I + \mathcal{R}_{IB}\dot{V}_B|_I \quad (2.9)$$

The law of composition of accelerations states that

$$\dot{V}_B|_I = \dot{V}_B|_B + \Omega_B \times V_B|_B \quad (2.10)$$

Thus, we obtain

$$\dot{V}_I|_I = \mathcal{R}_{IB}\Omega_B \times V_B|_I + \mathcal{R}_{IB}(\dot{V}_B|_B + \Omega_B \times V_B|_B) \quad (2.11)$$

The last term in this equation contains the Coriolis component $\Omega_B \times mV_B|_B$ that was missing from the expression of forces w.r.t. the body frame in the hover model. The missing gyroscopic component $\Omega_B \times \mathcal{I}\Omega_B$ in the expression of moments for the hover model can be recovered in the same manner once this expression is transformed in the inertial frame.

2.3 Equations of motion

2.3.1 Translational motion

We introduce here the equations that produce translational motion. The forces producing this motion constitute the term $\vec{\mathcal{F}}|_B$ in Eq. (2.4): are respectively. These are:

- the gravity $\vec{\mathcal{F}}_G$, acting on the body mass with a constant gravity acceleration g , described w.r.t. the inertial frame \mathcal{R}_I as $\vec{\mathcal{F}}_G|_I = (0, 0, mg)$;

- the wind action $\vec{\mathcal{F}}_W$ in terms of wind disturbances N_W in the north, east and downward directions, thus expressed w.r.t. \mathcal{R}_I as $\vec{\mathcal{F}}_W|_I = (W_N, W_E, W_D)$;
- the aerodynamic $\vec{\mathcal{F}}_A$ resulting from the action of the wind on the cabin produces: cabin drag D_C , cabin side force Y_C and cabin lift force L_C . All of these are given w.r.t. the airframe \mathcal{R}_A as $\vec{\mathcal{F}}_A|_A = (-D_C, Y_C, -L_C)$.
- the main rotor force $\vec{\mathcal{F}}_M$ results from the lift force L_M , side force Y_M and drag force D_M generated by the main rotor blades, described w.r.t. the main rotor frame \mathcal{R}_R as $\vec{\mathcal{F}}_M|_R = (D_M, Y_M, L_M)$.
- the tail rotor force $\vec{\mathcal{F}}_T$ results from the lift force L_T , side force T_T and drag force D_T generated by the tail rotor blades, described w.r.t. the tail rotor frame \mathcal{R}_T as $\vec{\mathcal{F}}_T|_T = (D_M, Y_M, L_T)$.

Summing these forces, after a transformation to the body frame, we obtain

$$\begin{aligned}
\vec{\mathcal{F}}|_B &= \vec{\mathcal{F}}_W|_B + \vec{\mathcal{F}}_G|_B + \vec{\mathcal{F}}_A|_B + \vec{\mathcal{F}}_M|_B + \vec{\mathcal{F}}_T|_B + \vec{\mathcal{F}}_C|_B \quad \text{with} \\
\vec{\mathcal{F}}_C|_B &= \begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix}; & \vec{\mathcal{F}}_T|_B &= \mathcal{R}_{BT} \begin{bmatrix} D_T \\ Y_T \\ L_T \end{bmatrix} \\
\vec{\mathcal{F}}_G|_B &= \mathcal{R}_{BI} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}; & \vec{\mathcal{F}}_A|_B &= \mathcal{R}_{BA} \begin{bmatrix} -D_C \\ Y_C \\ -L_C \end{bmatrix} \\
\vec{\mathcal{F}}_W|_B &= \mathcal{R}_{BI} \begin{bmatrix} W_N \\ W_E \\ W_D \end{bmatrix} & \text{and } \vec{\mathcal{F}}_M|_B &= \mathcal{R}_{BR} \begin{bmatrix} D_M \\ Y_M \\ L_M \end{bmatrix}
\end{aligned} \tag{2.12}$$

The above expression represented in the inertial frame reads as

$$\begin{aligned}
\vec{\mathcal{F}}|_I &= \vec{\mathcal{F}}_G|_I + \vec{\mathcal{F}}_W|_I + \vec{\mathcal{F}}_A|_I + \vec{\mathcal{F}}_M|_I \quad \text{with} \\
\vec{\mathcal{F}}_G|_I &= \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}; & \vec{\mathcal{F}}_A|_I &= \mathcal{R}_{IB}\mathcal{R}_{BA} \begin{bmatrix} -D_C \\ Y_C \\ -L_C \end{bmatrix} \\
\vec{\mathcal{F}}_W|_I &= \begin{bmatrix} W_N \\ W_E \\ W_D \end{bmatrix} & \text{and } \vec{\mathcal{F}}_M|_I &= \mathcal{R}_{IB}\mathcal{R}_{BR} \begin{bmatrix} D_M \\ Y_M \\ L_M \end{bmatrix}
\end{aligned} \tag{2.13}$$

In the case of the mathematical model of APID-MK3, we have the following assumptions regarding Eqs. (2.12) and (2.13):

1. The main rotor rotation speed Ω_M is assumed constant for the sake of simplicity (the main rotor force is a function of Ω_M^2).
2. The wind action is simply expressed as a white noise $N_w(0, V_W)$. It is considered as a perturbation for the control to compensate for.
3. The gravity g is supposed constant and depends slightly on the altitude $z \in [0, 200]m$ in the case of our platform.
4. We consider the body mass constant and concentrated in the C.O.G. of the helicopter, assumed fixed. The body mass change in time is considered as an external disturbance.
5. The cabin is assumed of spherical shape. The aerodynamic force $\vec{\mathcal{F}}_A$ is computed from this characteristic and the airspeed V_A . Thus its action is equal in all directions, and its description w.r.t. any frame is then the same.
6. Generally, the rotor is approximated as a rigid disc. The aerodynamics of the rotor depend on the aerodynamics of each of its blades. Because of the symmetry, the drag force D_M (and side force Y_M) of each blade eliminate the one generated by the other blade.
7. For the same reason, the components of the lift force L_M in the rotor disc plane eliminate each other, and only the component on the rotor z-axis (orthogonal to the disc plane) remains.
8. Since the tail rotor is also approximated as a rigid disc. The tail rotor $\vec{\mathcal{F}}_T$ is described in the same manner as $\vec{\mathcal{F}}_M$.
9. The action of the tail rotor $\vec{\mathcal{F}}_T$ is not represented in the equation of lateral motion, though it is responsible for a lateral drift of the helicopter when it rolls.
10. The Coriolis force $\vec{\mathcal{F}}_C$ is neglected in the description of the hover model of motion, due to small attitude rates and speeds.

Taking assumption 2 into account, the expression of the wind force in both the inertial and body frames is given as

$$\begin{aligned} \vec{\mathcal{F}}_W|_I &= \begin{bmatrix} N_w(0, V_N) \\ N_w(0, V_E) \\ N_w(0, V_D) \end{bmatrix} \quad \text{and} \\ \vec{\mathcal{F}}_W|_B &= \mathcal{R}_{BI}\vec{\mathcal{F}}_W|_I \iff \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} = \mathcal{R}_{BI} \begin{bmatrix} N_w(0, V_N) \\ N_w(0, V_E) \\ N_w(0, V_D) \end{bmatrix} \end{aligned} \quad (2.14)$$

Taking assumptions 3 and 4 into account, the expression of the gravity in both inertial and body frames is given as

$$\begin{aligned} \vec{\mathcal{F}}_G|_I &= \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad \text{and} \quad \vec{\mathcal{F}}_G|_B = \mathcal{R}_{BI}\vec{\mathcal{F}}_G|_I \\ \vec{\mathcal{F}}_G|_B &= \begin{bmatrix} X_G \\ Y_G \\ Z_G \end{bmatrix} = mg \begin{bmatrix} -\sin\theta \\ \sin\phi \cos\theta \\ \cos\phi \cos\theta \end{bmatrix} \end{aligned} \quad (2.15)$$

Taking into account assumption 5, that is $-D_C = Y_C = -L_C$, the cabin aerodynamics $\vec{\mathcal{F}}_A$ are given as follows

$$\begin{aligned} D_C = \frac{1}{2}C_dAV_A^2 = K_dV_A^2 \implies \vec{\mathcal{F}}_A|_A &= K_dV_A^2 \begin{bmatrix} -1 \\ 1 \\ -1 \end{bmatrix}; \quad \text{and} \quad \vec{\mathcal{F}}_A|_B = \mathcal{R}_{AB}\vec{\mathcal{F}}_A|_A \\ \iff \vec{\mathcal{F}}_A|_B = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} &= K_dV_A^2 \begin{bmatrix} -1 \\ 1 \\ -1 \end{bmatrix} \quad \text{and} \quad \vec{\mathcal{F}}_A|_I = \mathcal{R}_{IB}\vec{\mathcal{F}}_A|_B = K_dV_W^2 \begin{bmatrix} -1 \\ 1 \\ -1 \end{bmatrix} \end{aligned}$$

where C_d is the drag coefficient, $A = 4\pi R_C^2$ is the area of the cabin, R_C is the radius of the cabin and $K_d = 2C_d\pi R_C^2$ is a constant depending on the previously mentioned parameters. $V_A|_B = V_B|_B + \mathcal{R}_{BI}V_W|_I$ is the airspeed of the helicopter. Because of the model of motion is in hover mode, the assumption of small velocities implies $V_A \approx V_W$,

From assumption 6 and 7, we conclude that it follows that $\vec{\mathcal{F}}_M|_R = (0, 0, L_M)$. Furthermore, taking also into account the tilt of the main rotor, the cyclic pitch and roll enter into the description of the main rotor

forces as follows:

$$\begin{aligned}
L_M = K_M \Omega_M^2 \theta_M &\implies \vec{\mathcal{F}}_M|_R = \begin{bmatrix} 0 \\ 0 \\ L_M \end{bmatrix} \quad \text{and} \quad \vec{\mathcal{F}}_M|_B = \mathcal{R}_{BR} \vec{\mathcal{F}}_M|_R \\
\implies \vec{\mathcal{F}}_M|_B = \begin{bmatrix} X_M \\ Y_M \\ Z_M \end{bmatrix} &= K_M \Omega_M^2 \theta_M \begin{bmatrix} -\sin a_{1s} \\ \sin b_{1s} \\ -\cos a_{1s} \cos b_{1s} \end{bmatrix}
\end{aligned} \tag{2.16}$$

where θ_M is the collective pitch and K_M is a constant involving aerodynamic parameters of the main rotor. The effect of (X_M, Y_M) is neglected in the equations of translational motion because of small cyclics (a_{1s}, b_{1s}) . However, it will be used in the attitude equations since as the moments induced by them cannot be neglected. Thus we have that

$$\begin{aligned}
\vec{\mathcal{F}}_M|_B &\approx \begin{bmatrix} 0 \\ 0 \\ -K_M \Omega_M^2 \theta_M \end{bmatrix} \quad \text{and} \quad \vec{\mathcal{F}}_M|_I = \mathcal{R}_{IB} \vec{\mathcal{F}}_M|_B \\
\implies \vec{\mathcal{F}}_M|_I &= -K_M \Omega_M^2 \theta_M \begin{bmatrix} \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi \\ \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi \\ \cos \phi \cos \theta \end{bmatrix}
\end{aligned} \tag{2.17}$$

Assumption 8 results in

$$\begin{aligned}
L_T = K_\psi (\theta_T - \psi_T) &\implies \vec{\mathcal{F}}_T|_T = \begin{bmatrix} 0 \\ 0 \\ L_T \end{bmatrix} \quad \text{and} \quad \vec{\mathcal{F}}_T|_B = \mathcal{R}_{BT} \vec{\mathcal{F}}_T|_T \\
\implies \vec{\mathcal{F}}_T|_B = \begin{bmatrix} X_T \\ Y_T \\ Z_T \end{bmatrix} &= \begin{bmatrix} 0 \\ -L_T \\ 0 \end{bmatrix}
\end{aligned} \tag{2.18}$$

Assumption 9 states the use of $\vec{\mathcal{F}}_T|_B$ to counteract the effect of cabin spin due to the main rotor rotation. This force is neglected in the force equations, but will be considered in the equations of moments.

Finally the expression of Coriolis force $\vec{\mathcal{F}}_C|_B$ is given w.r.t. the body frame as

$$\vec{\mathcal{F}}_C|_B = \Omega_B \times mV_B \implies \vec{\mathcal{F}}_C|_B = \begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = m \begin{bmatrix} vr - wq \\ wp - ur \\ uq - pv \end{bmatrix} \tag{2.19}$$

Assumption 10 neglects the Coriolis force in the description of the hover model of motion. Thus, summing up Eqs. (2.14) to (2.17), we obtain the model of translational motion for APID-MK3 in the inertial frame.

Couplings between translational motions

From Eqs. (2.14) to (2.17), the only existing cross-coupling is the one between longitudinal and lateral motions. This is due to cross-coupling between attitude angles through the use of the transformation matrix \mathcal{R}_{IB} in Eq. (2.17).

2.3.2 Rotational motion

We introduce here the equations that describe rotational motion. The moments producing this motion and involved in $\vec{\mathcal{M}}|_B$ in Eq. (2.4) are:

- The anti-torque $\vec{\mathcal{M}}_D|_B$ resists the cabin rotation and is expressed in the inertial frame.
- The rotor aerodynamics moment $\vec{\mathcal{M}}_M|_B$ is produced by the main rotor force and is given in the main rotor frame.
- The tail rotor moment $\vec{\mathcal{M}}_T|_B$ is created in the C.O.G. of the cabin by the tail rotor force $\vec{\mathcal{F}}_T$, and is given in the tail rotor frame.

Thus, $\vec{\mathcal{M}}|_B$ can be expressed w.r.t. the body frame as follows

$$\vec{\mathcal{M}}|_B = \vec{\mathcal{M}}_W|_B + \vec{\mathcal{M}}_G|_B + \vec{\mathcal{M}}_A|_B + \vec{\mathcal{M}}_M|_B + \vec{\mathcal{M}}_T|_B + \vec{\mathcal{M}}_C|_B \quad (2.20)$$

The equivalent expression in the inertial frame is

$$\vec{\mathcal{M}}|_I = \vec{\mathcal{M}}_W|_I + \vec{\mathcal{M}}_G|_I + \vec{\mathcal{M}}_A|_I + \vec{\mathcal{M}}_M|_I + \vec{\mathcal{M}}_T|_I + \vec{\mathcal{M}}_C|_I \quad (2.21)$$

In the case of the mathematical model of APID-MK3, we have the following assumptions regarding Eqs. (2.20) and (2.21):

1. The moment of inertia \mathcal{I} is represented as a diagonal matrix, which implies that no coupling between the attitude angles is assumed.
2. Generally, the C.O.G. is always assumed to be in a fixed position, and this permits a computation of the moments of all the forces involved in the model.
3. Generally, the gravity force acts on the C.O.G., it does not generate any rotational moment and thus, it has no impact on the rotational motion.

4. The aerodynamics $\vec{\mathcal{F}}_A|_B$ applied on the helicopter cabin are located in its C.O.G. and because of this, it has no impact on the rotational motion.
5. The action of the wind force on the tail boom is neglected, thus no moment is generated.
6. When the main rotor tilts, its force $\vec{\mathcal{F}}_M|_B$ generates the moment $\vec{\mathcal{M}}_M|_B$. This moment is function of $\vec{\mathcal{F}}_M|_B$ and the distance between the rotor hinge and the C.O.G. of the body H_M .
7. The tail rotor center position is supposed to be axial to the x-axis, and of distance $-h_T$ from the C.O.G. ($H_T = (-l_T, 0, 0)$). The main rotor center position is axial to the z-axis and of distance $-h_M$ from the C.O.G. ($H_M = (0, 0, -h_M)$).
8. The main rotor spin creates an anti-torque which induces yaw turns. This anti-torque Q_M is damped by an off-set on the yaw (ψ_T) in order to balance the helicopter's directional turn. Thus, this anti-torque will not be represented in the APID model of motion $Q_M = 0$.
9. The tail rotor spin creates an anti-torque which induces roll turns, because directed in the y-axis only. This anti-torque Q_T is not damped generally. We will not consider this anti-torque in our equations $Q_T = 0$.
10. There exists centrifugal forces on the main rotor blades, which create centrifugal moments. This is due to the rotor hinge configuration. Because of our assumption on the rotor geometry, we will not consider these centrifugal forces/moments.
11. Besides the geometry of the main rotor, we assume that the hinge radius is very small ($e \approx 0$).
12. The gyroscopic moment $\vec{\mathcal{M}}_C$ is neglected in the description of the hover model of motion, due to small attitude rates.
13. Eq. (2.4) stresses the fact that the rotational accelerations derived w.r.t. the inertial frame are equal to the ones derived in the body frame.

Assumptions 2 and 3 imply that $\vec{\mathcal{M}}_G|_B = 0$. Assumptions 2 and 4 imply that $\vec{\mathcal{M}}_A|_B = 0$. Assumptions 2 and 5 imply that $\vec{\mathcal{M}}_W|_B = 0$.

Assumption 2 is used to determine the moments created by both the main and tail rotors' forces. This is done using the distance between C.O.G. and the main hinge center H_M , and between C.O.G. and tail hinge

center H_T . Thus in the case of the main rotor we have

$$\begin{aligned} \vec{\mathcal{M}}_M|_B &= H_M \times \vec{\mathcal{F}}_M|_B; \quad \text{with } H_M = \begin{bmatrix} 0 \\ 0 \\ -h_M \end{bmatrix} \quad \text{and} \\ \vec{\mathcal{F}}_M|_B &= K_M \Omega_M^2 \theta_M \begin{bmatrix} -\sin a_{1s} \\ \sin b_{1s} \\ -\cos a_{1s} \cos b_{1s} \end{bmatrix} \implies \vec{\mathcal{M}}_M|_B = \begin{bmatrix} R_M \\ M_M \\ N_M \end{bmatrix} = \begin{bmatrix} K_M \Omega_M^2 \theta_M \sin b_{1s} h_M \\ -K_M \Omega_M^2 \theta_M \sin a_{1s} h_M \\ 0 \end{bmatrix} \end{aligned} \quad (2.22)$$

As to the tail rotor, using the assumption 7, we obtain

$$\begin{aligned} \vec{\mathcal{M}}_T|_B &= H_T \times \vec{\mathcal{F}}_T|_B; \quad \text{with } H_T = \begin{bmatrix} -l_T \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \\ \vec{\mathcal{F}}_T|_B &= K_\psi (\theta_M - \psi_T) \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \implies T_{oT}|_B = \begin{bmatrix} R_T \\ M_T \\ N_T \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ K_\psi (\theta_M - \psi_T) l_T \end{bmatrix} \end{aligned} \quad (2.23)$$

Assumption 1 is used to describe the moment, due to inertia, on the attitude rates as a linear function on theses. That is

$$\vec{\mathcal{M}}_D|_B = \begin{bmatrix} R_M \\ M_M \\ N_M \end{bmatrix} = \begin{bmatrix} d_\phi \dot{\phi} \\ d_\theta \dot{\theta} \\ d_\psi \dot{\psi} \end{bmatrix} \quad (2.24)$$

where $(d_\phi, d_\theta, d_\psi)$ depend on the matrix of inertia of the cabin and given drag moment coefficients.

Finally, based on assumption 1, we can express the gyroscopic moment w.r.t. the body frame as

$$\vec{\mathcal{M}}_C|_B = \Omega(\Omega \mathcal{I}) \implies \vec{\mathcal{M}}_C|_B = \begin{bmatrix} qr(\mathcal{I}_y - \mathcal{I}_z) \\ pr(\mathcal{I}_z - \mathcal{I}_x) \\ pq(-\mathcal{I}_x - \mathcal{I}_y) \end{bmatrix} \quad (2.25)$$

Assumption 12 neglects the effect of this moment in the equations of rotational motion. Assumption 13 will be used to support an approximation by which Euler angles rates and attitude angle rates are made equal $(\dot{\phi}, \dot{\theta}, \dot{\psi}) \approx (\dot{p}, \dot{q}, \dot{r})$. Thus, this allows the equations for rotational motion in the body frame to be changed for those in the inertial frame. Summing up the terms w.r.t. the inertial frame in Eqs. (2.22),(2.23) and (2.24), we obtain the model for rotational motion for APID-MK3.

Couplings between rotational motions

From Eqs. (2.22) to (2.23), it is obvious that we do not consider any coupling between the attitude angles or their rates. However there a coupling between main-collective and the longitudinal cyclic. See Eq. (2.22). This in turn causes a coupling between rotational and translational motion, in particular pitch may cause a loss of altitude. Similarly, we have the same effect of roll angle on the altitude, because of the couplings between the main collective and the lateral cyclic.

2.4 Mathematical model of APID-MK3

2.4.1 APID-MK3 general model

The model for APID-MK3, derived w.r.t. the inertial frame is as follows

$$\begin{aligned}
\ddot{x} &= \frac{1}{m}(N_w(0, V_N) - K_d V_W^2 - K_M \Omega_M^2 \theta_M (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)) \\
\ddot{y} &= \frac{1}{m}(N_w(0, V_E) + K_d V_W^2 - K_M \Omega_M^2 \theta_M (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)) \\
\ddot{z} &= \frac{1}{m}(N_w(0, V_D) + g - K_d V_W^2 - K_M \Omega_M^2 (\theta_M \cos \phi \cos \theta)) \\
\ddot{\phi} &= \frac{1}{\mathcal{I}_x}(d_\phi \dot{\phi} + K_M \Omega_M^2 h_M \theta_M b_{1s}) \\
\ddot{\theta} &= \frac{1}{\mathcal{I}_y}(d_\theta \dot{\theta} - K_M \Omega_M^2 h_M \theta_M a_{1s}) \\
\ddot{\psi} &= \frac{1}{\mathcal{I}_z}(d_\psi \dot{\psi} + K_\psi (\theta_T - \psi_T))
\end{aligned} \tag{2.26}$$

It has to be noticed that this model is a simplified version of the original model of APID-MK3. The latter is given as

$$\begin{aligned}
\ddot{x} &= \frac{1}{m}(N_w(0, V_N) - K_d V_W^2 - T_M (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)) \\
\ddot{y} &= \frac{1}{m}(N_w(0, V_E) + K_d V_W^2 - T_M (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)) \\
\ddot{z} &= \frac{1}{m}(N_w(0, V_D) + g - K_d V_W^2 - T_M (\theta_M \cos \phi \cos \theta)) \\
\ddot{\phi} &= \frac{1}{\mathcal{I}_x}(d_\phi \dot{\phi} + F_\phi h_M) \\
\ddot{\theta} &= \frac{1}{\mathcal{I}_y}(d_\theta \dot{\theta} + F_\theta h_M) \\
\ddot{\psi} &= \frac{1}{\mathcal{I}_z}(d_\psi \dot{\psi} + K_\psi (\theta_T - \psi_T))
\end{aligned} \tag{2.27}$$

where T_M is the rotor trust, T_ϕ and T_θ are the forces obtained by rotation of the main rotor, and are given by

$$\begin{aligned} T_M &= \cos(\arcsin b_{1s}) \cos(\arcsin a_{1s}) F_M \\ T_\phi &= \sin(\arcsin b_{1s}) \cos(\arcsin a_{1s}) F_M \\ T_\theta &= -\sin(\arcsin a_{1s}) F_M \end{aligned} \quad (2.28)$$

and $F_M = K_M \Omega_M^2 \theta_M$. The assumption of small angles ($\approx [-0.25, 0.25]rad$) applied to the cyclics a_{1s} and b_{1s} implies a simplification as follows:

$$\begin{aligned} T_M &\approx \cos(b_{1s}) \cos(a_{1s}) F_M \Leftrightarrow T_M \approx K_M \Omega_M^2 \theta_M \\ T_\phi &\approx \sin(b_{1s}) \cos(a_{1s}) F_M \Leftrightarrow T_\phi \approx K_M \Omega_M^2 \theta_M b_{1s} \\ T_\theta &\approx -\sin(a_{1s}) F_M \Leftrightarrow T_\theta \approx -K_M \Omega_M^2 \theta_M a_{1s} \end{aligned} \quad (2.29)$$

which leads to the expressions presented in Eq. (2.26). Furthermore, the control inputs are given as

$$\begin{aligned} \dot{a}_{1s} &= K_S a_{1s} + K_C u_{a_{1s}} \\ \dot{b}_{1s} &= K_S b_{1s} + K_C u_{b_{1s}} \\ \dot{\theta}_M &= K_S \theta_M + K_C u_{\theta_M} \\ \dot{\theta}_T &= K_S \theta_T + K_C u_{\theta_T} \end{aligned} \quad (2.30)$$

where $K_S = 300$ is the damping coefficient for the cyclics (a_{1s}, b_{1s}) and collectives (θ_M, θ_T), and $K_C = 300$ is the sensitivity of the control inputs $u_{(\cdot)}$.

2.4.2 Longitudinal motion model

First, we will have to transform Eqs. (2.26) in the body frame. Then, the equation used for longitudinal acceleration in the body frame, for APID-MK3, is derived from (2.12) as follows:

$$\dot{u} = \frac{1}{m} (N_w(0, Vw_F) - mg \sin \theta - K_d V_W^2 - K_M \Omega_M^2 \theta_M \sin a_{1s}) \quad (2.31)$$

where $X_W = N_w(0, Vw_R)$ is the wind term (in front direction); $X_G = -mg \sin \theta$ is the gravity term; $X_A = -K_d V_W^2$ is the aerodynamic term, and $X_M = K_M \Omega_M^2 \theta_M \sin a_{1s}$ is the main rotor force term. Now, in the case of a general VTOL, we have

$$\dot{u} = \frac{1}{m} (-mg \sin \theta - \frac{\rho}{2} C_D A C_x V_A^2 - L_M \sin a_{1s} - D_H - D_V) \quad (2.32)$$

also, where $X_G = -mg \sin \theta$; $X_A = -\frac{\rho}{2} C_D A_{Cx} V_A^2$; $X_M = -L_M \sin a_{1s}$; $X_H = -D_H$ is the drag of the horizontal stabilizer, and $X_V = -D_V$ is the drag of the vertical stabilizer. A_{Cx} is the frontal area of the cabin and C_D is its drag coefficient.

Now, one can see the differences between Eq. (2.31) and (2.32)

- the lack of X_W in Eq. (2.32) is not important since it is just a disturbance which can easily be introduced in Eq. (2.32) for the purpose of control design;
- the lack of the terms X_V and X_H in Eq. (2.31) is not significant at all since they represent stabilizers which are not present in APID-MK3 platform model; and
- the difference between the terms Z_A in Eqs. (2.31) and (2.32) is not significant since $V_A|_B \approx V_W|_B$ in hover mode.

2.4.3 Lateral motion model

The equation used for lateral acceleration in the body frame, for APID-MK3, is derived from (2.12) as follows

$$\dot{v} = \frac{1}{m} (N_w(0, Vw_R) + mg \sin \phi \cos \theta + K_d V_W^2 + K_M \Omega_M^2 \theta_M \sin b_{1s}) \quad (2.33)$$

where $Y_W = N_w(0, Vw_R)$ is the wind term (in right direction); $Y_G = mg \sin \phi \cos \theta$ is the gravity term; $Y_A = K_d V_W^2$ is the aerodynamic term, and $Y_M = K_M \Omega_M^2 \theta_M \sin b_{1s}$ is the main rotor force term. Now, in the case of a general VTOL, we have

$$\dot{v} = \frac{1}{m} (mg \sin \phi \cos \theta + \frac{\rho}{2} C_Y A_{Cy} V_W^2 + L_M \sin b_{1s} - L_T + Y_H - L_V) \quad (2.34)$$

also, where $Y_G = mg \cos \phi \cos \theta$; $Y_A = \frac{\rho}{2} C_L A_{Cz} V_A^2$; $Y_M = L_M \sin b_{1s}$; $Y_T = -L_T$ is the tail rotor force term; Y_H is the side force of the horizontal stabilizer, and $Y_V = -L_V$ is the lift force of the vertical stabilizer. A_{Cy} is the side area of the cabin and C_Y is its side force coefficient.

Now, one can see the differences between Eq. (2.33) and (2.34)

- the lack of Y_W in Eq. (2.34) is not important since it is just a disturbance which can easily be introduced in Eq. (2.34) for the purpose of control design;
- the lack of the terms Y_V and Y_H in Eq. (2.33) is not significant at all since they represent stabilizers which are not present in APID-MK3 platform model;

- the difference between the terms Y_A in Eqs. (2.33) and (2.34) is not significant since $V_A \approx V_W$ in hover mode.
- the lack of the tail rotor term Y_T in Eq. (2.33) means that this is an unmodeled dynamics in this equation, due to assumption 9 in Section 2.3.1. Normally this is taken care of by the design of a robust controller.

2.4.4 Vertical motion model

The equation used for vertical acceleration in the body frame, for APID-MK3, is derived from (2.12) as follows:

$$\dot{w} = \frac{1}{m}(N_w(0, Vw_D) + mg \cos \phi \cos \theta - K_d V_W^2 - K_M \Omega_M^2 \theta_M \cos a_{1s} \cos b_{1s}) \quad (2.35)$$

where $Z_W = N_w(0, Vw_D)$ is the wind term (in downward direction); $Z_G = mg \sin \theta$ is the gravity term; $Z_A = -K_d V_W^2$ is the aerodynamic term, and $Z_M = -K_M \Omega_M^2 \theta_M \cos a_{1s} \cos b_{1s}$ is the main rotor force term. Now, in the case of a general VTOL, we have

$$\dot{w} = \frac{1}{m}(mg \cos \phi \cos \theta - \frac{\rho}{2} C_L A_{Cz} V_A^2 - L_M \cos a_{1s} \cos b_{1s} - L_H - Y_V) \quad (2.36)$$

also, where $Z_G = mg \cos \phi \cos \theta$; $Z_A = -\frac{\rho}{2} C_L A_{Cz} V_A^2$; $Z_M = -L_M \cos a_{1s} \cos b_{1s}$; $Z_H = -L_H$ is the lift of the horizontal stabilizer, and $Z_V = -Y_V$ is the side force of the vertical stabilizer. A_{Cz} is the bottom area of the cabin and C_L is its lift coefficient.

Now, one can see the differences between Eq. (2.35) and (2.36)

- the lack of Z_W in Eq. (2.36) is not important since it is just a disturbance which can easily be introduced in Eq. (2.36) for the purpose of control design;
- the lack of the terms Z_V and Z_H in Eq. (2.35) is not significant at all since they represent stabilizers which are not present in APID-MK3 platform model; and
- the difference between the terms Z_A in Eqs. (2.35) and (2.36) is not significant since $V_A|_B \approx V_W|_B$ in hover mode.

In conclusion, we can see that the APID-MK3 model has all the necessary components of a general VTOL model. The only important component that lacks in the APID-MK3 model is the tail rotor force contribution

in lateral motion. This is the element of unmodeled dynamics in the APID-MK3 model which should be taken into account in the flight controller design.

2.4.5 Roll model

The equation used for roll acceleration in the body frame is, for APID-MK3, derived from (2.26) as follows:

$$\dot{p} = \frac{1}{\mathcal{I}_x} (d_p \dot{p} + K_M \Omega_M^2 h_M \theta_M \sin b_{1s}) \quad (2.37)$$

where $R_C = d_p \dot{p}$ is the moment created by the cabin roll due to the action of the main rotor tilt, and $R_M = Y_M h_M = K_M \Omega_M^2 h_M \theta_M \sin b_{1s}$ is the moment created by the main rotor force. Now, in the case of a general VTOL, we have

$$\dot{p} = \frac{1}{\mathcal{I}_x} \left(\begin{aligned} & -\left(\frac{dR}{db_{1s}}\right) b_{1s} - Q_M \sin a_{1s} + L_M h_M \sin b_{1s} \\ & -L_M y_M \cos a_{1s} \cos b_{1s} - L_T h_T + Y_V h_V \end{aligned} \right) \quad (2.38)$$

where $R_{M_s} = -\left(\frac{dR}{db_{1s}}\right) b_{1s}$ is the longitudinal stiffness of the main rotor when tilting; $R_{Q_M} = -Q_M \sin a_{1s}$ is the anti-torque generated by the main rotor rotation; $R_M = Y_M h_M - Z_M y_M = L_M h_M \sin b_{1s} - L_M y_M \cos a_{1s} \cos b_{1s}$ is the moment generated by the main rotor force; $R_T = -L_T h_T$ is the moment generated by the tail rotor force, and $R_V = Y_V h_V$ is the moment due to the vertical stabilizer side force. $\frac{dR}{db_{1s}} = \frac{e n_b m_b R_M (R_M \Omega_M)^2}{4 R_M}$ is the stiffness term, given as a function of the hinge radius e , the main rotor radius R_M , the mass of the blades m_b , the number of blades n_b , and the rotor speed of rotation Ω_M .

Now, one can see the differences between Eq. (2.37) and (2.38)

- the lack of $Z_M y_M$ in Eq. (2.37) can be explained as follows. Due to assumption 7 in Section 2.3.2, we have the position of the tail rotor at the coordinates $H_M = (0, 0, -h_M)$ instead of the position $H_M = (l_M, y_M, -h_T)$, where the latter is valid in the VTOL case. Thus, in the case of APID-MK3, $Z_M y_M = 0$;
- the lack of R_T in Eq. (2.37) can be explained as follows. Due to assumption 7 in Section 2.3.2, we have the position of the tail rotor at the coordinates $H_T = (-l_T, 0, 0)$ instead of the position $H_T = (-l_T, 0, -h_T)$, where the latter is valid in the VTOL case. Thus, in the case of APID-MK3, the tail rotor force does not generate a moment. This is again the case of unmodeled dynamics;
- the lack of R_V in Eq. (2.37) is due to the fact that APID-MK3 does not have a vertical stabilizer;

- the lack of the main rotor stiffness R_{M_s} in Eq. (2.37) is due to assumption 11 in Section 2.3.2 which states that $e = 0$. Thus, $\frac{dR}{db_{1s}} = 0$;
- the lack of the main rotor anti-torque R_{Q_M} in Eq. (2.37) is due to the assumption 8 in Section 2.3.2. This assumption states that $Q_M = 0$ in APID-MK3 platform. However, the term R_C in Eq. (2.37), representing the moment of rotation of the body due the main rotor tilt can be used as a substitute for the stiffness and anti-torque terms.

2.4.6 Pitch model

The equation used for pitch acceleration in the body frame is, for APID-MK3, as follows

$$\dot{q} = \frac{1}{\mathcal{I}_y} (d_q \dot{q} - K_M \Omega_M^2 h_M \theta_M \sin a_{1s}) \quad (2.39)$$

where $M_C = d_q \dot{q}$ is the moment created by the cabin pitch due to the action of the main rotor tilt, and $M_M = -X_M h_M = -K_M \Omega_M^2 h_M \theta_M \sin b_{1s}$ is the main rotor moment. Now, in the case of a general VTOL, we have

$$\begin{aligned} \dot{q} = \mathcal{I}^{-1} (& -\left(\frac{dM}{da_{1s}}\right) a_{1s} + Q_M \sin b_{1s} + L_M l_M \cos a_{1s} \cos b_{1s} \\ & + L_M h_M \sin a_{1s} + Z_H l_H - Q_T + Z_H h_H - X_H h_H - X_V h_V) \end{aligned} \quad (2.40)$$

where $M_{M_s} = \left(\frac{dM}{da_{1s}}\right) a_{1s}$ is the lateral stiffness of the main rotor when tilting; $M_{Q_M} = Q_M \sin b_{1s}$ is the anti-torque generated by the main rotor rotation; $M_M = Z_M l_M - X_M h_M = L_M h_M \sin a_{1s} + L_M l_M \cos a_{1s} \cos b_{1s}$ is the moment generated by the main rotor force; $M_V = -X_V h_V$ is the moment generated by the vertical stabilizer drag, and $M_H = Z_H l_H - X_H h_H$ is the moment generated by the horizontal stabilizer lift and drag. $\frac{dM}{da_{1s}} = \frac{e n_b m_b R_M (R_M \Omega_M)^2}{4 R_M}$ is the stiffness term.

Now, one can see the differences between Eq. (2.39) and (2.40)

- the lack of $Z_M l_M$ in Eq. (2.39) can be explained as follows. Due to assumption 7 in Section 2.3.2, we have the position of the tail rotor at the coordinates $H_M = (0, 0, -h_M)$ instead of the position $H_M = (l_M, y_M, -h_T)$, where the latter is valid in the VTOL case. Thus, in the case of APID-MK3, $Z_M l_M = 0$;
- the lack of the tail rotor anti-torque $-Q_T$ in Eq. (2.39) is due to assumption 9 in Section 2.3.2 which states that $Q_T = 0$. This is yet another case of unmodeled dynamics present in the APID-MK3 model;

- the lack of M_V and M_H in Eq. (2.37) is due to the fact that APID-MK3 does not have a vertical and a horizontal stabilizers.
- the lack of the main rotor stiffness R_{M_s} in Eq. (2.39) is due to assumption 11 in Section 2.3.2 which states that $e = 0$. Thus, $\frac{dM}{da_{1s}} = 0$;
- the lack of the main rotor anti-torque M_{Q_M} in Eq. (2.39) is due to the assumption 8 in Section 2.3.2. This assumption states that $Q_M = 0$ in APID-MK3 platform. However, the term M_C in Eq. (2.39), representing the moment of rotation of the body due the main rotor tilt can be used as a substitute for the stiffness and anti-torque terms.

2.4.7 Yaw model

The equation used for yaw acceleration in the body frame is, for APID-MK3, as follows

$$\dot{r} = \frac{1}{\mathcal{I}_z} (d_r \dot{r} + K_\psi (\theta_T - \psi_T)) \quad (2.41)$$

where $N_C = d_r \dot{r}$ is the moment created by the cabin yaw due to the main rotor rotation, and $N_T = Y_T l_T = K_\psi (\theta_T)$ is moment generated by the tail rotor force, and $N_{Q_T} = -K_\psi \psi_T$ is a compensation for the anti-torque of the main rotor. Now, in the case of a general VTOL, we have

$$\dot{r} = \frac{1}{\mathcal{I}_z} (-Q_M \cos a_{1s} \cos b_{1s} + L_M l_M \sin b_{1s} + L_M y_M \sin a_{1s} + L_T l_T - Y_V l_V) \quad (2.42)$$

where $N_{Q_M} = -Q_M \cos a_{1s} \cos b_{1s}$ is the anti-torque generated by the main rotor rotation; $N_M = Y_M l_M - X_M y_M = L_M l_M \sin b_{1s} + L_M y_M \sin a_{1s}$ is the moment generated by the main rotor force; $N_T = L_T l_T$ is the moment generated by the tail rotor force, and $N_V = -Y_V l_V$ is the moment generated by the vertical stabilizer side force.

Now, one can see the differences between Eq. (2.41) and (2.42)

- the lack of N_M in Eq. (2.41) can be explained as follows. Due to assumption 7 in Section 2.3.2, we have the position of the tail rotor at the coordinates $H_M = (0, 0, -h_M)$ instead of the position $H_M = (l_M, y_M, -h_T)$, where the latter is valid in the VTOL case. Thus, in the case of APID-MK3, $N_M = 0$;
- the lack of N_T in Eq. (2.41) can be explained as follows. Due to assumption 7 in Section 2.3.2, we have the position of the tail rotor at the coordinates $H_T = (-l_T, 0, 0)$ instead of the position

$H_T = (-l_T, 0, -h_T)$, where the latter is valid in the VTOL case. Thus, in the case of APID-MK3, the tail rotor force does not generate a moment. This is again the case of unmodeled dynamics;

- the lack of M_V in Eq. (2.41) is due to the fact that APID-MK3 does not have a vertical stabilizer.
- the lack of the main rotor anti-torque N_{Q_M} in Eq. (2.41) is due to the assumption 8 in Section 2.3.2. This assumption states that $Q_M = 0$ in APID-MK3 platform. However, the term N_{Q_T} in Eq. (2.41) is used as a substitute for the main rotor anti-torque term.

In conclusion, we can see that the terms in the APID-MK3 model are due to simplifications on the main and tail rotors' configurations. These simplifications are reflected in assumptions from Sections 2.3.1 and 2.3.2.

2.5 Comparison with other helicopter models

This section describes the models developed in Carnegie Mellon and Berkeley for their UAV platforms. Both are based on a Yamaha R-50 unmanned helicopter. Section 2.5.1 introduces the Berkeley model of the Yamaha -R50, which is a hover model in the sense of Section 2.2.2. Section 2.5.2 presents the CMU platform, which is similar to the Berkeley one but with two additional equations. These two models are compared to the APID-MK3 model in the body frame.

2.5.1 The Berkeley model

The Berkeley model is of the hover type, (see Section 2.2.2), and is derived under a number of assumptions:

1. velocity and attitude are assumed to be very small so that the simplifications ($\sin x \approx x$, $\cos x \approx 1$) are valid
2. the helicopter body has small velocity and attitude angles in every direction, the Coriolis and gyroscopic terms are ignored ($\Omega \times mV_B = 0$, and $\Omega \times \mathcal{I}\Omega = 0$)
3. only the vertical drag force will be considered as it counteracts the gravity $\vec{\mathcal{F}}_A = (0; 0; Z_F)$
4. the vertical stabilizer does not generate any force on the system in hover $\vec{\mathcal{F}}_V = (0; 0; 0)$
5. only the vertical drag force is generated by the horizontal stabilizer $\vec{\mathcal{F}}_H = (0; 0; Z_H)$

6. the tail rotor shaft is aligned along the y axis and it does not generate any significant forces in other directions it generates only the lateral thrust $\vec{F}_T = (0; -L_T; 0)$, and yaw moment and anti-torque in pitch axis $\vec{M}_T = (0; -Q_T; 0)$
7. the rotor is equated as a rigid disc which can tilt about the longitudinal and lateral axis. The resulting rotor equations of motions are two first order differential equations, for the lateral and longitudinal cyclics. $\vec{F}_M = (-T_M a_{1s}, T_M b_{1s}, -T_M)$

The main rotor is the source of vertical lift, horizontal force, and anti-torque. It also generates the rotor stiffness, that is rolling and pitching moments by flapping. This flapping of the blades described by the rotor equations of motions for the lateral and longitudinal flapping (cyclics) as follows:

$$\begin{aligned}\dot{a}_{1s} &= -\frac{a_{1s}}{\tau_f} - q + A_{b_{1s}} b_{1s} + A_{u_{a_{1s}}} u_{a_{1s}} + A_{u_{b_{1s}}} u_{b_{1s}} \\ \dot{b}_{1s} &= -\frac{b_{1s}}{\tau_f} - p + B_{a_{1s}} a_{1s} + B_{u_{a_{1s}}} u_{a_{1s}} + B_{u_{b_{1s}}} u_{b_{1s}}\end{aligned}\quad (2.43)$$

Then the Berkeley model is given as:

$$\begin{aligned}\dot{u} &= -\frac{1}{m} L_M a_{1s} - g\theta \\ \dot{v} &= \frac{1}{m} (L_M b_{1s} - L_T) + g\phi \\ \dot{w} &= \frac{1}{m} (-L_M + Z_H + Z_F) + g \\ \dot{p} &= \frac{1}{\mathcal{I}_x} (R_{M_s} - Q_M a_{1s} + L_M h_M b_{1s} - L_M y_M - L_T h_T) \\ \dot{q} &= \frac{1}{\mathcal{I}_y} (M_{M_s} + Q_M b_{1s} + L_M h_M a_{1s} + L_M l_M - Q_T + Z_H) \\ \dot{r} &= \frac{1}{\mathcal{I}_z} (-Q_M + L_M l_M b_{1s} + L_T l_T) \\ \dot{a}_{1s} &= -q - \frac{a_{1s}}{\tau_f} + A_{b_{1s}} b_{1s} + A_{u_{a_{1s}}} u_{a_{1s}} + A_{u_{b_{1s}}} u_{b_{1s}} \\ \dot{b}_{1s} &= -p - \frac{b_{1s}}{\tau_f} + B_{a_{1s}} a_{1s} + B_{u_{a_{1s}}} u_{a_{1s}} + B_{u_{b_{1s}}} u_{b_{1s}}\end{aligned}\quad (2.44)$$

We will compare the model above with the one describing APID-MK3.

The longitudinal motion for APID-MK3 (\dot{u}_A) and Berkley (\dot{u}_B) is given as

$$\begin{aligned}\dot{u}_A &= \frac{1}{m} (N_w(0, V_F) - K_d V_W^2 - mg\theta - K_M \Omega_M^2 \theta_M a_{1s}) \\ \dot{u}_B &= \frac{1}{m} (-mg\theta - L_M a_{1s})\end{aligned}\quad (2.45)$$

The APID-MK3 equation of longitudinal motion is “richer” since it has the wind term and cabin drag force. Though the wind in the APID-MK3 case is an external disturbance and can be also introduced as such in the Berkeley model. Also the drag term has been considered in the Berkeley model to be zero because of their assumption 3. Also, the term $K_M \Omega_M^2 \theta_M a_{1s}$ in the APID-MK3 model correspond to the term $L_M a_{1s}$ in the case of Berkeley model and represents the main rotor force.

The lateral motion for APID-MK3 (\dot{v}_A) and Berkley (\dot{v}_B) is given as

$$\begin{aligned}\dot{v}_A &= \frac{1}{m}(N_w(0, V w_R) + K_d V_W^2 + mg\phi + K_M \Omega_M^2 \theta_M b_{1s}) \\ \dot{v}_B &= \frac{1}{m}(mg\theta + L_M b_{1s} + L_T)\end{aligned}\quad (2.46)$$

The APID-MK3 equation of lateral motion is again “richer” in terms of wind action and cabin side force, but it is lacking the term of tail rotor force. The latter is due to assumption 8 in Section (2.3.2). Also, the term $K_M \Omega_M^2 \theta_M b_{1s}$ in the APID-MK3 model correspond to the term $L_M b_{1s}$ in the case of Berkeley model and represents the main rotor force.

The vertical motion for APID-MK3 (\dot{w}_A) and Berkley (\dot{w}_B) is given as

$$\begin{aligned}\dot{w}_A &= \frac{1}{m}(N_w(0, V w_D) - K_d V_W^2 + mg - K_M \Omega_M^2 \theta_M) \\ \dot{w}_B &= \frac{1}{m}(mg - L_M + Z_H + Z_F)\end{aligned}\quad (2.47)$$

The APID-MK3 equation of vertical motion is again “richer” in terms of wind action and cabin. Furthermore, the the term $K_d V_w^2$ in the APID-MK3 model correspond to the term Z_F in the case of Berkeley model and represents the cabin lift force. Also, the term $K_M \Omega_M^2 \theta_M$ in the APID-MK3 model correspond to the term L_M in the case of Berkeley model and represents the main rotor force. The lack of the term Z_H in APID-MK3 model is due to the absence of horizontal stabilizer.

The roll acceleration for APID-MK3 (\dot{p}_A) and Berkley (\dot{p}_B) is given as

$$\begin{aligned}\dot{p}_A &= \frac{1}{\mathcal{I}_x}(d_\phi \dot{\phi} + K_M \Omega_M^2 \theta_M b_{1s} h_M) \\ \dot{p}_B &= \frac{1}{\mathcal{I}_x}(R_{M_s} - Q_M a_{1s} + L_M h_M b_{1s} - L_M y_M - L_T h_T)\end{aligned}\quad (2.48)$$

The rotor stiffness R_{M_s} and anti-torque $Q_M a_{1s}$ due to the main rotor are present in the Berkeley model. These terms are corresponding to the term $d_\phi \dot{\phi}$ in the APID-MK3 model. Furthermore, the the term $-L_T l_T$ is lacking in the APID-MK3 model. This due to assumption 7 ($l_M = 0$) in Section 2.3.2. Also, the term

$K_M h_M \Omega_M^2 \theta_M b_{1s}$ in the APID-MK3 model corresponds to the term $L_M h_m b_{1s}$ in the case of Berkeley model and represents the moment generated by the main rotor force. Because of the assumption 7 in Section 2.3.2, $y_M = 0$. This eliminates the term $L_M y_M$ representing part of the moment due to the main rotor force in the APID-MK3 model of motion, when it exists in the case of Berkeley model.

The pitch acceleration for APID-MK3 (\dot{q}_A) and Berkley (\dot{q}_B) is given as

$$\begin{aligned}\dot{q}_A &= \frac{1}{\mathcal{I}_y} (d_\theta \dot{\theta} - K_M \Omega_M^2 \theta_M a_{1s} h_M) \\ \dot{q}_B &= \frac{1}{\mathcal{I}_y} (M_{M_s} + Q_M b_{1s} + L_M h_M a_{1s} - L_M l_M - Q_T + Z_H l_H)\end{aligned}\quad (2.49)$$

The terms representing the rotor stiffness M_{M_s} and anti-torque $Q_M b_{1s}$ due to the main rotor are present in the Berkeley model. Their action is covered by the term $d_\theta \dot{\theta}$ in the APID-MK3 model. Furthermore, the term $L_M h_M a_{1s}$ in Berkeley model corresponds to the term $K_M \Omega_M^2 \theta_M h_M a_{1s}$ in the APID-MK3 model. The stabilizer term $Z_H l_H$ lacks in the case of APID.-MK3 model because the platform does not have a horizontal stabilizer. Because of the assumption 7 in Section 2.3.2, $l_M = 0$. This eliminates the term $L_M l_M$ representing part of the moment due to the main rotor force in the APID-MK3 model of motion, when it exists in the case of Berkeley model. Finally, the term Q_T in Berkeley model is not represented in the APID-MK3 model due to assumption 9 in Section 2.3.2.

The yaw acceleration for APID-MK3 (\dot{r}_A) and Berkley (\dot{r}_B) is given as

$$\begin{aligned}\dot{r}_A &= \frac{1}{\mathcal{I}_z} (d_\psi \dot{\psi} + K_\psi (\theta_M - \psi_T)) \\ \dot{r}_B &= \frac{1}{\mathcal{I}_z} (-Q_M + L_M l_M b_{1s} + T_T l_T)\end{aligned}\quad (2.50)$$

The term representing the anti-torque Q_M is present in the Berkeley model and its action is covered by the term $d_\psi \dot{\psi}$ in the APID-MK3 model. The tail rotor force creates a moment represented in both models in terms of $T_T l_T$ for Berkeley and $K_\psi \theta_M$ for APID-MK3.

Because of the assumption the assumption 7 in Section 2.3.2, $l_M = 0$. This eliminates the main rotor term $L_M l_M b_{1s}$ in the APID-MK3 model of motion, when it exists in the case of a VTOL model.

The remaining two equations represent the the dynamics related to the lateral b_{1s} and longitudinal a_{1s} cyclics given in Eq. (2.43). We compare here the Berkeley model of the rotor actuation with the one related to the APID-MK3 model given in Eq. (2.30).

For the longitudinal cyclic a_{1s} the equations for APID-MK3 (a_{1sA}) and the Berkeley platform (a_{1sB}) are given as

$$\begin{aligned}\dot{a}_{1sA} &= K_S a_{1s} + K_C u_{a_{1s}} \\ \dot{a}_{1sA} &= -q - \frac{a_{1s}}{\tau_f} + A_{b_{1s}} b_{1s} + A_{u_{a_{1s}}} u_{a_{1s}} + A_{u_{b_{1s}}} u_{b_{1s}}\end{aligned}\quad (2.51)$$

We can see that all the terms in the Berkeley model of rotor dynamics for the longitudinal cyclic a_{1s} are not present in the case of APID-MK3 model: The damping term $K_S a_{1s}$ for APID-MK3 corresponds to $-\frac{a_{1s}}{\tau_f}$ in the Berkeley model. The sensitivity to control inputs $K_C u_{a_{1s}}$ for APID-MK3 corresponds to $A_{u_{a_{1s}}} u_{a_{1s}}$ in the Berkeley model. However a cross-coupling term $A_{b_{1s}} b_{1s}$ in the Berkeley model is not present in the APID-MK3 model. Moreover, the pitch rate q is also present in the the Berkeley model and lacks in the case of APID-MK3. These terms are mainly due to the action of centrifugal forces in the main rotor, which are not taken into account in the case of APID-MK3 model. Similarly, for the lateral cyclic b_{1s} the equations for APID-MK3 (b_{1sA}) and the Berkeley platform (b_{1sB}) are given as

$$\begin{aligned}\dot{b}_{1sA} &= K_S b_{1s} + K_C u_{b_{1s}} \\ \dot{b}_{1sA} &= -p - \frac{b_{1s}}{\tau_f} + B_{a_{1s}} a_{1s} + B_{u_{a_{1s}}} u_{a_{1s}} + B_{u_{b_{1s}}} u_{b_{1s}}\end{aligned}\quad (2.52)$$

Again, we see that all the terms in the Berkeley model of rotor dynamics for the lateral cyclic b_{1s} are not present in the case of APID-MK3 model: The damping term $K_S b_{1s}$ for APID-MK3 corresponds to $-\frac{b_{1s}}{\tau_f}$ in the Berkeley model. The sensitivity to control inputs $K_C u_{b_{1s}}$ for APID-MK3 corresponds to $B_{u_{b_{1s}}} u_{b_{1s}}$ in the Berkeley model. However a cross-coupling term $B_{a_{1s}} a_{1s}$ in the Berkeley model is not present in the APID-MK3 model. Moreover, the roll rate p is also present in the the Berkeley model and lacks in the case of APID-MK3. These terms are mainly due to the action of centrifugal forces in the main rotor, which are not taken into account in the case of APID-MK3 model.

2.5.2 The CMU model

The CMU model is identical to the Berkeley one regarding the equations of translational and rotational motion. Furthermore, the CMU model has also the two equations describing the main rotor flapping in terms of the lateral b_{1s} and longitudinal a_{1s} cyclics. The difference is that the system is augmented with two

additional equations for the Bell-Hiller stabilization. These are given as follows:

$$\begin{aligned}\dot{c} &= -\frac{c}{\tau_s} - q + Cu_{a_{1s}} \\ \dot{d} &= -\frac{d}{\tau_s} - p + Du_{b_{1s}}\end{aligned}\quad (2.53)$$

where c and d are respectively the longitudinal and lateral stabilizer angles. C and D are input derivatives and τ_s is the stabilizer bar's time constant. The dynamics \dot{c} and \dot{d} are not represented in the case of APID-MK3 model.

Furthermore, the yaw dynamics of CMU model can be modeled as a simple first order, for practical purposes, namely to simplify the yaw general equation given in (2.44) as follows:

$$\dot{r} = \frac{1}{\mathcal{I}_z}(-Q_M + L_M l_M b_{1s} + L_T l_T) \quad (2.54)$$

The yaw acceleration simplified equation is given as

$$\dot{r} = N_r r + N_{\theta_T}(\theta_T - r_{fb}) \quad (2.55)$$

where N_r is the yaw damping coefficient; N_{θ_T} is the sensitivity of the pedal control θ_T , and r_{fb} is a yaw feedback term given as a low-pass filter. Its expression is given as follows:

$$\dot{r}_f = K_r r - K_{r_{fb}} r_{fb} \quad (2.56)$$

We can see now a similarity between the equation of yaw dynamics presented for APID-MK3 and the CMU model, given as

$$\dot{r} = \frac{1}{\mathcal{I}_z}(d_r \dot{r} + K_\psi(\theta_T - \psi_T)) \quad (2.57)$$

The term representing the yaw damping $N_r r$ in the CMU model corresponds to the term $\frac{1}{\mathcal{I}_z} d_r \dot{r}$ in the APID-MK3 model; the sensitivity term $N_{\theta_T}(\theta_T - r_{fb})$ in the CMU model corresponds to the term $\frac{1}{\mathcal{I}_z} K_\psi(\theta_T - \psi_T)$. The only difference between them is that, in APID-MK3 model, the term ψ_T is a constant offset, and in the case of CMU model, this term is represented by a low-pass filter expression.

2.6 Summary

This chapter presented in detail the mathematical model of APID-MK3 used for the design of flight controllers. The model is a simplified version of the original APID-MK3 model and is obtained from it under a

number of realistic assumptions. The viability of the simplified model is demonstrated later on in the thesis by using the same controllers on both models with very similar control performance results.

The main contributions of this chapter are as follows:

1. a comparison between a general VTOL model on one side and the simplified and original APID-MK3 models on the other;
2. a comparison between the simplified APID-MK3 model on one side and two different models of an APID-MK3-like unmanned platform, namely Yamaha R50, on the other.

The comparisons show that the APID-MK3 model is realistic enough both from the point of view of a generic VTOL aircraft and a different unmanned platform. The comparisons with the models of Yamaha R-50 show enough similarities to allow the conclusion that the flight controllers developed in Chapter 3 can also be used for the control of this type of unmanned platform as well.

Chapter 3

Flight Controller Design

3.1 Introduction

The subject of this chapter is the design of a flight controller that executes, in a stable and robust manner,

- tracking of trajectories describing curvilinear translational (or horizontal) motion at relatively high speed and
- set-point regulation for fast acceleration/deceleration, hovering and climb.

We do not deal here with the problem of automatically generating the motion trajectories and set-point values – these are assumed given. Furthermore, the robustness of the flight controller is defined as its ability to compensate for:

- external disturbances in terms of wind gusts;
- model parameter uncertainties in terms of changing payload; and
- sensor noise for attitude control signals.

The chapter is organized as follows. Section 3.2 presents the overall cascaded control scheme for the flight controller and defines the control tasks to be executed by its inner- and outer-loop controllers. Section 3.3 introduces fuzzy gain-scheduled (FGS) controller design and Section 3.4 the so-called Mamdani fuzzy controllers. Section 3.5 presents the design of the inner- and outer-loop controllers respectively: the outer-loop uses Mamdani controllers to determine desired attitude angles that can achieve desired velocity at a given altitude. Then the inner-loop takes these desired attitude angles as inputs and generates the actuator

deflections that will result in that attitude. Section 3.6 describes an alternative design where the outer-loop can determine not only desired attitude angles but, also a desired collective main rotor collective pitch. Providing these as inputs to the inner-loop causes the nonlinear attitude dynamics to be transformed into a linear one and thus provides for a linear attitude controller design. However the overall flight controller remains nonlinear because of the nonlinear optimization – gradient descent method (GDM) - used to produce the desired attitude angles and main rotor collective.

3.2 The control scheme

Having in mind that VTOL vehicles of any kind are maneuvered by controlling their attitude angles, i.e., roll, pitch, and yaw, it is only natural to design the flight controller as consisting of two cascaded controllers: a translational controller and an attitude controller. This cascaded structure implies two control loops:

- the inner loop, that has the fastest dynamics, is the attitude controller. It takes desired attitude angles as inputs and generates the actuator commands that will result in the desired attitude;
- the outer-loop controls the slower translational rate variables. It takes desired velocity/position as input and generates desired attitude angles that will produce the desired velocity/position.

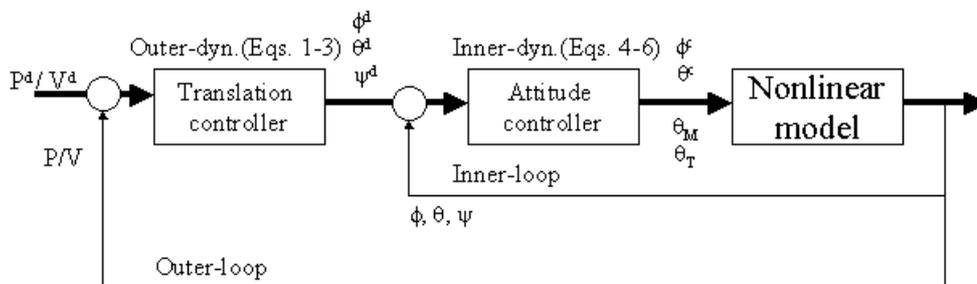


Figure 3.1: Overall flight control scheme

The above scheme for the overall flight controller is presented in Fig. 3.1 and it has to be mentioned here that it is common for all flight controller designs considered in the introductory chapter. However, we differ somewhat from this general control scheme. In the case of the Mamdani plus FGS controller design, the inputs to the inner-loop are not only desired attitude angles but also a desired altitude, see Fig. 3.2. In the

case of the GDM controller design, the inputs to the inner-loop are desired attitude angles and a desired value for the main rotor collective, see Fig. 3.3. This leads to the fact that in the first case we do not have a pure attitude controller but an altitude-and-attitude controller. In the second case though we have a pure attitude controller.

Fig. 3.1 presents the overall flight control scheme, where, V^d or P^d stand for desired velocity or position respectively, that is $V^d = (V_x, V_y, V_z)^d$ or $P^d = (x, y, z)^d$. The outer-loop regulates these reference values and outputs desired attitude angles. The inner-loop control is designed to regulate desired values of the attitude angles $(\phi, \theta, \psi)^d$ and feeds the model with control inputs for the cyclic angles (ϕ_c, θ_c) and collective angles (θ_M, θ_T) to control the main and tail rotors respectively.

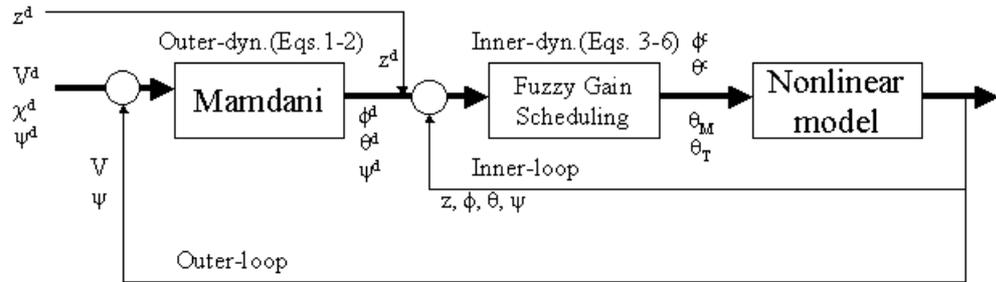


Figure 3.2: Control scheme for the fuzzy gain scheduling method

Fig. 3.2 presents Mamdani plus FGS controllers which inputs are desired horizontal velocity $V^d = (V_x, V_y)^d$ and the desired heading χ^d . The Mamdani controller computes the desired attitude angles $(\phi, \theta, \psi)^d$. These values –plus a desired altitude– are fed to the FGS controller which outputs the values for the control of the model in term of cyclic (ϕ_c, θ_c) and collective angles (θ_M, θ_T) for the main and tail rotors respectively.

One important reason for having altitude-and-attitude controller is as follows. The vertical motion of the helicopter depends on the relation between its weight and the lift force generated by the main rotor blades. If the lift force is greater than the weight, the helicopter accelerates upwards (climb); if it is less than the weight, the helicopter accelerates downwards (descent); and if it is equal to the weight, the helicopter remains at a constant altitude (hover). The horizontal motion of the helicopter (longitudinal – along the x -axis; and lateral – along the y -axis) occurs when there is a horizontal force component. Such a force is generated by inclining the lift force in the desired direction, inducing by that the trust force. However, because of the coupling

between the different types of motion, the following effect is observed: when the lift force is inclined, creating a horizontal motion (trust/drag), the magnitude of the vertical component is decreased under the action of the weight, thus, causing loss of altitude. That is why we would like to control the attitude angles in such a way that a desired horizontal motion is produced but, without loss of altitude. This obviously can be achieved by a controller that is able to simultaneously regulate both the attitude angles and the altitude.

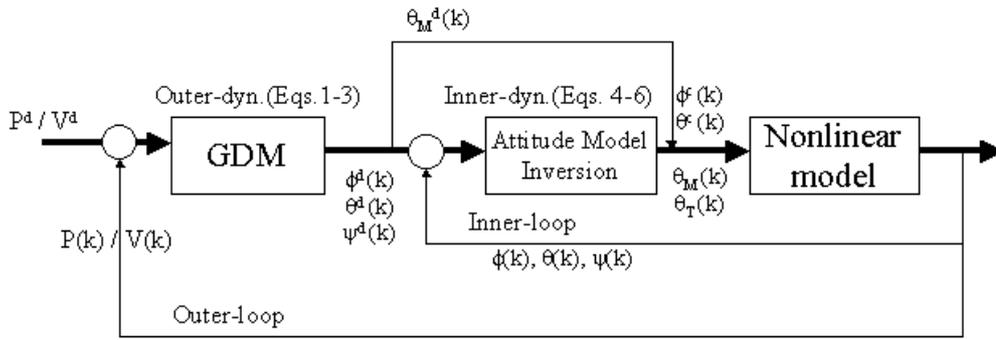


Figure 3.3: Control scheme for the gradient descent method

In Fig. 3.3 the inputs to the model are the control inputs for the cyclics (ϕ_c, θ_c) and the collectives (θ_M, θ_T) for the main and tail rotors respectively. They are computed in the inner-loop by solving the attitude equations in the model using a model-inversion method from desired attitude angles. The outer-loop controller uses a GDM to derive the main collective θ_M and the desired attitude angles $(\phi, \theta, \psi)^d$ given desired velocities V^d or position P^d .

The collective pitch θ_M in the case of the Mamdani plus FGS controller is obtained by providing the FGS controller with a desired altitude z^d , and then a desired collective pitch is computed. That is, θ_M is derived in the inner-loop. In the case of the GDM controller, θ_M is derived in the outer-loop and is then injected in the inner-loop. Another difference between the two controllers is the way the control of the yaw is done. To understand this, one has to keep in mind the following: The air- and body-frames can be assumed the same. The GDM controller is based on this assumption while the Mamdani plus FGS controller is not.

3.3 Fuzzy Gain-Scheduled Control

During the past decade two principally different approaches to the design of fuzzy controllers have emerged: heuristics-based and model-based designs. The latter are known under the name of Takagi-Sugeno (TS) fuzzy models and controllers or fuzzy gain schedulers.

The main applications of the heuristics-based design are in cases when a given plant is subject to well-known and understood manual control strategy by an experienced operator. In this case it is enough to translate the operator's manual control strategy into a set of fuzzy if-then rules in order to obtain an equally good, or even better, wholly automatic fuzzy control system. However, the complexity of the fuzzy control system is normally confined to this of P- or PD-controllers and can rarely be applied to the control of MIMO plants. In our case, the heuristic-based design is applied to the outer-control loop of the flight controller. Since a set of well-known heuristics describing the relationship between horizontal velocities and attitude angles is readily available, these heuristics can be translated into a set of fuzzy if-then rules.

The main applications of the model-based design are in cases when there is an available open-loop plant model, normally a nonlinear one of the MIMO type. For this type of models, of central interest are the issues of stability and robustness of the closed loop system as well as the ease with which one can automate the controller design. The attitude-and-altitude dynamics of a helicopter are a typical example of such a nonlinear MIMO plant. However, there is no general method for designing nonlinear controllers. What is available today, is a collection of alternative and complementary techniques, each best applicable to a particular class of nonlinear systems. This explains why the helicopter's original nonlinear model has to be "modified" in one way or another in order a particular design technique to be used. In this context, the advantage of using Takagi-Sugeno models is that a large class of nonlinear plants, including the attitude dynamics of a helicopter, can be well represented by these models, without the need to modify the original nonlinear dynamics in any significant way.

Our goal in this chapter is to integrate the above two types of fuzzy controller designs in an overall flight controller, by taking advantage of easily available heuristics for its outer-loop and a MIMO nonlinear model for its inner-loop. In the following section, we present the Takagi-Sugeno (TS) fuzzy model in some detail. Although the TS fuzzy models are based on fuzzy logic we will not go deeper into the fuzzy logic theories. Interested readers are directed to [21] for a comprehensive reading. We will simply use the TS fuzzy model as a tool to approximate/represent nonlinear functions and systems. Section 3.3.1 presents the general structure

of the TS fuzzy model. The TS model used in the context of dynamical systems is outlined in Section 3.3.2. The problem of how to obtain a TS model as a representation or approximation of nonlinear functions is given in Section 3.3.3. Section 3.3.4 presents TS fuzzy controllers. In Section 3.4 the Mamdani type of controllers are discussed. In Section 3.5 the design of the fuzzy flight controller is presented. Finally Section 3.6 introduces the GDM flight controller.

3.3.1 Structure of the Takagi-Sugeno model

The TS fuzzy model, originally proposed in [1], is composed of a fuzzy if-then rule base that partitions a space – usually called the *universe of discourse* – into fuzzy regions described by the *rule antecedents*. The *consequent* of each rule i is usually a functional expression $y_i = f_i(x)$. A common format of a rule i is as follows:

$$\text{Rule } i : \text{IF } \theta_1 \text{ is } F_1^i \text{ AND } \theta_2 \text{ is } F_2^i \text{ AND } \dots \text{ AND } \theta_q \text{ is } F_q^i \text{ THEN } y_i = f_i(x).$$

The vector $\theta \in R^q$ contains the *premise variables* and may be a subset of the independent variables $x \in R^n$. Each premise variable θ_j has its own universe of discourse that is partitioned into fuzzy regions by the fuzzy sets describing the *linguistic variable* F_{jk} . The premise variable θ_j belongs to a fuzzy set k with a *truth value* given by the *membership function* $\mu_{jk}(\theta_j) : R \rightarrow [0, 1]$ for $k = 1, 2, \dots, s_j$ where s_j is the number of fuzzy sets for premise variable j . The notation F_j^i and μ_j^i refers to the specific linguistic variable and its membership function respectively that correspond to the premise variable θ_j in rule i . That is, $F_j^i \in \{F_{j1}, F_{j2}, \dots, F_{js_j}\}$ and $\mu_j^i(\theta_j) \in \{\mu_{j1}(\theta_j), \mu_{j2}(\theta_j), \dots, \mu_{js_j}(\theta_j)\}$.

The truth value (or activation degree) h_i for the complete rule i is computed using the aggregation operator AND, also called a *t-norm*, which is often denoted by $\otimes : [0, 1] \times [0, 1] \rightarrow [0, 1]$,

$$h_i(\theta) = \mu_1^i(\theta_1) \otimes \mu_2^i(\theta_2) \otimes \dots \otimes \mu_q^i(\theta_q). \quad (3.1)$$

There is a number of different t-norms available, see [21]. However, in this work we will use the simple algebraic product, thus Eq. (3.1) reads

$$h_i(\theta) = \prod_{j=1}^q \mu_j^i(\theta_j). \quad (3.2)$$

The degree of activation for rule i is then normalized as

$$w_i(\theta) = \frac{h_i(\theta)}{\sum_{r=1}^l h_r(\theta)}, \quad (3.3)$$

where l is the number of rules. This normalization implies that

$$\sum_{i=1}^l w_i(\theta) = 1. \quad (3.4)$$

In conclusion, the response of the TS model, for a given x and θ , is a weighted sum of the consequent functions, f_i , which reads

$$y = \sum_{i=1}^l w_i(\theta) f_i(x). \quad (3.5)$$

Note that y in Eq. (3.5) is a convex combination of the local functions (models) f_i , a fact that facilitates stability analysis.

3.3.2 TS models for dynamical systems

This section presents the TS model described in Section 3.3.1 in the context of dynamical systems. A general continuous dynamical system may be given as

$$\begin{aligned} \dot{x} &= f(x, u, \vartheta) \\ y &= g(x, u, \vartheta), \end{aligned} \quad (3.6)$$

where $f : R^n \times R^m \times R^s \rightarrow R^n$ and $g : R^n \times R^m \times R^s \rightarrow R^p$. $\vartheta \in R^s$ is a vector of possibly time varying parameters. The functions f and g may very well be represented by a TS fuzzy system. Letting θ be a subset of x , u and ϑ we can write rule i in a fuzzy rule base as

Rule i : IF θ_1 is F_1^i AND θ_2 is F_2^i AND ... AND θ_q is F_q^i

$$\text{THEN } \begin{cases} \dot{x} = \hat{f}_i(x, u, \vartheta) \\ y = \hat{g}_i(x, u, \vartheta) \end{cases},$$

where $\hat{f}_i : R^n \times R^m \times R^s \rightarrow R^n$ and $\hat{g}_i : R^n \times R^m \times R^s \rightarrow R^p$. The TS fuzzy system is then written

$$\begin{aligned} \dot{x} &= \sum_{i=1}^l w_i(\theta) \hat{f}_i(x, u, \vartheta) \\ y &= \sum_{i=1}^l w_i(\theta) \hat{g}_i(x, u, \vartheta), \end{aligned} \quad (3.7)$$

and the weights $w_i(\theta)$ are computed as described in Eq. (3.3). The choice of the consequent functions \hat{f}_i and \hat{g}_i depends on the application. We will confine ourselves to the following case:

$$\begin{aligned} \dot{x} &= \sum_{i=1}^l w_i(\theta) (A_i x + B_i u) = A(\theta) x + B(\theta) u \\ y &= \sum_{i=1}^l w_i(\theta) C_i x = C(\theta) x, \end{aligned} \quad (3.8)$$

or the more general form, derived from the Taylor expansion

$$\begin{aligned}\dot{x} &= \sum_{i=1}^l w_i(\theta)(A_i x + B_i u + a_i) = A(\theta)x + B(\theta)u + a(\theta) \\ y &= \sum_{i=1}^l w_i(\theta)(C_i x + c_i) = C(\theta)x + c(\theta),\end{aligned}\tag{3.9}$$

where $A_i \in R^{n \times n}$, $B_i \in R^{n \times m}$, $a_i \in R^n$, $C_i \in R^{p \times n}$ and $c_i \in R^p$. If it is not clear from the context, we will in the following label (3.8) as a *homogeneous* TS fuzzy system while we denote (3.9) as an *affine* TS fuzzy system.

3.3.3 Obtaining TS fuzzy models

There are basically two principal ways to obtain a TS fuzzy model: nonlinear identification from experimental data or using linearization of some kind. A good exposure of different methods for TS fuzzy model identification is given in the collection [22]. A comprehensive description of a method for automatic identification of TS models for control purposes is also presented in [23]. The purpose of this section is to discuss how a TS fuzzy model of the form (3.8) or (3.9) can be obtained from the general nonlinear system description in Eq.(3.6). Thus, a nonlinear system is already given and we will not deal with identification, and thereby redirecting interested readers to the references mentioned above. It has been shown that a TS model of type (3.8) can approximate any smooth nonlinear function and its first order derivative [24]. Furthermore, in [25] it is shown that an affine TS system may also be able to approximate the second order derivatives of a smooth nonlinear function.

Although the universal approximation capabilities are of a considerable theoretical interest we will only confine ourselves with the mechanism of how to approximate a nonlinear function by a TS model in this section. Two approaches are presented: the first one is based on approximation, using Taylor expansion of the nonlinear function in a number of points. The second one tries to find a linearizing transformation such that the nonlinear function is exactly represented by the fuzzy system over a specified domain. The two techniques have one thing in common: they strive to capture the nonlinearity of the original nonlinear system into the rule base.

Approximation by Taylor expansion

The idea of this method is to do a first-order Taylor expansion in different points θ_i and let the rule base describe the validity of the obtained linear model in each point. First, one has to decide which variables

in x , u and ϑ , i.e., the premise variables θ , capture the nonlinearities in Eq. (3.6). A reasonable number of linearization points together with a partition of the universe of discourse must also be chosen. The consequent system parameters are then obtained by

$$A_i = \left. \frac{\partial f}{\partial x}(x, u, \vartheta) \right|_{x_i, u_i, \vartheta_i}, \quad B_i = \left. \frac{\partial f}{\partial u}(x, u, \vartheta) \right|_{x_i, u_i, \vartheta_i}, \quad C_i = \left. \frac{\partial g}{\partial x}(x, u, \vartheta) \right|_{x_i, u_i, \vartheta_i} \quad (3.10)$$

with the affine terms

$$a_i = f(x_i, u_i, \vartheta_i) - A_i x_i - B_i u_i \quad (3.11)$$

and

$$c_i = g(x_i, u_i, \vartheta_i) - C_i x_i. \quad (3.12)$$

Obviously, a TS model obtained by this method is an extension of a linear system obtained through linearization in an equilibrium point. As a parallel, in classical gain scheduling one does often approximate a nonlinear system with a number of linear systems obtained through linearization over the equilibrium manifold ε of the nonlinear system in Eq. (3.6), as

$$\varepsilon = \{(x, u, \vartheta) \mid f(x, u, \vartheta) = 0\}. \quad (3.13)$$

See [26] for a discussion about linearization and classical gain scheduling. When linearizing outside the equilibrium manifold we obtain an affine term as in (3.9), see also the discussion in [27] and [28]. This implies that our approximation may resemble the original nonlinear system even outside the equilibrium manifold. We may also say that the fuzzy system approximates the flow of the vector field represented by Eq. (3.6) as discussed in [27]. The example below shows how to approximate a simple function using linearization, in terms of first-order Taylor expansion.

Example 1 Consider the approximation of the function $f(x) = \sin(x)$ over the domain $[-\pi, \pi]$. Fig. 3.4 show the original function together with approximations using three $(-2.6, 0$ and $2.6)$ respectively five $(-2.6, -1.57, 0, 1.57, 2.6)$ linearization points. Fig. 3.5 shows the chosen membership functions.

From Example (1) we can immediately see that more linearization points – i.e., an increased granularity in the fuzzy partition – the better approximation accuracy.

Approximation by a linearizing transformation

Here we describe how a TS fuzzy model (without affine terms) may be represented by using sector bounded nonlinearities. This approach is thoroughly described in [29] and [30] and only an outline is given here. This technique can at least be used for continuously differentiable systems with $f(0, 0) = 0$. The idea is to bound

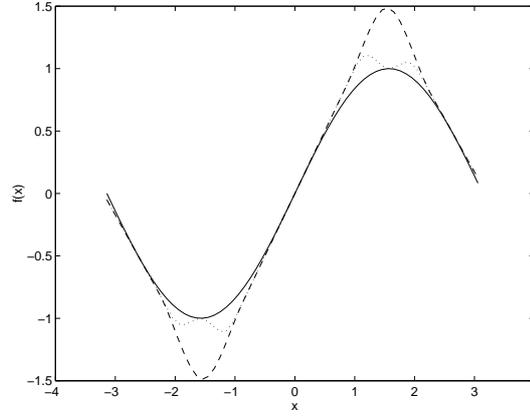


Figure 3.4: Real function (solid), 3-mf approximation (dashed), 5-mf approximation (dotted)

nonlinearities by sectors described by linear subsystems. The resulting system is then written as a convex combination of these subsystems by transforming the original nonlinearities into fuzzy membership functions.

Example 2 Consider the same function as in Example 1, $f(x) = \sin(x)$. We would like to express this function as a convex combination of linear functionals as follows

$$\mu_1(x)a_1x + (1 - \mu_1(x))a_2x = \sin(x) \quad (3.14)$$

with $\mu_1(x) \geq 0$ and $\mu_1(x) + (1 - \mu_1(x)) = 1$. Solving for $\mu_1(x)$ gives

$$\mu_1(x) = \begin{cases} \frac{\sin(x) - a_2x}{a_1x - a_2x} & x \neq 0 \\ 0 & x = 0 \end{cases} \quad (3.15)$$

If we let $x \in [x_{min}, x_{max}] = [-\pi, \pi]$ we set $a_1 = \min_x \sin(x) = -1$ and $a_2 = \max_x \sin(x) = 1$. Hence Eq. (3.15) becomes

$$\mu_1(x) = \begin{cases} \frac{x - \sin(x)}{2x} & x \neq 0 \\ 0 & x = 0 \end{cases} \quad (3.16)$$

Inserting Eq. (3.16) in to Eq. (3.14) we obtain equality for all $x \in [x_{min}, x_{max}]$. Hence, it is possible to exactly represent $\sin(x)$ over an interval using a convex combination with the weights $w_1(x) = \mu_1(x)$ and $w_2(x) = 1 - \mu_1(x)$.

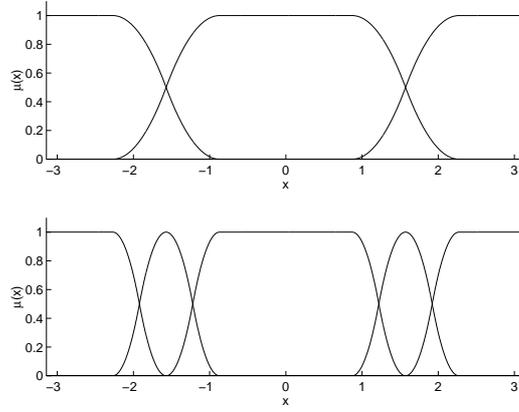


Figure 3.5: Membership functions to approximate $\sin(x)$

The reasoning used in this example is easily extended to a wide variety of nonlinear functions and in particular to nonlinear systems of the form

$$\begin{aligned}
 f(x, u) = & \begin{bmatrix} a_{11}(x, u) & a_{12}(x, u) & \dots & a_{1n}(x, u) \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}(x, u) & a_{n2}(x, u) & \dots & a_{nn}(x, u) \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \\
 & + \begin{bmatrix} b_{11}(x, u) & b_{12}(x, u) & \dots & b_{1m}(x, u) \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1}(x, u) & b_{n2}(x, u) & \dots & b_{nm}(x, u) \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}.
 \end{aligned} \tag{3.17}$$

Each nonlinearity is taken care of separately. The nonlinearities are assumed to be bounded as in Example (2): $a_{ij} \in [\min_{x,u} a_{ij}(x, u), \max_{x,u} a_{ij}(x, u)]$ for x, u belonging to the universe of discourse. The result from Example (2) shows that this technique is likely to be more powerful than the Taylor expansion approach described in the previous section: both with respect to the number of rules required and with the respect to the absence of an affine term. However, one problem that may arise -in the context of control design for the obtained TS system- is that the linear subsystems may be uncontrollable and/or unobservable respectively. The reason is that the relationships between states and inputs may be hidden in the membership functions and will not show up in the linear subsystems. Another issue is that the resulting sub-models, in general, do not represent local system dynamics, but only sector boundaries. Sometimes it may be desirable to have the intuitive feeling for the local system behavior encoded explicitly in the fuzzy model.

3.3.4 Takagi-Sugeno controllers

We will concentrate here on output feedback controllers that do not depend on an explicit observer scheme, and we will constrain the model with certain assumptions when necessary. It is also assumed that the varying parameters in θ are measurable.

$$\begin{aligned}\dot{x} &= \sum_{i=1}^l w_i(\theta)(A_i x + B_i u + a_i) = A(\theta)x + B(\theta)u + a(\theta) \\ y &= \sum_{i=1}^l w_i(\theta)(C_i x + c_i) = C(\theta)x + c(\theta)\end{aligned}\tag{3.18}$$

Static output feedback fuzzy control

Consider the problem of how to construct a static output feedback stabilizing fuzzy controller for a fuzzy system with linear consequents

$$\begin{aligned}\dot{x} &= \sum_{i=1}^l w_i(\theta)(A_i x + B_i u) = A(\theta)x + B(\theta)u \\ y &= \sum_{i=1}^l w_i(\theta)C_i x = C(\theta)x.\end{aligned}\tag{3.19}$$

A fuzzy controller may be formed as

$$u = \sum_{i=1}^l w_i(\theta)K_i y = K(\theta)y.\tag{3.20}$$

The closed loop system then becomes

$$\begin{aligned}\dot{x} &= A(\theta)x + B(\theta)K(\theta)y = (A(\theta) - B(\theta)K(\theta)C(\theta))x \\ &= \sum_{i=1}^l \sum_{j=1}^l \sum_{k=1}^l w_i(\theta)w_j(\theta)w_k(\theta)(A_i + B_i K_j C_k)x.\end{aligned}\tag{3.21}$$

To simplify the presentation, assume for a moment that $B_i = B$ and $C_i = C$ for $i = 1, 2, \dots, l$. Then Eq. (3.21) simplifies to

$$\dot{x} = \sum_{i=1}^l w_i(\theta)(A_i + BK_i C)x.\tag{3.22}$$

Now using a quadratic Lyapunov function, $V(x) = x^T P x$, the following stability theorem for Eq. (3.22) can be verified

Theorem 1 [10] *If there exists a symmetric positive definite matrix P such that*

$$(A_i + BK_i C)^T P + P(A_i + BK_i C) < 0\tag{3.23}$$

then the system in Eq. (3.22) is asymptotically stable.

One attempt to formulate the design is to form the feasibility problem

$$\begin{aligned} \text{Find } P, K_i \quad & \text{s.t. } (A_i + BK_iC)^T P + P(A_i + BK_iC) < 0 \\ & i = 1, 2, \dots, l. \end{aligned} \quad (3.24)$$

The problem (3.24) is indeed very difficult to solve, because it is not convex in P and K_i . Furthermore, there does not exist a linearizing variable substitution in this case. Heuristic approaches have been proposed for solving (3.24), for example, the so-called $V - K$ iteration [31]. However, using these kind of methods one can only hope to achieve convergence to a local minimum.

The discussion above shows that static output feedback fuzzy controllers may be hard to find. However, in the next section it is shown how it may be possible to obtain full order dynamic output feedback controllers by solving a convex programming problem.

Dynamic output feedback fuzzy controllers

In this section it is shown how a dynamic output feedback fuzzy controller can be designed.

Consider the system

$$\begin{aligned} \dot{x} &= \sum_{i=1}^l w_i(\theta)(A_i x + B_i u + a_i) = A(\theta)x + B(\theta)u + a(\theta) \\ y &= \sum_{i=1}^l w_i(\theta)(C_i x + c_i) = C(\theta)x + c(\theta) \end{aligned} \quad (3.25)$$

System (3.25) can be thought of as a polytopic linear parameter varying system subjected to certain disturbances, stemming from the affine terms. The idea here is to make use of the framework for gain scheduled \mathcal{H}_∞ controllers in order to: 1) Shape the closed loop transient dynamics so that it conforms to the performance specifications and 2) Design the controller to reject the influence of the affine terms. We will concentrate on the first step for the moment. In [10] it is shown how to design a gain scheduled controller with guaranteed \mathcal{H}_∞ performance γ for the following general LPV system

$$\begin{aligned} \dot{x} &= A(\theta)x + B_1(\theta)w + B_2(\theta)u \\ z &= C_1(\theta)x + D_{11}(\theta)w + D_{12}(\theta)u \\ y &= C_2(\theta)x + D_{21}(\theta)w + D_{22}(\theta)u \end{aligned} \quad (3.26)$$

where $A \in R^{n \times n}$, $B_1 \in R^{n \times m_1}$, $B_2 \in R^{n \times m_2}$, $C_1 \in R^{p_1 \times n}$ and $C_2 \in R^{p_2 \times n}$. θ is allowed to vary in a parameter box Θ with l extreme points and the LPV matrices depend affinely on θ . The system (3.26)

is therefore constrained to vary in a polytope with vertices given by the extreme points in Θ . Thus, setting $a(\theta) = 0$ and $c(\theta) = 0$ in (3.18) yields to a system that can be written in the form of (3.26). In addition, the following assumptions must hold:

Assumption 1 $D_{22}(\theta) = 0$ or $D_{22i} = 0$ for $i = 1, \dots, l$.

Assumption 2 $B_2(\theta)$, $C_2(\theta)$, $D_{12}(\theta)$ and $D_{21}(\theta)$ are parameter independent or $B_{2i} = B_2$, $C_{2i} = C_2$, $D_{12i} = D_{12}$ and $D_{21i} = D_{21}$ for $i = 1, \dots, l$.

Assumptions (1) and (2) may seem to be restrictive in practice. However, it is often possible to augment the plant with linear filters representing the actuator and sensor dynamics and thereby make the input and output matrices parameter independent. The objective is to find an internally stabilizing parameter-dependent dynamic output feedback controller, with the infinity norm of the transfer function from w to z less than γ , $\|T_{zw}\|_\infty \leq \gamma$, of the form

$$\begin{aligned} \dot{x}_c &= A_K(\theta)x_c + B_K(\theta)y \\ u &= C_K(\theta)x_c + D_K(\theta)y \end{aligned} \quad (3.27)$$

with the controller parameters

$$\begin{aligned} \Omega(\theta) &:= \begin{bmatrix} A_K(\theta) & B_K(\theta) \\ C_K(\theta) & D_K(\theta) \end{bmatrix} \\ &\in \text{Co} \left\{ \begin{bmatrix} A_{Ki} & B_{Ki} \\ C_{Ki} & D_{Ki} \end{bmatrix}, i = 1, \dots, l \right\}. \end{aligned} \quad (3.28)$$

From the convex solvability condition theorem [10], there exists a LPV controller that guarantees quadratic \mathcal{H}_∞ performance γ over Θ if and only if there exist symmetric matrices $R, S \in \mathcal{R}^{n \times n}$ satisfying the $2r + 1$ linear matrix inequalities

$$\tilde{\mathcal{N}}_R^T \left[\begin{array}{cc|c} A_i R + R A_i^T & R C_{1i}^T & B_{1i} \\ C_{1i} R & -\gamma I & D_{11i} \\ \hline B_{1i}^T & D_{11i}^T & -\gamma I \end{array} \right] \tilde{\mathcal{N}}_R < 0 \quad i = 1, \dots, l \quad (3.29)$$

$$\tilde{\mathcal{N}}_S^T \left[\begin{array}{cc|c} A_i^T S + S A_i & S B_{1i} & C_{1i}^T \\ B_{1i}^T S & -\gamma I & D_{11i}^T \\ \hline C_{1i} & D_{11i} & -\gamma I \end{array} \right] \tilde{\mathcal{N}}_S < 0 \quad i = 1, \dots, l \quad (3.30)$$

$$\begin{bmatrix} R & I \\ I & S \end{bmatrix} \geq 0 \quad (3.31)$$

with

$$\tilde{\mathcal{N}}_R = \left[\begin{array}{c|c} \mathcal{N}_R & 0 \\ \hline 0 & I \end{array} \right] \quad \text{and} \quad \tilde{\mathcal{N}}_S = \left[\begin{array}{c|c} \mathcal{N}_S & 0 \\ \hline 0 & I \end{array} \right],$$

where \mathcal{N}_R and \mathcal{N}_S denote the null space of (B_2^T, D_{12}^T) and (C_2, D_{21}) respectively. If a feasible solution is found, a closed loop Lyapunov matrix X_{cl} can be obtained by computing (via singular value decomposition) two matrices M and N such that

$$MN^T = I - RS \quad (3.32)$$

and solve the following matrix equation for X_{cl}

$$\begin{bmatrix} S & I \\ N^T & 0 \end{bmatrix} = X_{cl} \begin{bmatrix} I & R \\ 0 & M^T \end{bmatrix} \quad (3.33)$$

Now, given a closed loop Lyapunov matrix X_{cl} the vertex controllers

$$\Omega_i = \begin{bmatrix} A_{Ki} & B_{Ki} \\ C_{Ki} & D_{Ki} \end{bmatrix} \quad (3.34)$$

can be found (from the use of the bounded real lemma, see e.g. [32], extended to polytopic systems) by solving the following system of LMIs

$$\begin{bmatrix} A_{cli}^T X_{cl} + X_{cl} A_{cli} & X_{cl} B_{cli} & C_{cli}^T \\ B_{cli}^T X_{cl} & -\gamma I & D_{cli}^T \\ C_{cli} & D_{cli} & -\gamma I \end{bmatrix} < 0 \quad (3.35)$$

$i = 1, \dots, l$

with

$$\begin{aligned} A_{cli} &= \begin{bmatrix} A_i + B_2 D_{Ki} C_2 & B_2 C_{Ki} \\ B_{Ki} C_2 & A_{Ki} \end{bmatrix} \\ B_{cli} &= \begin{bmatrix} B_{1i} + B_2 D_{Ki} D_{21} \\ B_{Ki} D_{21} \end{bmatrix} \\ C_{cli} &= \begin{bmatrix} C_{1i} + D_{12} D_{Ki} C_2 & D_{12} C_{Ki} \end{bmatrix} \\ D_{cli} &= \begin{bmatrix} D_{11i} + D_{12} D_{Ki} D_{12} \end{bmatrix} \end{aligned}$$

Thus, a LPV controller can be designed for the linear part of (3.18). The controller is then parameterized on-line by using measurements of θ and the convex decomposition given by the fuzzy rule base, i.e., the convex combination:

$$\begin{bmatrix} A_K(\theta) & B_K(\theta) \\ C_K(\theta) & D_K(\theta) \end{bmatrix} = \sum_{i=1}^l w_i(\theta) \begin{bmatrix} A_{Ki} & B_{Ki} \\ C_{Ki} & D_{Ki} \end{bmatrix} \quad (3.36)$$

Due to assumptions (1) and (2), the affine term $c(\theta)$ disappears and attention must only be given to $a(\theta)$. The approach taken here is to consider $a(\theta)$ as a “measurable” disturbance. That is, given a particular θ from measurements, it is always possible to compute the disturbance $a(\theta)$ acting on the model. By feed-forward of the computed $a(\theta)$ to the controller it is possible to make the controller compensate for it, see Fig. 3.6. Modeling the disturbance and its measurement is easily done by adding entries in B_1 and D_{21} respectively. The controller is then synthesized according to the steps outlined above.

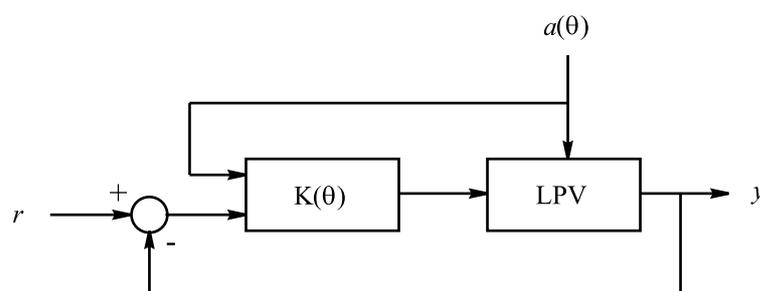


Figure 3.6: Closed loop with an affine term as a “measurable” disturbance

3.4 Mamdani-type controllers

A Mamdani fuzzy model/controller is a static MIMO system. It is formed by the following components: a fuzzy rule base, a fuzzy inference engine, a fuzzifier operator, and a defuzzifier operator. In what follows we will briefly describe each of these components and for the sake of simplicity we will consider a Mamdani SISO model/controller. The results easily can be extended to the MIMO case and for a detailed description the reader is referred to [33].

3.4.1 The fuzzy rule base

The fuzzy rule base consists of a set of n fuzzy *if-then* rules each one of the form:

$$\text{if } X \text{ is } A^{(k)} \text{ then } Y \text{ is } B^{(k)}, \quad k = 1, \dots, n,$$

where

- $A^{(k)}$ is the linguistic value of the input variable X in the k th rule. For example, X can represent the error between desired longitudinal velocity and the actual one and $A^{(k)}$ may express “small negative error”. “ x is $A^{(k)}$ ” is interpreted by a given membership function $\int_{\mathcal{X}} \mu_{A^{(k)}}(x)/x$; and
- $B^{(k)}$ is the linguistic value of the output Y in the i th rule. For example, Y can represent the desired pitch that is required to correct the above error and $B^{(k)}$ may express “small positive pitch”. “ y is $B^{(k)}$ ” is interpreted by a given membership function $\int_{\mathcal{Y}} \mu_{B^{(k)}}(y)/y$

The “meaning” of a *single rule* is represented as a fuzzy relation defined as:

$$\forall i : \tilde{R}_i^{(k)} = \int_{\mathcal{X} \times \mathcal{Y}} \min(\mu_{A^{(k)}}(x), \mu_{B^{(k)}}(y)) / (x, y).$$

Then the meaning of the *whole set of rules* is defined as:

$$\tilde{R}_m = \bigcup_{k=1}^n \tilde{R}_m^{(k)},$$

which means that

$$\begin{aligned} \forall x, y : \mu_{\tilde{R}_m}(x, y) &= \max_k \mu_{\tilde{R}_m^{(k)}}(x, y) \\ &= \max_k \min(\mu_{A^{(k)}}(x), \mu_{B^{(k)}}(y)). \end{aligned}$$

The fuzzifier operator

Given a crisp input x^* , i.e., a specific value of the input variable X , this crisp input is *fuzzified* in the following manner:

$$\forall e : \mu^*(x) = \begin{cases} 1 & \text{if } x = x^* \\ 0 & \text{otherwise} \end{cases}$$

The fuzzy inference engine

After the crisp input x^* is fuzzified the inference with the set of rules is performed via the use of the operation *composition* between the fuzzified crisp input and \tilde{R}_m as:

$$\tilde{Y}_{out} = \mu^* \circ \tilde{R}_m, \quad \text{i.e.,} \quad \forall y : \mu_Y(y) = \max_k \min(\mu_{A^{(k)}}(x^*), \mu_{B^{(k)}}(y))$$

The fuzzy set \tilde{Y}_{out} is the output that corresponds to the given fuzzy input x^* and is denoted as $\tilde{Y}_{out} = \int_{\mathcal{Y}} \mu_{\tilde{Y}_{out}}^*(y)/y$

The defuzzifier operator

The defuzzifier operator is applied on the fuzzy output given by the membership function \tilde{Y}_{out} , for example by finding its center-of-gravity y^* . That is, the center of the area covered by the membership function $\tilde{Y} =$

$$y^* = \frac{\int_{\mathcal{Y}} y \mu_{B^{(k)}}^*(y) dy}{\int_{\mathcal{Y}} \mu_{B^{(k)}}^*(y) dy}$$

3.4.2 The Mamdani PD-controller

A Mamdani model can be instantiated as a Mamdani PD-controller in the following manner. Let e denote the error, that is the difference between the desired and the actual system state at sample time k , $e(k) = x_d - x(k)$. Let also Δe denote the change-of-error, that is $\Delta e(k) = e(k) - e(k-1)$, and u be the control output that is to be produced given particular values for $e(k)$ and $\Delta e(k)$. Then a Mamdani PD-controller consists of a set of fuzzy *i-f-then* rules of the form:

$$\text{if } e \text{ is } E^{(k)} \text{ and } \Delta e \text{ is } \Delta E^{(k)} \text{ then } u \text{ is } U^{(k)}, \quad k = 1, \dots, n,$$

where

- $E^{(k)}$ is the linguistic value of the error e in the k th rule. For example, $A^{(k)}$ may express “small negative error”. “ x is $E^{(k)}$ ” is interpreted by a given membership function $\int_{\mathcal{E}} \mu_{E^{(k)}}(e)/e$; and
- $U^{(k)}$ is the linguistic value of the control input u in the i th rule. For example, $U^{(k)}$ may express “small positive pitch”. “ u is $U^{(k)}$ ” is interpreted by a given membership function $\int_{\mathcal{U}} \mu_{U^{(k)}}(u)/u$

The linguistic values taken by e and Δe are normally classified as “positive”, “negative”, and “zero” and represented as appropriate fuzzy sets on the universes of discourse of these two variables. For example, “positive small error” expresses the fact that the state x is below its desired value x_d , and the difference between x and x_d is rather small. Furthermore, a “a positive big change-of-error” means that the state x at time k is significantly smaller than x_d at time $k-1$.

When e and Δe are considered together this expresses the “dynamics” of the system. For example, the combination (“positive e ” and “negative Δe ”) means that the current process state is below its desired value, and increasing since the change-of-error is negative. Thus, the current state shows a tendency of approaching the desired value from below.

The linguistic values for the control input u are similarly divided in “positive”, “negative”, and “zero”. For example, a “small positive u ” means that the current u should be changed to a rather small value. Such a course of action only depends on the current values of e and Δe . For example, consider the case (“positive big e ” and “positive big Δe ”). This means that the current state is at a large distance below its desired value and it has settled at this position after having made a large step in the direction of the desired value. Thus, since the state is moving in the direction of the desired value with a large step, the appropriate control input would be “small positive u ” so that the state is moved further up in the direction of the desired value without overshooting it.

In this thesis we utilize the structure of the above Mamdani PD-controller in the outer control loop of the flight controller. In our case the control input is a desired value for either pitch, roll, or yaw which has to be maintained in order to achieve a desired velocity.

3.5 The Fuzzy Flight Controller

The two-step synthesis for the fuzzy flight controller proceeds as follows:

1. given desired horizontal velocity at certain altitude, a set of Mamdani-type of linguistic rules computes desired attitude angles that help achieve this desired velocity at the given altitude. The rules are heuristic in nature and reflect the experience of a human “pilot” who is an expert in remotely controlling the vehicle;
2. on the basis of TS model for the dynamics of both vertical motion and attitude angles, TS control laws (FGS controller) that achieve the desired attitude angles at a given altitude are designed.

The current control system for APID-MK3 does not utilize the full range of the rotor attitude angles. As a consequence, this produces lower rate-of-change of the attitude angles ϕ , θ and ψ , and consequently the control is done on rather small ranges for these – all this reduces manoeuvrability w.r.t. these angles. In this context, the objective of our study is to design an attitude controller which acts on much larger ranges of the attitude angles, i.e., $-\pi/4 \leq \phi \leq +\pi/4$, $-\pi/4 \leq \theta \leq +\pi/4$, $-\pi \leq \psi \leq +\pi$, by utilizing the full range of the rotor attitude angles. The latter, for the purpose of this study, are in the interval *approx.* $[-0.25, 0.25]$ rad.

3.5.1 The FGS controller for the inner-loop

The design approach used here consists of the following steps:

- Decoupling the nonlinearities in the control inputs by adding first-order actuator transfer functions – as a result, the nonlinearities are moved into the state;
- The new model is linearized either using Taylor series expansion around appropriately chosen points in the state space, or by bounding the nonlinearities in the state by linear functions – in this way the nonlinear model is approximated by a TS fuzzy model, which boils down to convex combination of linear sub-models;
- A gain scheduled output feedback \mathcal{H}_∞ controller for the so-obtained approximated model is designed.

In what follows we will describe in more detail the above three steps of the design. The mathematical model used for the attitude/altitude control of APID MK-III, defined in the inertial frame, is of the form:

$$\begin{aligned}
\ddot{z} &= \frac{1}{m}(Z_w + Z_s - K_M \Omega_M^2 \theta_M \cos \phi \cos \theta) \\
\ddot{\phi} &= -a\dot{\phi} + dK_M \Omega_M^2 (b_{1s} + N_\phi) \theta_M \\
\ddot{\theta} &= -b\dot{\theta} - eK_M \Omega_M^2 (a_{1s} + N_\theta) \theta_M \\
\ddot{\psi} &= -c\dot{\psi} + f((\theta_T + N_\psi) + \psi_T)
\end{aligned} \tag{3.37}$$

where the state vector is $(z, \phi, \theta, \psi, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi})$, i.e., altitude, attitude angles and their respective rates. The control inputs are $(b_{1s}, a_{1s}, \theta_M, \theta_T)$, i.e., these are the usual control inputs in terms of lateral and longitudinal cyclics, and collective angles for the main and tail rotors. The first equation describes the dynamics of altitude motion where Z_w is a wind force in the z-axis, and Z_s is the gravity force on the cabin. $(N_\phi, N_\theta, N_\psi)^T$ represents the noise associated with the attitude angles. $a, b, c, d, e, f, K_M, \Omega_M$, and ψ_T are model parameters, and their specific values have been introduced in Chapter 2 together with the full model description.

The above model has to be transformed in the form of Eq. (3.18), with the vector $a(\theta)$ being the affine term representing wind accelerations and attitude angles noise $a(\theta) = (\frac{Z_w}{m}, N_\phi, N_\theta, N_\psi)^T$. All the outputs of the model are directly measurable (attitude angles and their rates, position and velocity). Thus the expression in 3.18 is reduced to the identity matrix and $c(\theta) = 0$.

In the above model, the control inputs are produced by servo-actuators and that is why we will introduce the transfer functions that relate the outputs from these servo-actuators to the control inputs. The servo-actuators used in APID-MK3 are first-order transfer functions with saturation, augmented with a linear model for the Bell-Hiller mixer and angles-to-signals plus signals-to-angles transformations, as shown in Fig. 3.7.

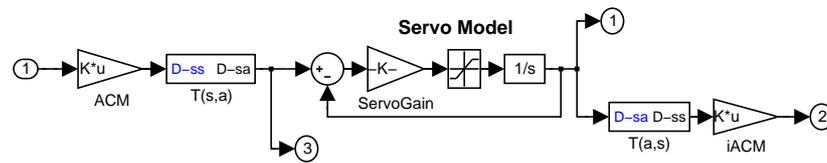


Figure 3.7: Servo-actuator diagram including the Bell-Hiller mixer

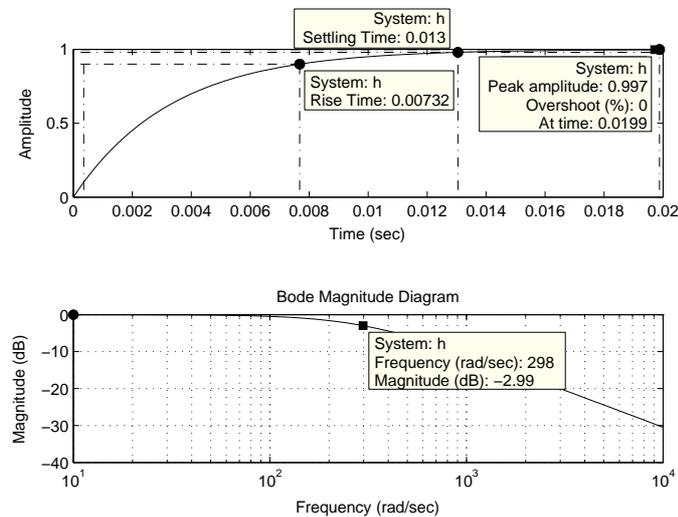


Figure 3.8: Bode diagram and step response for the servo-actuator

The servo-actuator presented in Fig. 3.7 is simplified in the form of a first-order function which still obeys the constraints for the actuated signals to be within the range $[-1.8, +1.8]$ for (b_{1s}, a_{1s}) and $[-1, +1]$ for (θ_M, θ_T) . The Bode diagram for the simplified servo-actuator is given in Fig. 3.8. We verify that the outputs produced from the original (see Fig. 3.9 middle) and simplified servo-actuators (see Fig. 3.9 bottom)—once proportionally amplified—are very similar and are still within the range $[-1, +1]$. This in turn implies that the rotor angles produced by the servo-actuators' outputs are realistic, that is they are within their admissible ranges (approx. $[-0.25, +0.25]$ rad).

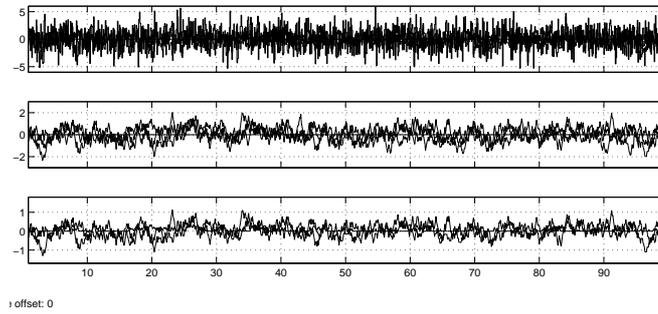


Figure 3.9: Boundaries of inputs, and outputs for the servo-actuators

The simplified expression for the servo-actuators transfer functions are as follows

$$\begin{aligned}
 \dot{b}_{1s} &= -300 b_{1s} + 300 u_{b_{1s}}, \\
 \dot{a}_{1s} &= -300 a_{1s} + 300 u_{a_{1s}}, \\
 \dot{\theta}_M &= -300 \theta_M + 300 u_{\theta_M}, \\
 \dot{\theta}_T &= -300 \theta_T + 300 u_{\theta_T},
 \end{aligned} \tag{3.38}$$

Now, we can expand the model with the above actuators whose outputs are the cyclic angles (b_{1s}, a_{1s}) and the collective angles (θ_M, θ_T) , and whose inputs are the signals $(u_{b_{1s}}, u_{a_{1s}}, u_{\theta_M}, u_{\theta_T})$. As a result, this will shift the nonlinearities –due to couplings between the control inputs– into nonlinearities between the state variables, as mentioned in Section 3.3.4.

As a result, the model from Eq. (3.37) becomes as follows:

$$\begin{aligned}
\dot{x}_3 &= x_9 \\
\dot{x}_4 &= x_{10} \\
\dot{x}_5 &= x_{11} \\
\dot{x}_6 &= x_{12} \\
\dot{x}_9 &= \frac{1}{m}(Z_w + Z_s - K_M \Omega_M^2 x_{15} (\cos x_{10} \cos x_{11})) \\
\dot{x}_{10} &= -a x_{10} + d K_M \Omega_M^2 x_{15} (x_{13} + N_\phi) \\
\dot{x}_{11} &= -b x_{11} - e K_M \Omega_M^2 x_{15} (x_{14} + N_\theta) \\
\dot{x}_{12} &= -c x_{12} + f ((x_{16} + N_\psi) + \psi_T), \\
\dot{x}_{13} &= -300 x_{13} + 300 u_{b_{1s}}, \\
\dot{x}_{14} &= -300 x_{14} + 300 u_{a_{1s}}, \\
\dot{x}_{15} &= -300 x_{15} + 300 u_{\theta_M}, \\
\dot{x}_{16} &= -300 x_{16} + 300 u_{\theta_T},
\end{aligned} \tag{3.39}$$

where the $(x_3, \dots, x_6, x_9, \dots, x_{12})$ corresponds to $(z, \phi, \theta, \psi, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi})$, i.e., altitude, attitude angles and their respective rates. (x_{13}, \dots, x_{16}) are $(b_{1s}, a_{1s}, \theta_M, \theta_T)$, i.e., these are the usual control inputs in terms of lateral and longitudinal cyclics, and collective angles for the main and tail rotors. Furthermore, $u_{b_{1s}}$, $u_{a_{1s}}$, u_{θ_M} , and u_{θ_T} are the commanded cyclic roll and pitch together with the main and tail rotor collective angles. Note that b_{1s} , a_{1s} , θ_M and θ_T are now pseudo state variables.

3.5.2 Linearization of the inner-loop model

Consider again the model described in Eq. (3.39). For each of the nonlinear terms in this model we choose a linear bounding such that the fuzzy system obtained represents exactly the nonlinear system. Now, we consider $\cos(x_{10}) \cos(x_{11}) x_{15}$, $x_{13} x_{15}$ and $x_{14} x_{15}$ to be the nonlinear terms subject to linear bounding – these reside in the attitude equations associated with \dot{x}_9 , \dot{x}_{10} and \dot{x}_{11} respectively. The state variables involved in these nonlinear terms satisfy:

$$\begin{aligned}
x_{10}, x_{11} &\in [-\pi/4, \pi/4], \\
x_{15} &\in [\pi/18, 5\pi/18].
\end{aligned} \tag{3.40}$$

The state variable x_{15} is trivially bounded by

$$.1745 < x_{15} < .8727. \quad (3.41)$$

$\cos(x_{10})$ and $\cos(x_{11})$, taking into account the bounds from Eq. (3.40), can be bounded by the two constant functions:

$$0.7071 < \cos(x_{10}) < 1, \quad 0.7071 < \cos(x_{11}) < 1. \quad (3.42)$$

The above bounds result in

$$0.5 < \cos(x_{10}) \cos(x_{11}) < 1. \quad (3.43)$$

Then the above three nonlinear terms can be represented via the use of the derived upper and lower bounds –as described in Example (2)– by Eq. (3.14), and in the following manner:

$$\begin{aligned} x_{13}x_{15} &= F_1^1 0.8727x_{13} + F_1^2 0.1745x_{13}, \\ x_{14}x_{15} &= F_1^1 0.8727x_{14} + F_1^2 0.1745x_{14}, \\ \cos(x_{10}) \cos(x_{11})x_{15} &= F_2^1 x_{15} + F_2^2 0.5x_{15}, \end{aligned}$$

where $F_1^1, F_2^1 \in [0, 1]$, $F_1^2 = 1 - F_1^1$ and $F_2^2 = 1 - F_2^1$. By solving the above equations for F_1^1, F_1^2, F_2^1 and F_2^2 , see Example (2) Eq. (3.15), we obtain the following membership functions:

$$\begin{aligned} F_1^1(x_{15}) &= (x_{15} - 0.1745)/0.6981, \\ F_1^2(x_{15}) &= (0.8727 - x_{15})/0.6981, \\ F_2^1(x_{10}, x_{11}) &= 2 \cos(x_{10}) \cos(x_{11}) - 1, \\ F_2^2(x_{10}, x_{11}) &= 2 - 2 \cos(x_{10}) \cos(x_{11}). \end{aligned}$$

The graphs of the membership functions F_1^1 and F_1^2 related to the roll and pitch angles are shown in Fig. 3.10 left part, and the graphs of F_2^1 and F_2^2 related to the collective pitch are shown in Fig. 3.10 right part.

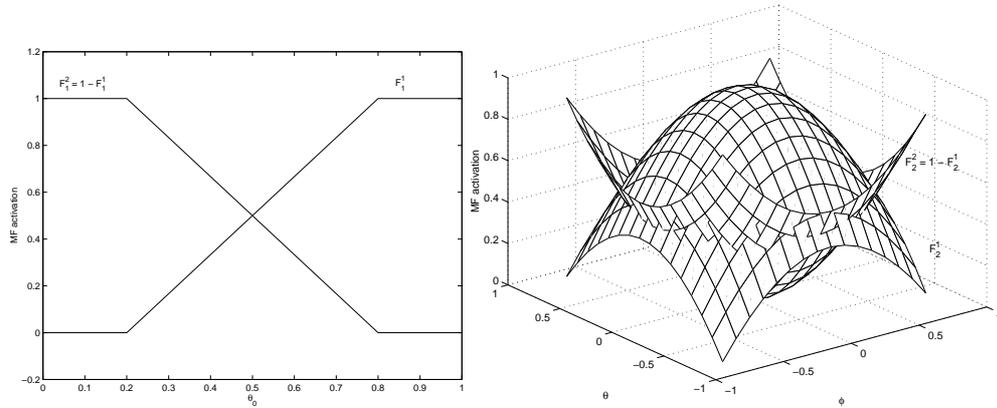


Figure 3.10: Membership functions $F_1^{(\cdot)}$ and $F_2^{(\cdot)}$

Takagi-Sugeno Controller for the inner-loop

The fuzzy model, with regard to the results established in Section 3.7, is then expressed as the following set of only four rules:

- 1 : IF x_{15} is F_1^1 and $\cos(x_{10}) \cos(x_{11})$ is F_2^1 THEN $\dot{x} = A_1 x + Bu + a(\theta)$,
- 2 : IF x_{15} is F_1^1 and $\cos(x_{10}) \cos(x_{11})$ is F_2^2 THEN $\dot{x} = A_2 x + Bu + a(\theta)$,
- 3 : IF x_{15} is F_1^2 and $\cos(x_{10}) \cos(x_{11})$ is F_2^1 THEN $\dot{x} = A_3 x + Bu + a(\theta)$,
- 4 : IF x_{15} is F_1^2 and $\cos(x_{10}) \cos(x_{11})$ is F_2^2 THEN $\dot{x} = A_4 x + Bu + a(\theta)$.

In the above rules the matrix A_1 is obtained from of Eq. (3.39) in the following manner. First, values of x_{10} , x_{11} , and x_{15} are chosen such that $F_1^1(x_{15}) = 1$, and $F_2^1(x_{10}, x_{11}) = 1$, namely, these are $x_{10} = x_{11} = -\pi/4$ and $x_{15} = 5\pi/18$. Second, we replace the previous values in Eq. (3.39) where A_1 is given by the equations associated with $(\dot{x}_3, \dots, \dot{x}_{12})$. The matrix B is represented by the equations associated with $(\dot{x}_{13}, \dots, \dot{x}_{16})$ and is thus the same for all the rules.

For illustration, we give the expression of the state-space representation for the first rule. If $x_{15} = 5\pi/18$

and $\cos x_{10} \cos x_{11} = 0.5$ then

$$A_1 = \begin{bmatrix} 0_4 & 1_4 & 0_4 \\ 0_4 & G_4 & H_4 \\ 0_4 & 0_4 & S_4 \end{bmatrix}; \quad B = \begin{bmatrix} 0_4 \\ 0_4 \\ -S_4 \end{bmatrix}; \quad a(\theta) = \begin{bmatrix} 0_4 \\ N \\ 0_4 \end{bmatrix}$$

$$C = \begin{bmatrix} 1_4 & 0_4 & 0_4 \\ 0_4 & 1_4 & 0_4 \end{bmatrix}; \text{ and } D = \begin{bmatrix} 0_4 \\ 0_4 \end{bmatrix}$$

where “ 0_i ” is a zero matrix of rank i , “ 1_i ” identity matrix of rank i , and G_4, H_4 and S_4 given by:

$$G_4 = \begin{bmatrix} 0 & 0 & -0.5K_M\Omega_M^2 & 0 \\ 0.8727dK_M\Omega_M^2 & 0 & 0 & 0 \\ 0 & -0.8727eK_M\Omega_M^2 & 0 & 0 \\ 0 & 0 & 0 & f \end{bmatrix};$$

$$H_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & -a & 0 & 0 \\ 0 & 0 & -b & 0 \\ 0 & 0 & 0 & -c \end{bmatrix}; \quad N = \begin{bmatrix} Z_w + Z_s \\ N_\phi \\ N_\theta \\ N\psi + f\psi_T \end{bmatrix} \quad \text{and} \quad S_4 = 300 \cdot 1_4$$

The rest of $A_2, A_3,$ and A_4 are obtained in the same manner. The global model resulting from the fuzzy rules corresponds to the one from the system described by Eq. (3.25), where the entries of C related to altitude/attitude angles and their rates are equal to 1 and furthermore, C is identical for all rules ($C_i = C$). Also, $D_i = D = 0$. Thus the global TS model corresponding to 3.39 is given as:

$$\begin{aligned} \dot{x} &= \sum_{i=1}^4 w_i(x_{10}, x_{11}, x_{15})(A_i x + Bu) + a(\theta) \\ y &= \sum_{i=1}^4 w_i(x_{10}, x_{11}, x_{15})(C_i x) = Cx \end{aligned} \quad (3.44)$$

In the above, w_i is the degree to which a rule is activated given some values for x_{10}, x_{11} and x_{15} . Then, according to Eqs. (3.1–3.4) in Section 3.3.1, we have that

$$\begin{aligned} w_1 &= F_1^1(x_{15}) \cdot F_2^1(x_{10}, x_{11}) & w_2 &= F_1^1(x_{15}) \cdot F_2^2(x_{10}, x_{11}) \\ w_3 &= F_1^2(x_{15}) \cdot F_2^1(x_{10}, x_{11}) & w_4 &= F_1^2(x_{15}) \cdot F_2^2(x_{10}, x_{11}) \\ \text{and} \quad \sum_{i=1}^4 w_i &= 1 \end{aligned} \quad (3.45)$$

Given the TS fuzzy model in Eq. (3.44), a FGS dynamic output feedback \mathcal{H}_∞ controller can be designed

- The available equations describing the dynamics of horizontal motion do not take into account aerodynamic effects related to the main rotor.
- Also the contributions of the tail rotor torque and force are neglected.

Thus, using the Eq. (3.47) to derive desired attitude angles, given desired horizontal velocities, is not be a reliable option. Instead, the Mamdani-type of linguistic controller uses the magnitude of acceleration and velocity-error to infer attitude angles that if achieved will reduce the velocity error to zero. Thus they “mimic” a human pilot’s behavior when trying to achieve certain desired velocities via manual control.

Mamdani rule for longitudinal velocity

In this context, the rules used to compute desired values for pitch are according to Section 3.4 of the form:

***IF** e_{v_x} is Neg and \dot{e}_{v_x} is Neg **THEN** θ^d is Pos,*

where e_{v_x} is the longitudinal velocity-error and \dot{e}_{v_x} is the longitudinal acceleration and θ^d the desired pitch angle. The “heuristic” interpretation of this particular rule is as follows: if the current longitudinal velocity is higher than the desired one and we are accelerating, i.e., we are moving further away from the desired velocity which is caused by a negative pitch angle. In order to bring the current velocity back to the desired one we have to slow down the longitudinal motion and reverse the acceleration. This is done by bringing the pitch from a negative to a positive angle. Furthermore, Neg and Pos are linguistic labels for the magnitudes of e_{v_x} , \dot{e}_{v_x} , and the pitch. The meaning of these linguistic labels is given by fuzzy sets defined on the physical domains of e_{v_x} , \dot{e}_{v_x} , and the pitch. Fig. 3.12 illustrates the above rule in terms of these membership functions. All in all there are 9 rules describing the relationship between e_{v_x} , \dot{e}_{v_x} and the pitch.

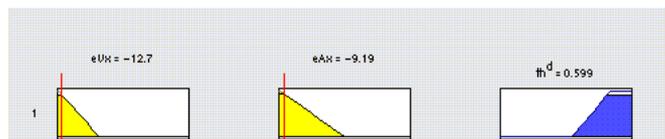


Figure 3.12: Rule for longitudinal speed with membership functions

Mamdani rule for lateral velocity

The rules used to compute desired values for roll are of the form:

IF e_{v_y} is Neg and \dot{e}_{v_y} is Neg **THEN** ϕ^d is Neg,

where e_{v_y} is the lateral velocity-error and \dot{e}_{v_y} is the lateral acceleration and ϕ^d the desired roll angle. The “heuristic” interpretation of this particular rule is as follows: if the current lateral velocity is higher than the desired one and we are accelerating, i.e., we are moving further away from the desired velocity which is caused by a positive roll angle. In order to bring the current velocity back to the desired one we have to slow down the lateral motion and reverse the acceleration. This is done by bringing the roll from a positive to a negative angle. Furthermore, Neg and Pos are linguistic labels for the magnitudes of e_{v_y} , \dot{e}_{v_y} , and the roll. Fig. 3.13 illustrates the above rule in terms of membership functions corresponding to these linguistic labels. All in all there are 9 rules describing the relationship between e_{v_y} , \dot{e}_{v_y} and the roll.

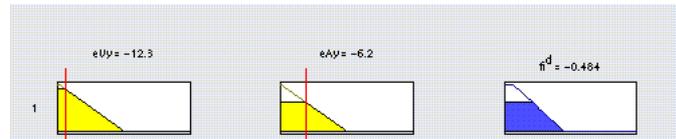


Figure 3.13: Rule for lateral speed with membership functions

Mamdani rule related to the heading

The desired value for the yaw is computed by rules as:

IF e_χ is Pos and \dot{e}_χ is Neg **THEN** ψ^d is Zero.

where e_χ is the heading-error and \dot{e}_χ is its rate of change and ψ^d the desired yaw. The “heuristic” interpretation of this particular rule is as follows: if the current heading is higher than the desired one and we are reducing it, i.e., we are moving closer to the desired heading which is caused by certain orientation of the horizontal velocity. In this case we maintain the current yaw. Furthermore, Neg, Pos, and Zero are linguistic labels for the magnitudes of e_χ , \dot{e}_χ , and the current yaw. Fig. 3.14 illustrates the above rule in terms

of membership functions corresponding to these linguistic labels. All in all there are 9 rules describing the relationship between e_x , \dot{e}_x and the yaw.

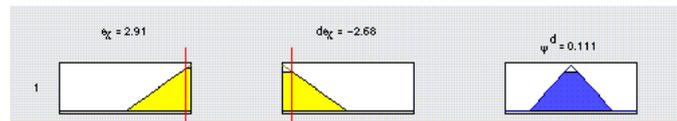


Figure 3.14: Rule for heading with membership functions

The first two types of rules neglect the cross-couplings between pitch and roll angles in the dynamics of longitudinal and lateral motions. However, these couplings are taken care by the heading rules that in addition also prevent side-slip by restricting the yaw to be always equal to the heading. Furthermore, the pitch and roll angles affect the dynamics of vertical motion so that they cause a drop in altitude. Preventing this is taken care of at the level of the TS controller. The control scheme computing desired attitude angles given desired horizontal velocities at a given altitude is presented in Fig. 3.15.

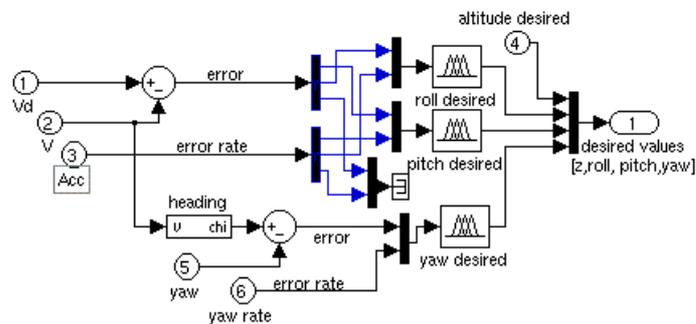


Figure 3.15: The Mamdani controller

3.5.4 Related work

The work by Sugeno [20] reports a hierarchical, Mamdani-type of a controller for the unmanned helicopter Yamaha R-50 by Yamaha Motors. The lower layer contains a number of Mamdani-type control modules: longitudinal (pitch control), lateral (roll control), collective (vertical control), rudder (yaw control), and coupling compensation modules. Furthermore, within each such module there is a number of sub-modules only some of which correspond directly to our Mamdani-type controller from Section 3.4. These are as follows:

- Longitudinal: this module includes a \dot{x} Mamdani-type controller. The \dot{x} controller infers a desired pitch angle using a velocity-error and its derivative and is identical to the one used by us. However, in our case the desired pitch angle is sent to the TS controller from Section 3.3 while in Sugeno's case it is sent to yet another controller from the same module;
- Lateral: this module includes a \dot{y} Mamdani-type controller. The \dot{y} controller infers a desired roll angle using a velocity-error and its derivative and is identical to the one used by us. However, in our case the desired roll angle is sent to the FGS controller while in Sugeno's case it is sent to yet another controller from the same module;
- Collective: this module includes a \dot{z} Mamdani-type controller. The \dot{z} controller infers a control value for the main collective using altitude, velocity-error and its derivative. In our case, the control value for the main collective is obtained by the FGS controller given a desired altitude;
- Rudder: this module, given a desired heading, infers a control input for the tail collective using yaw angle error and its rate of change. In our case we infer a desired yaw angle given a desired heading, yaw angle error and its rate of change. The desired yaw angle is sent to the FGS controller which in turn gives a control input for the tail collective.
- Coupling compensation: the use of this module is twofold: i) it takes into account cross-couplings between longitudinal/lateral and vertical motion; ii) it takes into account cross-couplings between yaw and roll during a turn. In our case the first type of cross-couplings are taken care of by the FGS controller. The second type are partially solved by guaranteeing that a turn is always performed in a directional manner.

3.6 The Gradient-Descent Flight Controller

The design consists of two steps:

1. a gradient descent optimization method is used to compute for each desired horizontal velocity/position –at desired altitude– the corresponding desired values for the attitude angles and the main rotor collective pitch.
2. a linear control scheme is used to regulate the attitude angles so that the helicopter achieves its desired horizontal velocity/position at the desired altitude.

3.6.1 The open loop model

The mathematical model, used for the control of APID-MK3 and defined in the inertial frame, is of the form:

$$\begin{aligned}
\ddot{x} &= \frac{1}{m}(X_w - K_M \Omega_M^2 \theta_M (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)) \\
\ddot{y} &= \frac{1}{m}(Y_w - K_M \Omega_M^2 \theta_M (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)) \\
\ddot{z} &= \frac{1}{m}(Z_w + Z_s - K_M \Omega_M^2 \theta_M \cos \phi \cos \theta) \\
\ddot{\phi} &= -a\dot{\phi} + dK_M \Omega_M^2 b_{1s} \theta_M \\
\ddot{\theta} &= -b\dot{\theta} - eK_M \Omega_M^2 a_{1s} \theta_M \\
\ddot{\psi} &= -c\dot{\psi} + f(\theta_T + \psi_T)
\end{aligned} \tag{3.48}$$

where the state vector is $(x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi})$, i.e., horizontal position, altitude, attitude angles and their rates. The control inputs are $(b_{1s}, a_{1s}, \theta_M, \theta_T)$, i.e., these are the usual control inputs in terms of lateral and longitudinal cyclics and collective angles for the main and tail rotors.

3.6.2 The inner loop attitude controller

The equations below describe the dynamic behavior of the attitude angles (ϕ, θ, ψ) :

$$\begin{aligned}
\ddot{\phi} &= -a\dot{\phi} + dK_M \Omega_M^2 \theta_M b_{1s}, \\
\ddot{\theta} &= -b\dot{\theta} - eK_M \Omega_M^2 \theta_M a_{1s}, \\
\ddot{\psi} &= -c\dot{\psi} + f(\theta_T - \psi_T),
\end{aligned} \tag{3.49}$$

where θ_M is derived from the translational equations and $(b_{1s}, a_{1s}, \theta_T)$ acts as a control inputs.

The control problem for the outer-loop is to calculate angles (ϕ, θ, ψ) and θ_M so that given desired velocities (\dot{x}_d, \dot{y}_d) and altitude z_d are reached in a certain amount of time. This requires specific angles $(\phi_d, \theta_d, \psi_d)$ and a corresponding collective θ_M . The control problem for the inner-loop is to regulate the angles $(\phi_d, \theta_d, \psi_d)$ calculated by the outer-loop. Given a desired collective pitch– determined by the outer-loop, the control in the inner-loop is trivially decoupled into three separate channels. The yaw channel is linear with respect to the tail collective θ_T with an off-set term ψ_T compensating for the anti-torque induced by the main rotor around its shaft.

The control law design for the inner-loop is straightforward: we require the inner-loop to follow a behavior

that is determined by three decoupled second order linear equations. These are given as follows:

$$\begin{aligned}
\ddot{\phi} &= k_{\dot{\phi}}\dot{\phi} + k_{\phi}(\phi - \phi_d) \\
\ddot{\theta} &= k_{\dot{\theta}}\dot{\theta} + k_{\theta}(\theta - \theta_d) \\
\ddot{\psi} &= k_{\dot{\psi}}\dot{\psi} + k_{\psi}(\psi - \psi_d)
\end{aligned} \tag{3.50}$$

where the proportional gains (k_{ϕ} , k_{θ} , and k_{ψ}), and derivative gains ($k_{\dot{\phi}}$, $k_{\dot{\theta}}$, and $k_{\dot{\psi}}$) have to fulfill a desired performance. These gains are designed in a classical way by a robust pole placement or other appropriate methods. Comparing the right-hand sides of Eqs.(3.49) and (3.50) the corresponding control inputs can then be calculated as:

$$\begin{aligned}
b_{1s} &= \frac{1}{dK_M\Omega_M^2\theta_M}(\dot{\phi}(k_{\dot{\phi}} + a) + k_{\phi}(\phi - \phi_d)) \\
a_{1s} &= \frac{-1}{eK_M\Omega_M^2\theta_M}(\dot{\theta}(k_{\dot{\theta}} + b) + k_{\theta}(\theta - \theta_d)) \\
\theta_T &= \frac{1}{f}(\dot{\psi}(k_{\dot{\psi}} + c) + k_{\psi}(\psi - \psi_d)) + \psi_T
\end{aligned} \tag{3.51}$$

3.6.3 The outer loop velocity controller

The design of the outer-loop controller proceeds as follows: given desired horizontal velocities/position for a desired altitude, a gradient descent optimization method, applied to the equations of translational motion computes desired values for attitude angles and the main rotor collective pitch. The equations at the outer-loop describe the dynamic behavior of the system in the inertial frame:

$$\begin{aligned}
\ddot{x} &= \frac{1}{m}(X_w - K_M\Omega_M^2\theta_M(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)), \\
\ddot{y} &= \frac{1}{m}(Y_w - K_M\Omega_M^2\theta_M(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)), \\
\ddot{z} &= \frac{1}{m}(Z_w + Z_s - K_M\Omega_M^2\theta_M\cos\phi\cos\theta),
\end{aligned} \tag{3.52}$$

In contrast to the inner-loop, the outer-loop control problem cannot be solved in a straightforward manner because of: (i) the nonlinearities in the right-hand sides of Eq. (3.52); and (ii) the fact that the system is over-determined which means that there are more control inputs than outputs. Similar to the design of the inner-loop controller, we introduce desired dynamics in the right-hand side of the translation equations as

follows

$$\begin{aligned}
\ddot{x} &= k_{\dot{x}}(\dot{x} - \dot{x}_d), \\
\ddot{y} &= k_{\dot{y}}(\dot{y} - \dot{y}_d), \\
\ddot{z} &= k_z \dot{z} + k_z(z - z_d),
\end{aligned} \tag{3.53}$$

where $k_{\dot{x}}$, $k_{\dot{y}}$, k_z , and k_z are gains designed according to a required performance in terms of time responses. In order to meet this performance, the right-hand sides of Eqs. (3.52) and (3.53) have to be forced to become equal by adjusting ϕ , θ , ψ , and θ_M . For this purpose, we use an optimization approach. Let both Eqs. (3.52) and (3.53) be written in a compact way

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \tilde{f}(\phi, \theta, \psi, \theta_M), \tag{3.54}$$

where the components of the vector function \tilde{f} are given by the right-hand sides of Eq. (3.52) and

$$\tilde{u}(x, y, z) = \begin{pmatrix} k_{\dot{x}}(\dot{x} - \dot{x}_d) \\ k_{\dot{y}}(\dot{y} - \dot{y}_d) \\ k_z \dot{z} + k_z(z - z_d) \end{pmatrix}. \tag{3.55}$$

Now, we calculate the vector $\xi = (\phi, \theta, \psi)^T$ and the scalar θ_M by an GDM procedure for each time step using the quadratic cost function

$$V = \frac{1}{2}(\tilde{u} - \tilde{f}(\xi, \theta_M))^T(\tilde{u} - \tilde{f}(\xi, \theta_M)). \tag{3.56}$$

Minimizing Eq. (3.56) is done by computing a change in V

$$\Delta V = -(\tilde{u} - \tilde{f}(\xi, \theta_M))^T(\tilde{B}_1 \Delta \xi + \tilde{B}_2 \Delta \theta_M) < 0, \tag{3.57}$$

where $\tilde{B}_1 = \frac{\partial \tilde{f}(\xi, \theta_M)}{\partial \xi} \in \mathfrak{R}^{3 \times 3}$ and $\tilde{B}_2 = \frac{\partial \tilde{f}(\xi, \theta_M)}{\partial \theta_M} \in \mathfrak{R}^{3 \times 1}$. This requires the computation of appropriate $\Delta \xi$ and $\Delta \theta_M$ that make $\Delta V < 0$. These are given as:

$$\Delta \xi = \begin{pmatrix} \Delta \phi \\ \Delta \theta \\ \Delta \psi \end{pmatrix} = \tilde{B}_1^T \alpha (\tilde{u} - \tilde{f}(\xi, \theta_M)), \tag{3.58}$$

and

$$\Delta\theta_M = \tilde{B}_2^T \beta(\tilde{u} - \tilde{f}(\xi, \theta_M)). \quad (3.59)$$

In the above equations,

$$\alpha = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{pmatrix}; \quad \beta = \begin{pmatrix} \beta_1 & 0 & 0 \\ 0 & \beta_2 & 0 \\ 0 & 0 & \beta_3 \end{pmatrix} \quad (3.60)$$

define the widths of the optimization steps in each component, and $\alpha_i, \beta_i > 0, i = 1, 2, 3$. The optimization process stops either if the norm $\|\tilde{u} - \tilde{f}(\xi, \theta_M)\|$ falls below a defined threshold $\epsilon > 0$, or if the number i of optimization steps exceeds a certain limit. The proper choice of both the widths of the optimization steps and α_i, β_i is so far achieved by trial-and-error. One can use here a learning procedure in order to avoid the current tedious trial-and-error process.

3.6.4 The outer-loop position control

The position controller is designed in the same manner as the velocity/altitude controller. The only changes that have to be introduced to the design procedure presented in the previous section are:

- Eq. (3.53) is replaced by:

$$\begin{aligned} \ddot{x} &= k_x(x - x_d) + ki_x \int (x - x_d) dt + k_{\dot{x}} \dot{x}, \\ \ddot{y} &= k_y(y - y_d) + ki_y \int (y - y_d) dt + k_{\dot{y}} \dot{y}, \\ \ddot{z} &= k_z \dot{z} + k_z(z - z_d), \end{aligned}$$

where k_x, k_y, k_z are proportional gains; ki_x, ki_y integral gains and $k_{\dot{x}}, k_{\dot{y}}$ and k_z derivative gains. They are designed according to a required performance in terms of time responses. Contrarily to the velocity controller, the position controller is limited by its step response behavior to reasonable and bounded amplitude. This has a direct impact of how a trajectory tracking routine has to be fed to the controller, with respect to the sampling period, for new reference values of the position to be tracked.

- Eq.(3.55) is accordingly replaced by:

$$\tilde{u} = \begin{pmatrix} k_x(x - x_d) + ki_x \int (x - x_d) dt + k_{\dot{x}} \dot{x} \\ k_y(y - y_d) + ki_y \int (y - y_d) dt + k_{\dot{y}} \dot{y} \\ k_z \dot{z} + k_z(z - z_d) \end{pmatrix}.$$

3.7 Summary

In this chapter the design of two different types of flight controllers was presented in detail. Common to both flight controllers is a model-based design, the use of nonlinear control approaches, and an inner-/outer-loop control scheme. The main contribution is the use of novel nonlinear control techniques: fuzzy control and GDM optimization.

The fuzzy control approach utilizes both heuristic fuzzy control and model-based fuzzy control combined in an outer- and inner-loop control scheme. The heuristic fuzzy control part of the overall flight controller is used in the outer loop for the generation of desired velocities and heading. The fuzzy model-based part is used in the inner-loop of the overall flight controller for the simultaneous regulation of attitude angles and altitude. The combination of the two results in a horizontal velocity controller that maintains desired velocities at a given altitude. The use of LMI-based H_∞ design provides for global stability and robust behavior in the face of external disturbances. The main disadvantage of the fuzzy model-based design is its conservativeness. However, this is an intrinsic property of all designs based on a global quadratic Lyapunov function and recent results on piece-wise quadratic Lyapunov functions can be used instead.

The GDM-based flight controller design results in a position or velocity flight controller. GDM optimization is used in the outer-loop to obtain desired attitude angles given desired horizontal velocities or positions. As a by-product of the GDM optimization one also obtains a desired value for the main rotor collective pitch. The availability of the latter has as a consequence that the inner-loop rotational dynamics becomes linear. In turn, powerful linear robust design techniques can be utilized for the design of the inner-loop attitude controller. The main disadvantage of the GDM approach is the tedious process of trial-and-error tuning of the width of the optimization steps. This can be circumvented by the use of learning techniques.

Chapter 4

Simulation results

4.1 Introduction

The purpose with this chapter on numerical experimentation is to demonstrate the following two features of the flight controllers developed in Chapter 3.

- First, their robustness w.r.t. external disturbances such as wind and mass change, and noise on the attitude control signals.
- Second, the stability to perform aggressive flying defined by fast acceleration/deceleration, climb/descent, and curvilinear trajectory tracking at high speed.

The experimental results reported here are derived in simulation using the nonlinear model described in Chapter 2. However, this model is a simplification (see Section 2.4.1) of the original nonlinear model. The control performance of both models is verified through simulation. The experiments on robustness solely relate to the robustness of the inner-loop control. The reason for this is as follows: the control of VTOL is done by changing the attitude angles for a desired altitude. The experiments on “aggressive” flight relate to the flight controller as a whole, i.e., both the inner- and outer-loop controllers.

The structure of this chapter is as follows: Section 4.2 presents the experiments performed with the fuzzy flight controller, where attitude and altitude robustness are addressed in Sections 4.2.2 and 4.2.3 respectively. In Section 4.2.4, aggressive flight experiments in terms of fast acceleration/deceleration, turns based on heading control, and curvilinear motion at high speed are presented. Section 4.3 presents the experiments performed with the GDM controller concerning velocity, altitude and position control. The experiments performed consider external disturbances such as wind, body mass change and noise on control inputs at the

attitude level. An approach to damp the wind effect on horizontal control will be presented in Section 4.2.4.

4.2 Simulation with the fuzzy flight controller

4.2.1 Robustness

To illustrate robustness we consider in this section

- the FGS attitude controller where the control input $s(b_{1s}, a_{1s}, \theta_T)$ to the attitude angles are subject to noise.
- the FGS altitude controller subject to external disturbances in terms of mass change and wind change.

4.2.2 Attitude control robustness

The numerical experiments are performed with the FGS controller from Section 3.3. This controller is used on the original and the simplified nonlinear models presented in Chapter 2 Section 2.4. The aim here is to compare control results for both the models in order to:

- verify that the simplified model is a good enough substitute for the original one.
- verify that the controller –designed for the simplified model– can also be used for the original one.

Experiment 1 *The task is to regulate the attitude angles around certain desired values (set-point control), given that the control inputs for the attitude angles are affected by white noise. The experiment is performed with a constant mass of 50 Kg and a constant wind speed of 10 m/s.*

In the context of this experiment (see Fig. 4.1), we:

- compare the simplified and original models' response to set-point control of attitude angles,
- compare the controller performance with and without noise on the control inputs for the attitude angles.

In the experiment illustrated in Figure 4.1, the attitude angles are subject to set-point control at their extreme values ($[\phi; \theta; \psi] = [\pm \frac{\pi}{4}; \pm \frac{\pi}{4}; \pm \pi]$). The left side of the figure presents results for the simplified model, and the right side for the original one. The middle part of the figure illustrates the injection of a 5 to 10% white noise to the attitude control signals. As one can see, this does not affect the performance of the controller. The settling time for the pitch and the roll is 6 sec. each, and it is 3 sec. for the yaw. The bottom part of the figure illustrates the magnitude of the control inputs to the attitude angles. After scale transformation, these control inputs are the same for the simplified and original models.

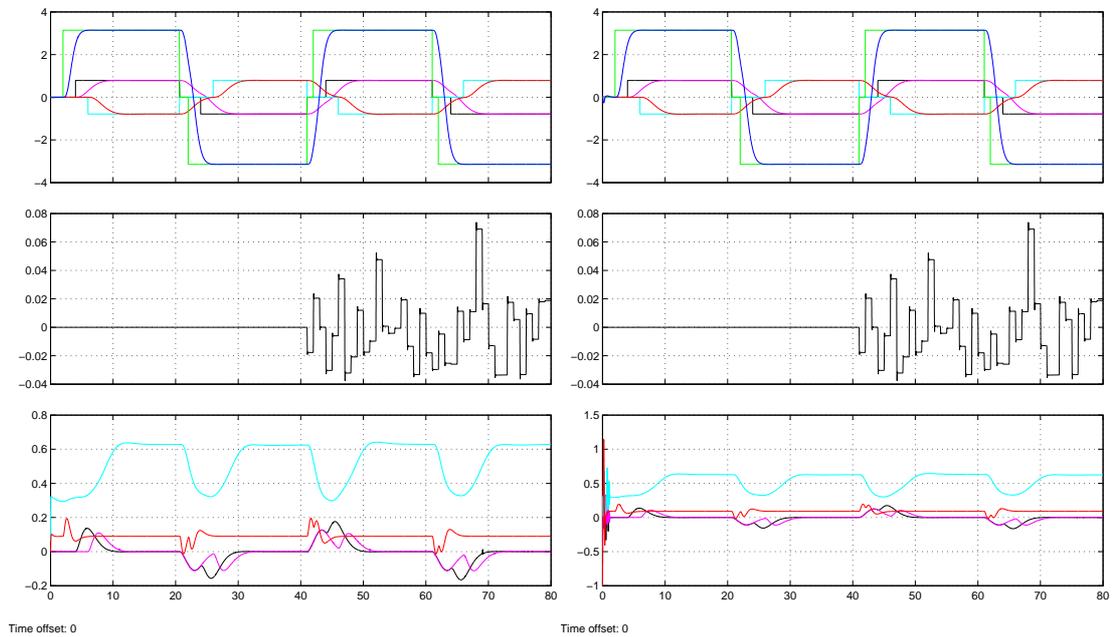


Figure 4.1: Exp.1: Attitude set-point regulation

We should be able to perform attitude control within the above-specified ranges without saturating the servo-actuators. Figure 4.2 (upper-left part) shows a comparison between the outputs from simplified and original servo-actuators. The lower-left part of the figure illustrates the impact of the outputs from the servo-actuators on the trust force: A slight drift of the collective pitch has a direct influence on the trust force of $\pm 5N$. The right-side of the figure illustrates the difference between the regulated attitude angles from the simplified and original models.

The need for the results presented in Fig. 4.2 is as follows. The output of the servo-actuators causes a change in the main rotor force. So it is necessary to verify that the simplifications made both at the level of both the servo-actuators and at the attitude dynamics approximate as close as possible their counterparts from the original model. We can see from Fig. 4.2 that this is indeed the case.

It has to be stressed here that the white noise model provided with the original model is of lower frequency than the one presented in our experiments. Thus a noise with a higher frequency may affect the control inputs to the attitude angles by overloading the servo-actuators. In order to avoid this we introduce a 3rd-order filter to cut the high-frequencies of the white noise, see Fig. 4.3.

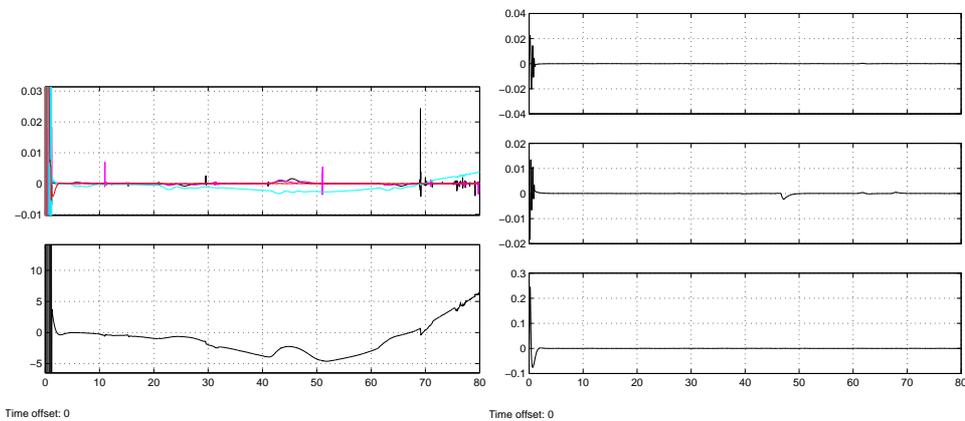


Figure 4.2: Input signals and main rotor force (left), and attitude angles (right) comparisons

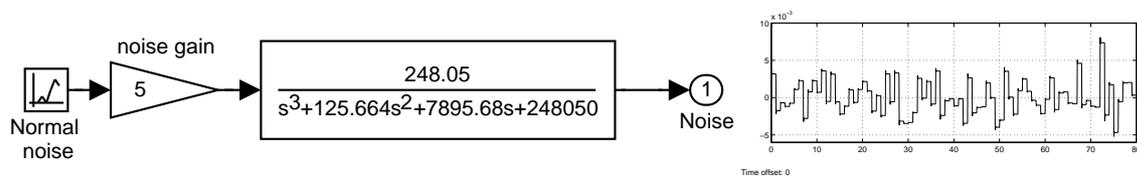


Figure 4.3: Filter for the noise on the input signals for the attitude angles

The noise effect is simulated as white noise with a used in the experiments is simulated as white noise with a mean-value 0 and variance 2 to 5 (see Fig. 4.3). It models roughly the noise that can be introduced by vibrations on the servo-actuators.

Experiment 2 *The task is to track desired trajectories for the attitude angles given that the control inputs to these angles are affected with noise. The experiment is performed with a constant mass of the helicopter of 50 Kg and a constant wind speed of 10 m/s.*

Figure 4.4 (left-side) shows the tracking errors for the attitude angles in the case of the simplified model and on the right-side we have the tracking errors for the original model. The attitude trajectory tracked is a sinusoidal one, and there is 5 to 10 % white noise added to the control inputs for the attitude angles. As one can see, the noise does not affect the performance of the controller. The settling time is approximately 3 sec. for the pitch and roll, and about 2 sec. for the yaw. The bottom part of the figure illustrates the control inputs to the attitude angles for the simplified and the original models. Here again, after a scale transformation, these turn out to be the same.

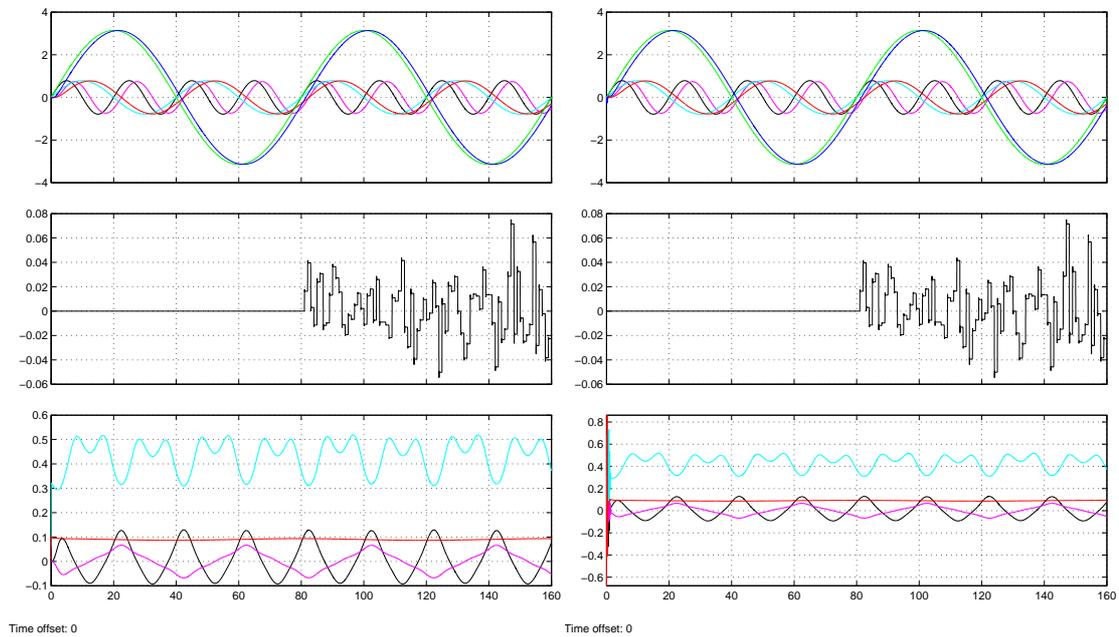


Figure 4.4: Exp.2: Attitude tracking

Figure 4.5 (left-bottom) compares the main rotor force associated with the simplified and original models. The main rotor force error does not in this case exceed $\pm 4N$, which is the equivalent of a fluctuation of the body mass of $\approx 0.4 \text{ Kg}$. This force is induced by the control inputs to the attitude angles illustrated by the left-top part of the figure. The right part of the figure presents the error between the attitude angles resulting from the simplified and original models.

One should point out here that the control inputs to the attitude angles do not exceed the limits, imposed by the servo-actuators of the original model ($[-1, +1]$). Thus the simplified servo-actuators' model –represented as 1st-order transfer function– with time constant $\tau = 20msec.$ and a saturation bounds $[-1, +1]$ does approximate well enough the original servo-actuators' model.

4.2.3 Altitude control robustness

We present here altitude set-point and tracking control by taking into account external disturbances such as wind and mass change. The simulations are performed on both the simplified and original models. Four cases are covered:

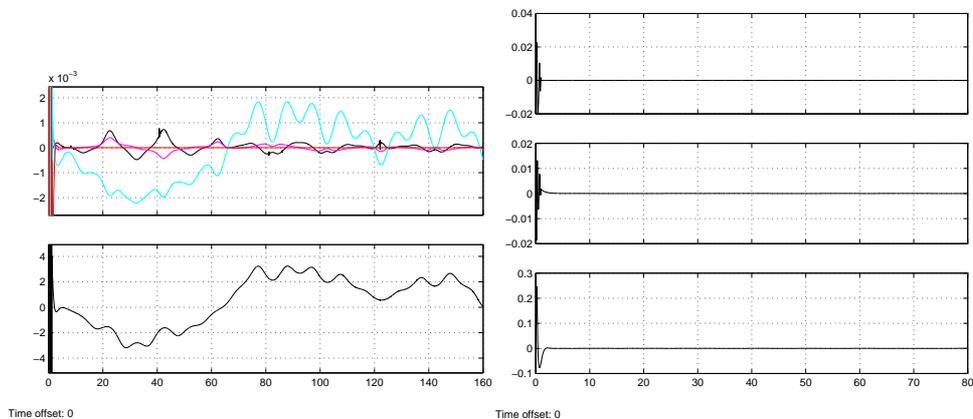


Figure 4.5: Input signals and main rotor force (left), and attitude angles (right) comparisons

- constant wind speed and body mass (Exp. 3)
- wind change and constant body mass (Exp. 4)
- constant wind and body mass change (Exp. 5)
- varying wind and body mass change (Exp. 6)

We will close this section by mentioning the limitations related to altitude control and the technical solution opted for in order to enhance its capabilities.

Altitude control with constant wind and mass

Experiment 3 *The task is to regulate around desired altitude set-points for the simplified and original models, body mass of 50 Kg and a wind speed of 10 m/s.*

Figure 4.6 (left side) shows results for the simplified model and on the right side are the results from the original model. The top of the figure illustrates a climb from 0 to 20m and then a descent from 20 to 0 meter. The settling time –in the case of both models– is 3 sec. The second from the top part of the figure illustrates the control signals delivered from the FGS controller for the simplified and original models. As one can see these are identical.

Figure 4.7 (bottom) presents the error between the main rotor forces delivered by the two models. The top part shows the error between the control signals (collectives and cyclics) for the two models that produce these main rotor forces. The force error between real and simple model does not in this case exceed $\pm 10N$

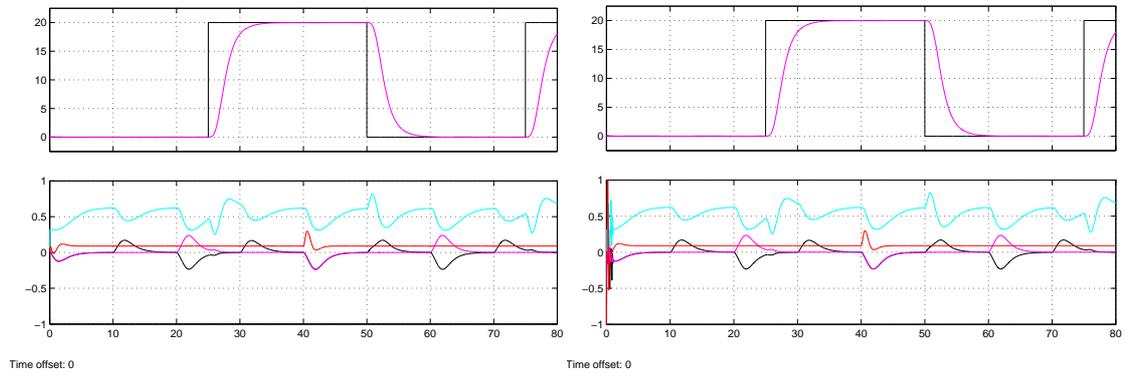


Figure 4.6: Exp.3: altitude control with constant wind and body mass

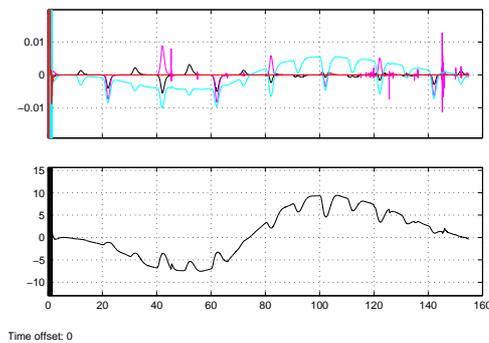


Figure 4.7: Input signals and main rotor force comparisons

which is the equivalent to a fluctuation of the body mass of $\approx 1Kg$. All this point out that the FGS-based attitude/altitude controller has the same performance in the case of the simplified and original models.

Figure 4.8 top illustrates the performance of the controller when tracking a sinusoidal altitude trajectory: top-left for the simplified model and top-right for the original model. The bottom part of the figure shows the control signals needed to achieve the tracking in the case of the simplified (bottom-left) and original (bottom-right) models. As one can see these signals are identical.

Wind effect on altitude control

Experiment 4 *The task is to track a desired altitude trajectory given a wind variation and constant mass of 50 Kg. The experiments are performed for both the simplified and original models. The wind variation is presented in two main ways:*

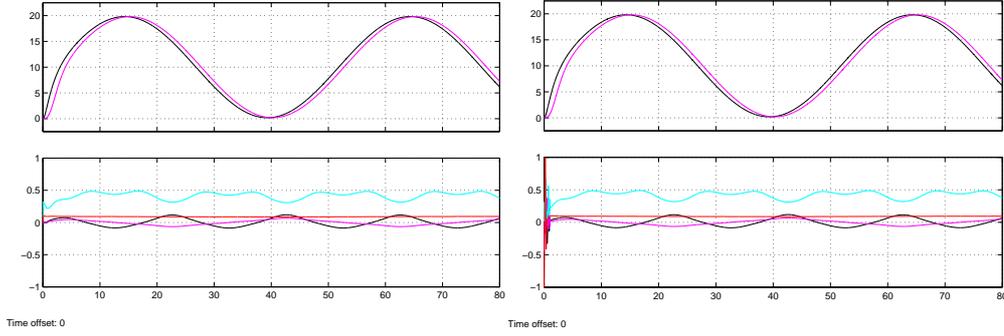


Figure 4.8: Altitude-tracking with constant wind and body mass

The wind model takes as input the wind speed V_W (in 3 directions), and is described by the following formula

$$\vec{\mathcal{F}}_W = N_w(0, 2) + \frac{1}{2}C_d A_C V_W^2; \quad \text{with } (A_C = 4\pi R_C^2)$$

where $N_w(0, 2)$ is a stochastic process defined by a white noise of amplitude $1m/s^{-2}$, a mean-value 0, and a variance of 2 and represents the wind turbulence. $\frac{1}{2}C_d A_C V_W^2$ is the cabin reaction to its motion and wind force (cabin drag force). A_C is the area of the cabin in each direction, and C_d is a given drag coefficient. The model of the wind is shown in Fig. 4.9 (top). Second from top, we have the wind turbulence. Third from top represents the cabin drag force profile, and at the bottom we have the sum of wind turbulence and cabin drag force.

A variation of the wind speed within the range $[-10, 10]m/s$ for both horizontal and vertical directions will produce the signal shown in Fig. 4.10. This signal represents the external disturbance due to variations in the wind speed.

Figure 4.11 (top-left) shows altitude tracking for sinusoidal altitude trajectory for the simplified and the original models. The wind speed follows a sinusoidal trajectory as well, see Fig 4.11 (bottom). The middle part of the figure shows the control inputs (collectives and cyclics) needed to track the desired altitude. Again after a scale transformation these are identical for the simplified and original models.

Mass effect on altitude control

A change of mass has an effect on the acceleration of the solid body. In the VTOL case, the force provided by its rotor counteract the effect of gravity by its lift component. The total force of the rotor has to be big enough to : 1) afford the lift necessary to maintain the heave of the helicopter; and/or 2) perform ascend motion,

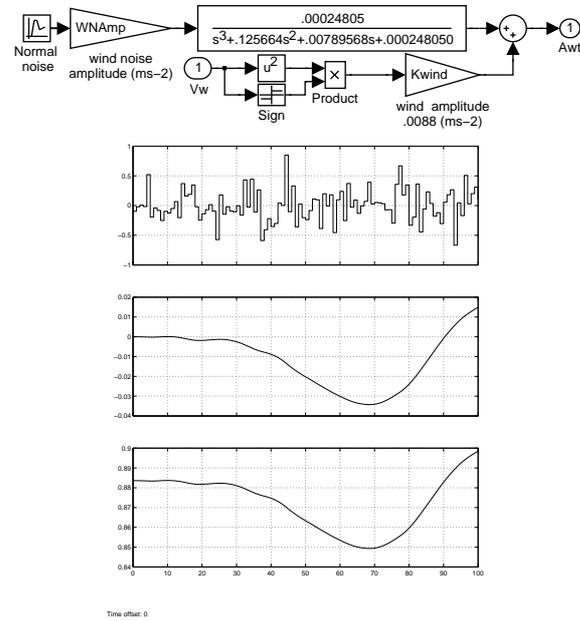


Figure 4.9: Wind model and wind force components

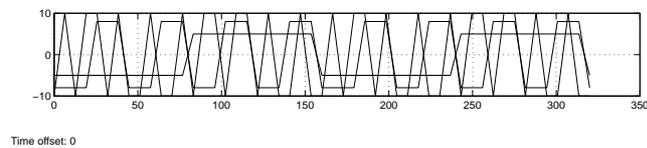


Figure 4.10: Wind disturbance signal

and/or 3) produce a trust for horizontal motion enough large to counteract the drag due to wind action on the body.

The mass of the helicopter may change for different reasons. One is that the helicopter is loaded with a sensory platform, which is equivalent to a mass increase 0 to 20 Kg. The other reason is the gas volume decrease, which varies the mass of the helicopter from 50 Kg (without payload) to 45 Kg –for a fuel reservoir of 5 liters and a fuel consumption constant along time.

Experiment 5 *The task is to track for a desired altitude trajectory given varying body mass. The experiment is performed both for the simplified and original models, constant win speed of 10 m/s, and the mass varying between 45 and 75 Kg.*

Figure 4.12 top shows the results from tracking a sinusoidal altitude trajectory for the simplified and

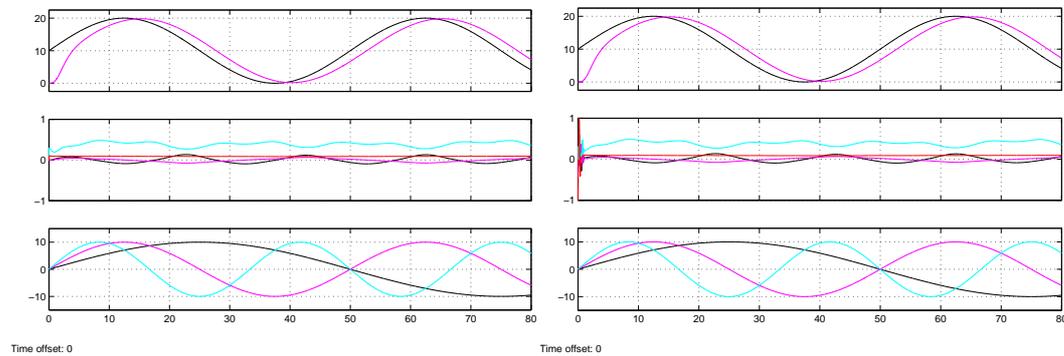


Figure 4.11: Exp.4: Altitude-tracking with varying wind speed

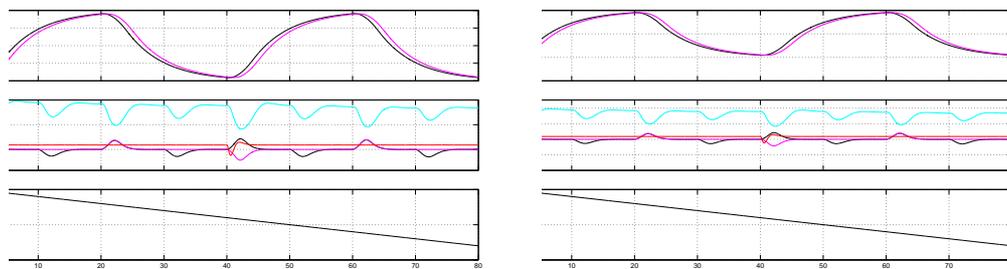


Figure 4.12: Exp.5: Altitude-tracking with decreasing body mass

original models. The middle part of the figure presents the control signals (cyclics and collectives) needed to achieve the tracking. The bottom part of the figure illustrates the varying mass profile during tracking.

Wind and mass effects on altitude control

Experiment 6 *The task is to track an altitude trajectory tracking in to account the accumulated effect of wind and body mass changes.*

Figure 4.13 (top) shows the tracking of a sinusoidal altitude trajectory, given a varying wind speed (Fig. 4.13, bottom) and a decreasing mass (Fig. 4.13, 3rd from top). Second from top, we present the control inputs (collectives and cyclics) needed to achieve the tracking. The left part of the figure present the results for the simplified model and the right part for the original one. After a scale transformation, the performance of the altitude controller for the simplified and original models turns out to be identical.

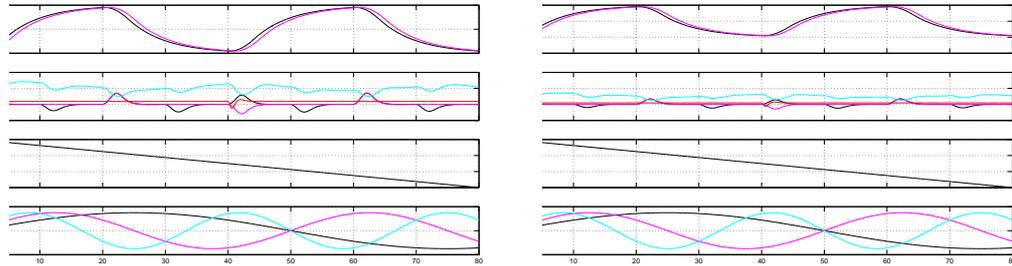


Figure 4.13: Exp.6: Altitude-tracking with wind and body mass changes

Limitation on the altitude controller

There are two limitations imposed on the action of the altitude controller.

1. a limit on the control output that is, all of the control signals producing the cyclics (pitch and roll) and collectives (main and tail) are in the interval $[-1, +1]$. This is due to limitations on the signals affordable by the servo-actuators.
2. a limit w.r.t the magnitude of change in reference value for the altitude. It turns out that a new reference value can be set max 10 m away from the previous one.

To cope with the second limitation, we adopt a simple 1st-order integrator with saturation. It is similar to the one used for the actuator, but with a time constant $\tau = 20msec$ and a first order filter with time constant $\tau = 9sec$. The integrator shapes the altitude reference value from step to ramp, and in this way, allows for a change of reference values for up to 300m. This is achieved still with the control inputs (cyclics and collectives) being within the range of $[-1, +1]$. It has to be noted here that all the following experiments are performed with the original model using the controller derived on the basis of the simplified model.

4.2.4 Aggressive flying

To illustrate aggressive flying, we consider:

- set-point velocity control for the purpose of fast acceleration/deceleration,
- set-point and tracking control for heading, with the purpose of performing turns and curvilinear motion patterns.

Fast acceleration/deceleration

Experiment 7 The task is to accelerate/decelerate by set-point control of reference velocities, and with a constant heading ($\chi = 0$).

The experiment is performed with body mass of 50 Kg and a wind speed of 10 m/s. The initial velocity reference is $V(t) = 15\text{m/s}$ and the helicopter should switch to $V(t) = 5\text{m/s}$, i.e., we have the case of deceleration.

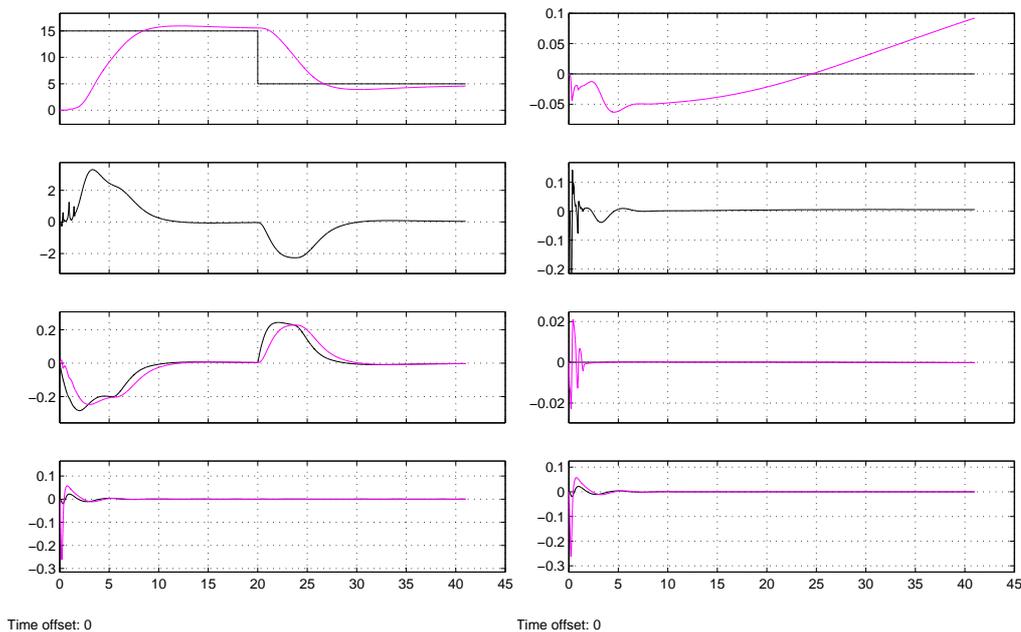


Figure 4.14: Exp.7: \dot{x} (left) and \dot{y} (right) set-point regulation

Figure 4.14 (left-side) illustrates the x-velocity channel, and the right-side illustrates the y-velocity channel. The top part of the figure presents the velocity components as a result of change of velocity reference while keeping the heading ($\chi = 0$). Second from top, we have the corresponding accelerations for the x- and y-velocity channels. Third from top, we illustrate the behavior of the pitch and roll respectively needed to perform this change of reference velocity. At the bottom part of the figure, one sees the behavior of the yaw as a result of keeping the above mentioned heading. The settling time, i.e., the time for which the system settles from $V = 15\text{m/s}$ to 5m/s is approximately sec. At the top part of the same figure, one can see the behavior of the x- and y-velocity channels when the reference speed change from $V = 0\text{m/s}$ to 15m/s , i.e.,

the case of acceleration.

Turns via heading control

In the following experiments, we will illustrate two types of turns: 1) a sharp turn is defined as a change of the reference value for heading, 2) a smooth turn is defined as the tracking of a given heading trajectory. The angle of a turn (ψ), is a function of the heading (χ). The experiments are performed with a body mass of 50 Kg and a wind speed of 10 m/s.

Experiment 8 *The task is to perform sharp turns by changing the reference heading while keeping constant velocity ($V(t) = 17m/s$). The reference heading should take successively the following reference values ($\chi(t) = 0, \pi/2, \pi, 3\pi/2$) where each of these reference values is kept for ($t = 20sec.$).*

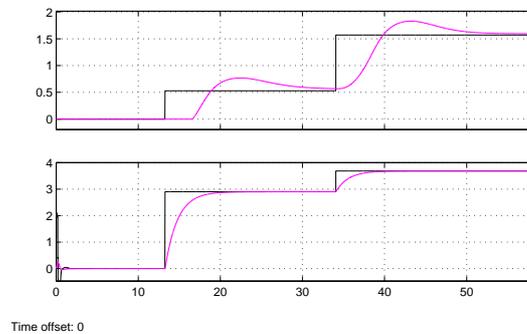


Figure 4.15: Exp.8: Sharp turns

Figure 4.15 (top-part) presents changes in reference and actual heading. The bottom part describes the turn angles corresponding to the above changes in heading. The settling time for the yaw is about 3 sec, while the heading settling time is about 8 sec.

Experiment 9 *The task is to perform smooth turns by tracking a reference heading trajectory while keeping constant velocity ($V(t) = 15m/s$). The reference heading trajectory is given as ($\chi(t) = t/10, \chi \in [0, 2\pi]$ modulo π).*

Figure 4.16 (top) presents changes in reference and actual heading. The bottom part describes the smooth turns corresponding to the above reference trajectory. The settling time for the heading/yaw is about 3 sec.

Curvilinear motion patterns at high speed

Executing curvilinear motion patterns can be done in two ways: 1) specifying a desired pattern in terms of Cartesian coordinates, 2) using heading, yaw and speed control. In the first case, one would need a position

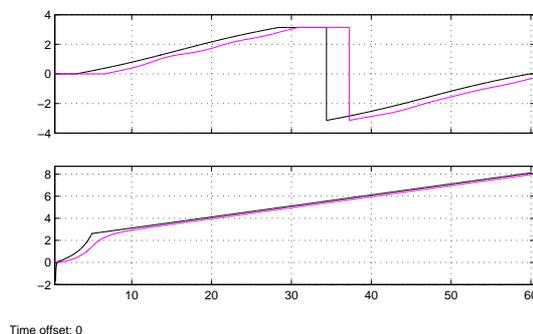


Figure 4.16: Exp.9: Smooth turns

controller. However, the fuzzy flight controller is not a position controller. So in this case we will resort to the second option, where the desired curvilinear motion pattern or trajectory is defined in terms of desired velocity magnitude V^d , and its orientation χ^d . Another issue here is the way in which the helicopter flies along a desired pattern or curvilinear trajectory, defined in the above terms. Let ψ^d or yaw be the angle between the nose of the helicopter and the x-axis of the inertial frame. Given a desired trajectory in the body frame, the desired value ψ^d is a function of \dot{x} and \dot{y} transformed from the body to the inertial frame.

1. flying nose-on-the-trajectory means tracking of ψ^d .
2. flying nose-off-the-trajectory means that we maintain the yaw angle ψ^d constant.

In the next experiments, we will perform nose-on-the-trajectory type of flights.

Experiment 10 *The task is to follow a predefined rectangular motion pattern. This pattern is defined by successive changes of desired heading $\chi(t) = 0, \pi/2, \pi, 3\pi/2$. The desired magnitude of the velocity is $V(t) = 17m/s$ where each of these reference values is kept for ($t = 25sec.$).*

Figure 4.17 (right-side) describes the rectangular pattern flown. In the left-side (top-part) of the figure, we have the behavior of the \dot{x} -channel. Second from top, we have the behavior of the \dot{y} -channel. Third from top, we have the yaw profile during the execution of the rectangular motion.

Experiment 11 *The task is to follow a predefined circular motion pattern. This pattern is defined by desired heading trajectory $\chi(t) = t/10, \chi \in [0, 2\pi]$ modulo π . The desired magnitude of the velocity is $V(t) = 17m/s$.*

Figure 4.18 (right-side) describes the circular pattern flown. In the left-side (top-part) of the figure, we have the behavior of the \dot{x} -channel. Second from top we have the behavior of the \dot{y} -channel. Third from top, we have the yaw profile during the execution of the rectangular motion.

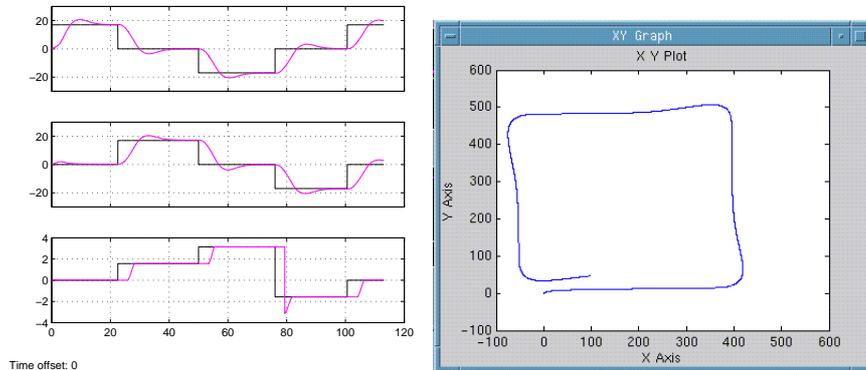


Figure 4.17: Exp.10: Rectangular pattern

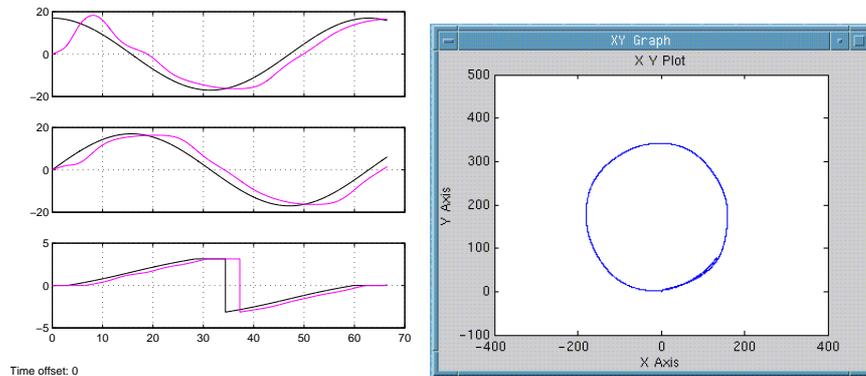


Figure 4.18: Exp.11: Circular pattern

Compensation strategies for wind/mass variations

So far, we have shown the robustness of the inner-loop controller (altitude/attitude controller) with respect to wind and body mass changes. However, flying curvilinear trajectories requires the use of the outer-loop controller, which should also be robust w.r.t the above two factors. In the inner-loop control, the robustness is achieved via the use of H_∞ control principles. In what follows, we will describe how robustness is achieved in the outer-loop control level.

One method could be to design the compensation in the form of Mamdani type of fuzzy controller with a proportional gain generating off-sets to the desired attitude angles $\Delta\theta$ and $\Delta\phi$. These offsets will be added to the desired attitude angles already provided by the outer-loop velocity controller (θ^d and ϕ^d). Figure 4.19

shows a scheme for the design of such a compensator.

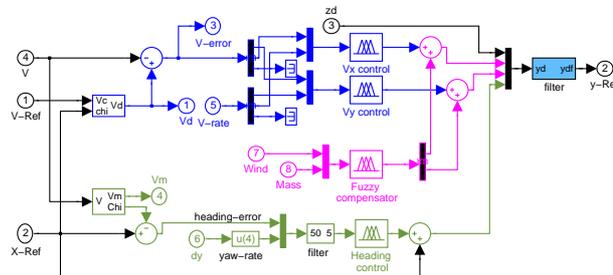


Figure 4.19: Fuzzy compensator scheme

The rules for the compensator may depend on the ratio body-acceleration/wind-acceleration. The first depends on the body mass and the other can be obtained by sensing of wind speed. The following example-rule illustrates a part of this compensator.

IF wind – speed is Small and mass is Medium **THEN** $\Delta\phi$ is Neg

In what follows, we will describe the method actually used to compensate for wind changes. We map the desired attitude angles –computed by the Mamdani fuzzy controller– to different ranges of wind speed. This mapping is done for a horizontal velocity V and heading χ both set to zero. The action of the controller under different wind speeds shows a different behavior and delivers non-zero roll and pitch angles.

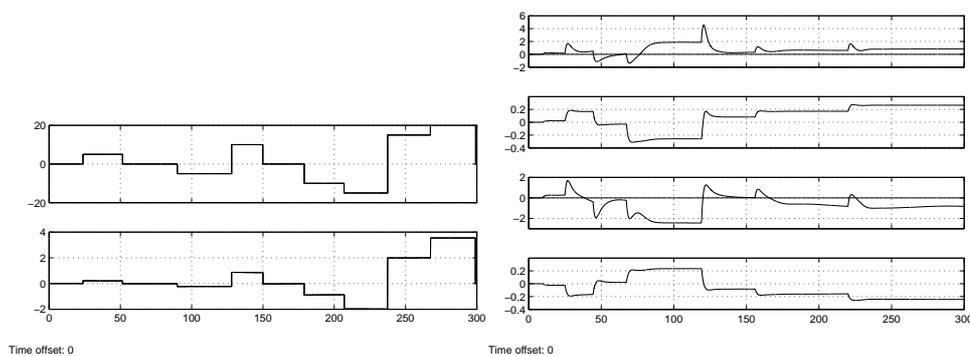


Figure 4.20: Drift effect of wind changes on horizontal velocities

On the left-side of Fig. 4.20, we see the changing wind speed and its acceleration. The right side illustrates from top to bottom the steady state errors for longitudinal velocity, pitch angle, lateral velocity and roll angle

respectively.

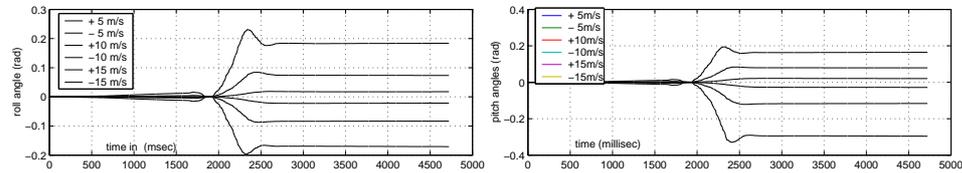


Figure 4.21: Wind action on the attitude angles: roll (left) and pitch (right)

Figure 4.21 illustrates the mapping between the wind speed change and the roll (left) and pitch (right) angles responses respectively. We use a spline function to determine values of the offsets for the attitude angles for each given wind speed. These offsets are subtracted from the desired attitude angles (pitch and roll), and then the so obtained new desired attitude angles are regulated by the controller. Figure 4.22 shows the spline function extracted from the mapping between wind speeds and attitude angles.

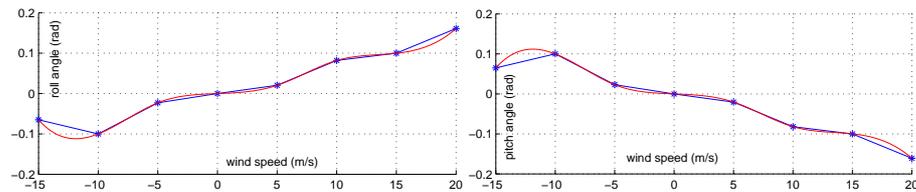


Figure 4.22: Wind mapping for the roll (left) and pitch (right)

Figure 4.23 shows the alternative wind compensator effectively implemented and used in the fuzzy flight controller.

Figure 4.24 illustrates the behavior of the horizontal velocities and their associated attitude angles under the action of different wind speeds. This behavior is represented with and without compensation for the wind action. At the top of the figure we have the comparison for \dot{x} with and without compensation for the wind action. Right below it, we have the same comparison for the associated pitch angle. Third from top of the figure we have the comparison for \dot{y} with and without compensation for the wind action. Right below it, we have the same comparison for the associated roll angle.

From the figure, we can see the following two aspects:

1. the velocities obtained with compensation are closer to the desired velocity, which is equivalent to say

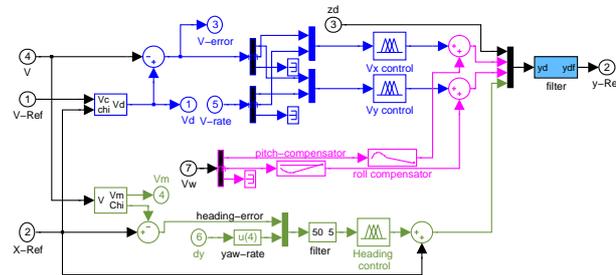


Figure 4.23: Wind mapping compensator scheme

that the helicopter “resists” the drift induced by the wind action. The steady-state error induced by the wind action is reduced.

2. the compensator has an effect also on the time response. Indeed, for the horizontal velocities to settle to their desired value, the attitude angles has to rise “fast” enough to their desired values.

Change of the body mass has the same effect as the one resulting from changes in wind speed (gusts). A loss of mass produces an increase of the body accelerations. This, we can have an equivalent mapping to the one presented above: mapping the attitude angle ranges of mass change, and then use this mapping to compensate in the same manner as in the case of wind changes. This suggests the introduction of body mass as an additional input to the velocity controller.

4.3 Simulation results for GDM controller

The numerical experiments are performed with the GDM controller presented in Section 3.6. We use here a simplified version of the nonlinear model and illustrate:

1. horizontal velocity control (Section 4.3.1)
2. vertical motion control (Section 4.3.2)
3. position control (Section 4.3.3).

The body mass of the helicopter is kept unchanged at 50 Kg for all experiments.

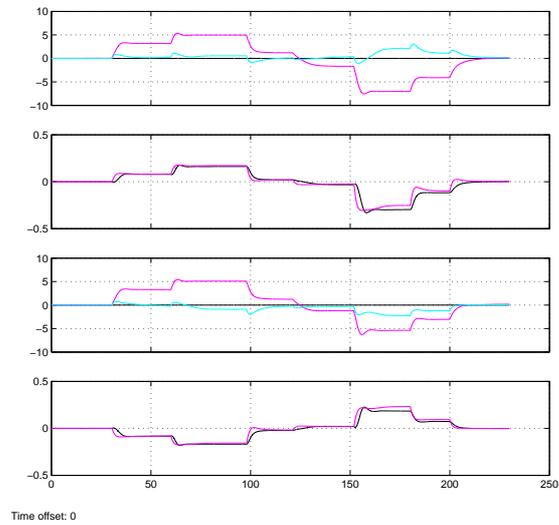


Figure 4.24: Wind action with and without compensation

4.3.1 Horizontal velocity control

Experiment 12 *The task is set-point regulation of horizontal velocity under strong (10 m/s) and weak (3 m/s) winds.*

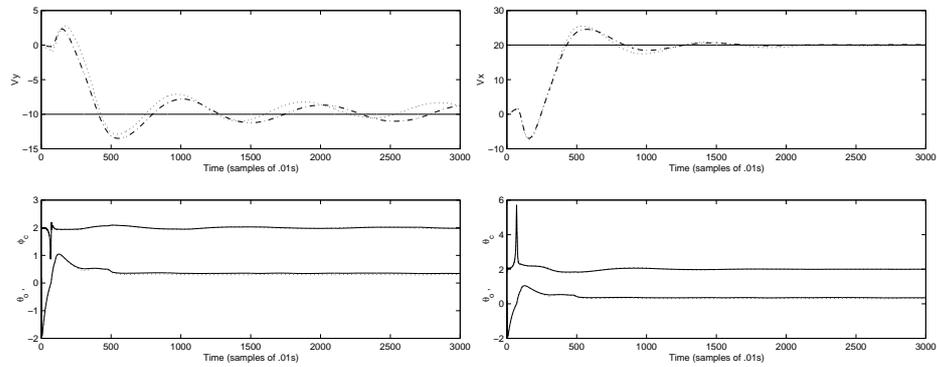


Figure 4.25: Exp.12: Low \dot{y} and high \dot{x} set-point regulation

Figure 4.25 (left-side) shows the results from set-point regulation around a desired lateral velocity $V_y^d = -10m/s$. The right-side of the figure illustrates regulation of high desired longitudinal velocity around a set-point $V_x^d = 20m/s$. The upper-left part shows low \dot{y} -velocity set-point regulation with strong and weak

wind action where: desired velocity $-$; velocity under strong wind $-.-$; and velocity under weak wind \dots are represented. The lower-left part illustrates the control inputs (b_{1s}, θ_M) for strong and weak wind respectively.

The response is oscillatory-damped and settles to its reference value after a time narrow to 40 sec. We can notice a similar response to strong and weak wind action for the lateral velocity control. The upper-right part shows high \dot{x} -velocity set-point regulation with strong and weak wind action. The lower-right part illustrates the control inputs (a_{1s}, θ_M) for strong and weak wind respectively. The settling time for longitudinal velocity is about 20 sec.

Experiment 13 *The task is tracking a desired velocity trajectory under strong (10 m/s) and weak (3 m/s) winds.*

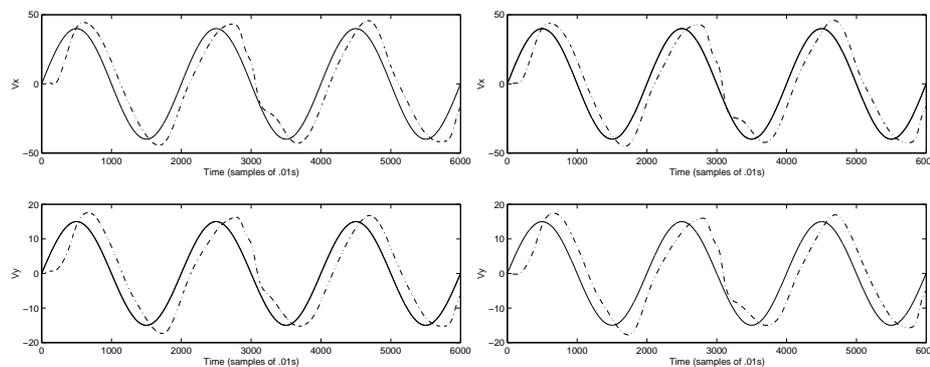


Figure 4.26: Exp.13: Velocity tracking with strong and weak wind

In the left-side of fig 4.26, we see the tracking results under the strong wind for both V_x (up) and V_y (down). In the left part we illustrate the tracking of these velocities under weak wind. The figures show desired versus actual trajectories. The time delay present when tracking a sinusoidal velocity reference along both the x- and y-channels is about 2 to 3 sec. for each channel and there is an overshoot of about 5%. A better tuning of the control coefficients would give a better speed profile and a smaller tracking error.

Experiment 14 *The task is a regulation under a set-point change, given a strong (10 m/s) and weak (3 m/s) wind.*

Figure 4.27 (Left-top) presents the behavior of the lateral velocity during set-point change and left-bottom part of the figure represents the associated control signals (collective pitch and cyclic roll). The right-top part of the figure shows of the behavior of the longitudinal velocity during set-point change and left-bottom part

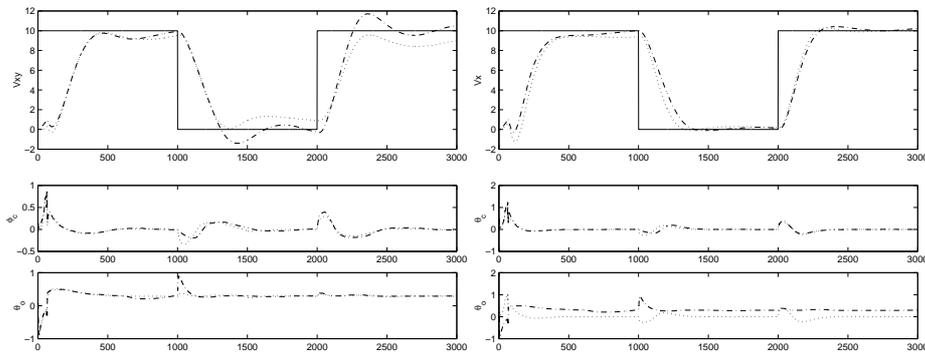


Figure 4.27: Exp.14: \dot{y} and \dot{x} set-point change with strong and weak wind

of the figure represents the associated control signals (collective pitch and cyclic pitch). Both results are obtained for strong and weak wind. The settling time for the x-channel is about 20 sec. but it is longer for the y-channel because of its oscillatory profile.

Unlike the case of velocity tracking, the change in the velocity reference values shows a behavior that is affected by the wind. This behavior is more damped in the case of x-channel and oscillatory in the y-channel. Furthermore, and the effect of the wind on the collective pitch is clearly distinguishable in the case of longitudinal speed control. The effect of the wind on the y-channel shows that stronger wind leads to an over-shoot, and weak wind leads to an under-shoot. Here again, tuning the controller parameters can help improving the performance.

4.3.2 Vertical motion control

Experiment 15 *The task here is altitude set-point regulation and altitude trajectory tracking in strong and weak wind.*

Figure 4.28 (Top-left) illustrates altitude set-point regulation for strong and weak wind. As one can see, the effect of wind is negligible. A climb of 50m takes about 5 sec. This corresponds to a fast rise time, and thus demands a collective pitch which grows nearly to its saturation limits. This can be seen in the left-bottom part of the figure. The right side of the figure shows in the top part attitude trajectory tracking. Here again, we see a time delay of 3 sec. In right-bottom part of the figure shows the corresponding control signals.

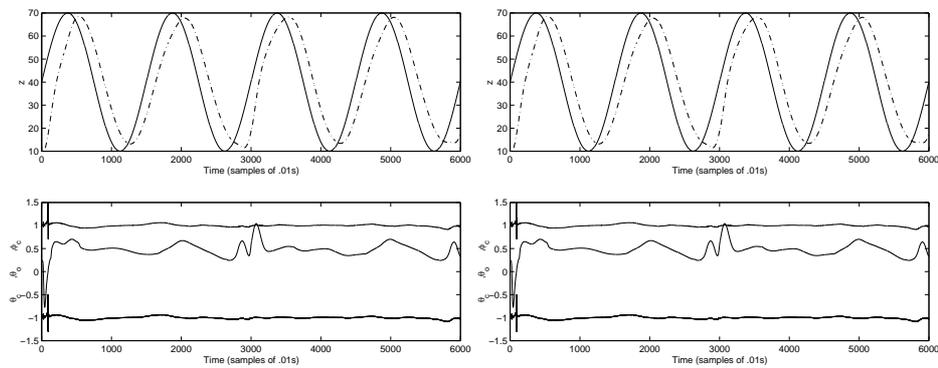


Figure 4.28: Exp.15: Altitude set-point regulation and trajectory tracking.

4.3.3 Position control

Experiment 16 *The task is to perform hover at a certain position $(x,y,z)=(50,50,50)m$ under strong (20 m/s) and weak (3 m/s) wind.*

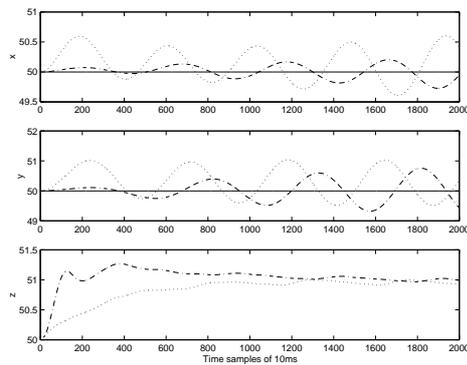


Figure 4.29: Exp.16: Hovering control with strong and weak wind

Figure 4.29 illustrates the results of the experiment. We can see an undamped oscillatory behavior in the x - and y -channels. These are about 0.5 m for x -channel and about 1 m for the y -channel. The oscillations are more pronounced in the case of a wind of 20m/s speed. Notice also that the wind action affects also the z -channel: there is an over-shoot of about 1 m. and steady-state is achieved after 12 sec.

Experiment 17 *The task is to position the helicopter at desired coordinates $(x,y,z)=(50,50,50)m$ starting from another position, under strong (20 m/s) and weak (3 m/s) wind.*

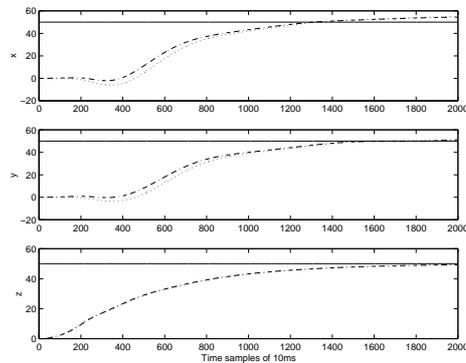


Figure 4.30: Exp.17: Positioning with strong and weak wind

Figure 4.30 illustrates the results of this simulation. As one can see the wind action, either weak or strong, has a negligible effect in terms of error w.r.t the desired position. However the rise time on all three channels seem to be rather slow (≈ 20 sec.), which is not due to the wind action, but rather to some peculiarities of the optimization procedure.

4.4 Summary

In this chapter we presented results from experiments which aim was two-fold:

1. to illustrate the robustness (in the face of external disturbances) of the fuzzy and GDM flight controllers proposed in Chapter 3; and
2. to illustrate the ability of the above mentioned controllers to perform aggressive flying.

The above two aims are fully achieved in the case of the fuzzy flight controller, while in the case of the GDM flight controller, they are partially fulfilled. For example, the fuzzy flight controller was exposed to external disturbances like, wind, mass change and noise, while the GDM based flight controller only to wind and noise.

An important feature of the experiments with the FGS controller is that they were performed on both the simplified and original APID-MK3 helicopter models and despite of this, the flight controller has shown very similar performance.

The results for the GDM based controller can be further improved if more advanced linear robust control methods are used. Only in this case, change of mass can be taken as yet another disturbance. Also, aggressive flying can be performed if robust tracking control methods are used.

Chapter 5

Summary and future work

This thesis presented two novel methods for the design of flight controllers for the unmanned APID-MK3 helicopter.

The flight controllers developed were based on a nonlinear model of APID-MK3 and tested in extensive simulation, showing their robustness in the face of external disturbances and also their ability to perform aggressive flying.

In Chapter 2, we presented in detail the mathematical model of APID-MK3 used for the design of flight controllers. The model is a simplified version of the original APID-MK3 model and is obtained from it under a number of realistic assumptions. Furthermore, we performed comparisons between the simplified APID-MK3 model on one side and two different models of an APID-MK3-like unmanned platform, namely Yamaha R50, on the other. The comparisons show that the APID-MK3 model is realistic enough both from the point of view of a generic VTOL aircraft and a different unmanned platform.

In Chapter 3, we presented the design of two different types of flight controllers: 1) a velocity controller based on FGS and heuristic fuzzy control; and 2) a velocity and position GDM-based controller. Common to both flight controllers is a model-based design, the use of nonlinear control approaches, and an inner-outer loop control scheme.

In Chapter 4, we presented results from experiments which illustrate the robustness of the fuzzy flight controller and illustrate its ability to perform aggressive flying. An important feature of the experiments with the FGS controller is that they were performed on both the simplified and original APID-MK3 helicopter models and despite of this, the flight controller has shown very similar performance.

In this context, the subject of this thesis has been achieving *tactical* autonomy for an unmanned helicopter.

This type of autonomy addresses the execution of basic flight modes such as “take off”, “landing”, “cruise flight” as well as more aggressive flight patterns. Here the major concern is twofold: 1) use and reliability of proprioceptive sensors (compass, GPS, gyros, etc.) to monitor the internal state of the UAV; and 2) robust and stable position/velocity control based on inputs from the UAV’s proprioceptive sensors. Thus this concern is related to the air-worthiness of an UAV in unmanned flight and unmanned landing/take-off. In our case, we were concerned with autonomy for achieving aggressive flying. Also, we assumed the proprioceptive sensors given, and our major concern was robust and stable velocity/position control.

One can address the functional type of UAV autonomy only when the air-worthiness of the vehicle has been verified. The tactical type of autonomy addresses mission execution in a safe and reliable manner. Typical mission examples include “track ground vehicle”, “follow coast line”, “deliver load” and autonomy requires making as few assumptions as possible about the environment encountered during mission execution; and that execution should be sensitive to the environment, and adapt to the contingencies encountered. A major concern in achieving functional autonomy is the use of exteroceptive sensors, like a camera or a laser range finder, to acquire information about the state of the environment as it is at the moment and based on this information to react instantly to it by adopting a behavior that complies with this state alone.

One sensor of interest in the WITAS project is a daylight video camera. Given such a sensor, a crucial issue is how to make it an integral part of the UAV’s control loop so that behaviors for tactical autonomy can be implemented in a robust and safe manner, and made to work under hard real-time constraints. In most cases, this would mean that behaviors are not designed according to the principle “look-then-move” but, according to a totally different “look-and-move” principle. The difference is roughly speaking the following: in the first case, the visual sensor is used to determine set-points as inputs for say, UAV’s attitude (roll, pitch and yaw) controller -thus, roll/pitch/yaw feedback is used to internally stabilize the UAV around the set-points in question; in the second case, a “visual controller” computes directly the roll/pitch/yaw inputs, thus using visual information alone to stabilize the UAV.

The type of control based on the “look-and-move” principle is known as visual servoing and has its origins in robotics, in particular in the area of manipulator control based on the use of exteroceptive sensors such as mono/stereo cameras. As for using visual servoing for the control of airborne vehicles there are only few studies available, and none of these answers some questions related to the fundamental aspects of the nature of visual servoing for UAV’s. The important issue here is: which concepts, principles, architectures, tasks,

and associated techniques, from visual servoing for robotics are applicable to the case of UAVs? After having studied the literature on visual control of UAVs we came to the conclusion that there was no systematic study that could provide the answer. The term “visual servoing” has been applied in a number of cases concerning particular autonomous capabilities of UAVs (fixed-wing and rotor aircraft), but there was no attempt to define: 1) what visual servoing for UAVs actually is?; 2) what are the restrictions on the applicability of existing visual servoing methods to the case of UAVs?; 3) what can be possibly new solutions when existing methods cannot be applied or are too restrictive.

In this context, our future work will aim at providing systematic answers to the above three questions.

Bibliography

- [1] T. Takagi and M. Sugeno, Fuzzy identification of Systems and its Applications to Modeling and Control, In: IEEE Trans. Systems, Man and Cybernetics, SMC-15(1), pp. 116–132, Jan. 1985.
- [2] T.J. Koo et al., Hierarchical Hybrid System Design on Berkley UAV, In: Int. Aerial Robotics Competition, Aug. 1998.
- [3] E. Frazzoli, et al., A Hybrid Control Architecture for Aggressive Manoeuvring of Autonomous Helicopters, In: Proc. 38th Conf. on Decision and Control, Phoenix, AZ-USA, Dec. 1999.
- [4] A. Packard and J. Doyle, The Complex Structural Singular value, In: Automatica, 29(1), pp. 71–109, 1993.
- [5] R.S. Sanchez-Pena and M. Sznaier, *Robust Systems: Theory and Applications*, John Willey and Sons, New York, NY-USA, 1998.
- [6] G. Stein and M. Athans, The LQG/LTR procedure for Multi-variable Feedback Control design, In: IEEE Transactions in Automatic Control, 32(2), pp. 105–114, 1987.
- [7] P. Bendotti, J.C. Morris, Identification and Stabilization of a Model Helicopter in Hover, In: American Control Conf. (ACC'95), 1995.
- [8] J.J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, Englewoods Cliffs, New Jersey, 0763, 1991.
- [9] J.E. Hanser, *Approximate Tracking for Nonlinear Systems with Application to Flight Control*, PhD thesis, College of England, University of California, Berkeley, CA-USA 1989.
- [10] P. Bergsten, *Observers and Controllers for Takagi-Sugeno Systems* PhD thesis Orebro Studies in technology 1, Orebro, Sweden, 2001.

- [11] J.S. Shamma, Linearization and Gain-Scheduling, In: W.S. Levine (ed.) *The Control handbook*, The EE Handbook Series, 20: pp. 388–396, CRC Press, Boca Raton, FL-USA, 1992.
- [12] R. Palm, D. Driankov and M. Hellendoorn, *Model-based Fuzzy Control*, Fuzzy gain scheduling and Sliding Mode Controllers, Springer Verlag, Berlin, Germany, 1993.
- [13] T. J. Koo and S. Sastry, Output tracking control design of a helicopter model based on approximate linearization, In: Proc. 37th IEEE Conf. on Decision and Control (CDC'98), pp. 3635–3640, Tampa, FL-USA, Dec. 1998.
- [14] J. Descrisse and C. Moog, Dynamic Decoupling for Inversible Nonlinear Systems, *Systems and Control letters*, 8: pp. 345-9, 1987.
- [15] H. Shim, T.J. Koo, F. Hoffmann, S. Sastry, A Comprehensive Study of Control Design for an Autonomous Helicopter, In: Proc. 37th IEEE Conf. on Decision and Control (CDC'98), pp. –, Tampa, FL-USA, Dec. 1998.
- [16] J.V.R. Prasad, A.J. Calis, e Y. Pei, J.E. Corban, Adaptive Non-linear Synthesis and Flight Test Evaluation on an Unmanned Helicopter, In: 1999 IEEE Int. Conf. on Control Applications (ICRA'99), 1999.
- [17] E. Frazzoli et al., Trajectory Tracking Control Design for Autonomous Helicopters Using a Backstepping Algorithm, In: American Control Conf. (ACC'00) Proc., pp. 4102–4107, Chicago, IL-USA, Jun. 2000.
- [18] M. Krstic and al., *Nonlinear and adaptive Control Design*, John Willey and Sons, New York, NY-USA, 1995.
- [19] R. Mahony, R. Lozano, Exact Path Tracking Control for an Autonomous Helicopter in Hover Manoeuvres, In: Proc. of the 2000 IEEE Int. Conf. on Robotic and Automation (ICRA'00), pp. 1245–1250, San Francisco, CA-USA, Apr. 2000.
- [20] M. Sugeno, Development of an intelligent Unmanned Helicopter, In: *Fuzzy Modeling and Control, Selected works of M. Sugeno*, T. H. Nguyen and N. R. Prasad (Eds.), CRC Press, pp. 13–43, Boca Raton, FL-USA, 1999.

- [21] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice Hall, New Jersey, USA, 1995.
- [22] H. Hellendorn and D. Driankov, *Fuzzy Model Identification: Selected Approaches*, Springer-Verlag, Berlin, Germany, 1997.
- [23] R. Babuška, *Fuzzy Modeling for Control*, Int. Series in Intelligent Technologies, Kluwer Academic, Boston, USA, 1998.
- [24] H. O. Wang et al., TS Fuzzy Model with Linear Rule Consequence and PDC Controller: A universal Framework for Nonlinear Control Systems, In: Proc. of the 9th IEEE Int. Conf. on Fuzzy Systems, 2: pp. 549–554, San Antonio, TX-USA, May 2000.
- [25] C. Fantuzzi and R. Rovatti, On the Approximation Capabilities of the Homogeneous Takagi-Sugeno Model, In: 5th IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE'96), 2: pp. 1067-1072, New Orleans, LA-USA, Sept. 1996.
- [26] J. S. Shamma, *Linearization and Gain-Scheduling: The Control Handbook*, W. S. Levine (Ed.), The Electrical Engineering Handbook Series, CRC Press, Chap.20, pp. 388-396, Boca Raton, Florida, USA, 1996.
- [27] T.A. Johansen et al., Off-equilibrium Linearization and Design of Gain-scheduled Control With Application to Vehicle Speed Control, In: Control Engineering Practice, 6(2):pp. 167–180, Feb. 1998.
- [28] T.A. Johansen et al., Transient Performance, Robustness and Off-Equilibrium Linearization in Fuzzy Gain Scheduled Control, In: *Advances in Fuzzy Control*, Physica-Verlag, D. Driankov and R. Palm (Eds.), 16:pp. 357–75, Heidelberg, Germany, 1998.
- [29] P. Korba, *A Gain-Scheduling Approach to Model-Based Fuzzy Control*, VDI Verlag GmbH, 837, Fortschritt-Berichte VDI, Reihe 8 Meß-, Steuerungs- und Regelungstechnik Series, Düsseldorf, Germany, 2000.
- [30] K. Tanaka et al., Generalized Takagi-Sugeno Fuzzy Systems: Rule Reduction and Robust Control, In: 9th IEEE Int. Conf. on Fuzzy Systems, 2: pp. 688–693, San Antonio, TX-USA, May 2000.

- [31] L. El Ghaoui and V. Balakrishnan, Synthesis of Fixed Structure Controllers via Numerical Optimization, In: 33rd IEEE Conf. on Decision and Control (CDC'94), 3: pp. 2678–2683, Lake Buena Vista, FL-USA, Dec. 1994.
- [32] S. Boyd et al., *Linear Matrix Inequalities In System and Control Theory*, Studies in Applied Mathematics Series, SIAM, Philadelphia, USA, 1994.
- [33] D. Driankov, H. Hellendorn, M. Reinfrank and R. Palm, *An Introduction to Fuzzy Control*, Springer-Verlag, 2nd Edition, Berlin, Germany, 1996.