Fault Tolerant Flight Control of Unmanned Aerial Vehicles

Iman Sadeghzadeh

A Thesis

in

The Department

of

Mechanical and Industrial Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy at

Concordia University

Montréal, Québec, Canada

January 2015

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared,

By : **Iman Sadeghzadeh**

Entitled : **Fault Tolerant Flight Control of Unmanned Aerial Vehicles**

and submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy (Mechanical Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. F. Haghighat

_____ Examiner
Dr. C.Y. Su

_____ Examiner
Dr. W.F. Xie

_____ Examiner
Dr. A. Aghdam External
Electrical and Computer Engineering

_____ Examiner
Dr. Q. Zhao External
University of Alberta

_____ Supervisor
Dr. Y.M. Zhang

Approved by: _____
Dr. A. Dolatabadi, Ph.D. Program Director
Department of Mechanical and Industrial Engineering

January 13, 2015

_____
Dr. Amir Asif, Dean
Faculty of Engineering & Computer Science

ABSTRACT

Fault Tolerant Flight Control of Unmanned Aerial Vehicles

Iman Sadeghzadeh, Ph.D.
Concordia University, 2015

Safety, reliability and acceptable level of performance of dynamic control systems are the major keys in all control systems especially in safety-critical control systems. A controller should be capable of handling noises and uncertainties imposed to the controlled process. A fault-tolerant controller should be able to control a system with guaranteed stability and good or acceptable performance not only in normal operation conditions but also in the presence of partial faults or total failures that can be occurred in the components of the system. When a fault occurs in a system, it suddenly starts to behave in an unanticipated manner. Thereby, a fault-tolerant controller should be designed for being able to handle the fault and guarantee system stability and acceptable performance in the presence of faults/damages. This shows the importance and necessity of Fault-Tolerant Control (FTC) to safety-critical and even nowadays for some new and non-safety-critical systems.

During recent years, Unmanned Aerial Vehicles (UAVs) have proved to play a significant role in military and civil applications. The success of UAVs in different missions guarantees the growing number of UAVs to be considerable in future. Reliability of UAVs and their components against faults and failures is one of the most important objectives for safety-critical systems including manned airplanes and UAVs. The reliability importance of UAVs is implied in the acknowledgement of the Office of the Secretary of Defense in the UAV Roadmap 2005-2030 by stating that, ”Improving UA [unmanned aircraft] reliability is the single most immediate and long-reaching need to ensure their success”. This statement gives a wide future scenery of safety, reliability and Fault-Tolerant Flight Control (FTFC) systems of UAVs.

The main objective of this thesis is to investigate and compare some aspects of fault-tolerant flight control techniques such as performance, robustness and capability of handling the faults and failures during the flight of UAVs. Several control techniques have been

developed and tested on two main platforms at Concordia University for fault-tolerant control techniques development, implementation and flight test purposes: quadrotor and fixed-wing UAVs. The FTC techniques developed are: Gain-Scheduled Proportional-Integral-Derivative (GS-PID), Control Allocation and Re-allocation (CA/RA), Model Reference Adaptive Control (MRAC), and finally the Linear Parameter Varying (LPV) control as an alternative and theoretically more comprehensive gain-scheduling based control technique. The LPV technique is used to control the quadrotor helicopter for fault-free conditions. Also a GS-PID controller is used as a fault-tolerant controller and implemented on a fixed-wing UAV in the presence of a stuck rudder failure case.

ACKNOWLEDGEMENTS

My greatest thanks and appreciation goes to my supervisor Dr. Youmin Zhang, who has given me the opportunity of joining his research group and provided support for my research. His careful reading of everything I write, and his thoughtful suggestions have vastly improved my research level.

I would like to express the deep appreciation to Dr. Abbas Chamseddine and Dr. Didier Theilliol for their vulnerable helps in many aspects of my research. I would say, the success of this thesis would not have been possible without them.

I show my gratitude to Leslie for her help and support with all administrative works during my stay at Concordia.

I thank my friends Mahyar, Hadi, Mohammad, Miad, Ankit, Narendra and all who always supported and helped me in my research and incented me to strive towards my goal.

This thesis work is dedicated to my wife Sahar, my daughter Nava and my parents Mehdi and Robabeh who have been a constant source of support and encouragement during the challenges of my graduate studies and life. I am truly thankful for having you in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

Nomenclature

T	Thrust, N
u	Input vector
K	DC motor's gain
ω	DC motor's bandwidth, N
ϕ	Roll angle, *rad*
θ	Pitch angle, *rad*
ψ	Yaw angle, *rad*
m	Mass, kg
g	Gravity, m/s^2
L	Rolling moment, $N.m$
M	Pitching moment, $N.m$
N	Yawing moment, $N.m$
d	Force to moment scaling factor
τ	Torque, $N.m^2$
K_P	Proportional gain of PID controller
K_I	Integral gain of PID controller
K_D	Derivative gain of PID controller
J_{xx}	Moments of inertia along X direction, $kg.m^2$
J_{yy}	Moments of inertia along Y direction, $kg.m^2$
J_{zz}	Moments of inertia along Z direction, $kg.m^2$

Abbreviations

AFTC	Active Fault-Tolerant Control
ASAC	Automatic Supervisory Adaptive Control
BMI	Bilinear Matrix Inequality
BRL	Bounded Real Lemma
BSC	Backstepping Control
CG	Center of Gravity
DCM	Direct Cosine Matrix
DFC	Direct Flight Control
DTC	Damage Tolerant Control
EDF	Electric Ducted Fan
EPO	Expanded Poly Olefin
ESC	Electrical Speed Controller
FBL	Flybarless
FTC	Fault-Tolerant Control
FDD	Fault Detection and Diagnosis
FDI	Fault Detection and Identification
FTR	Fourier Transform Regression
GCS	Ground Control Station
GNC	Guidance, Navigation and Control
GS	Gain-Scheduled
GPS	Global Positioning System
IMU	Inertial Measurement Unit
ISR	Intelligence, Surveillance and Reconnaissance
JPAS	Joint Precision Airdrop System
Li-Po	Lithium Polymer
LMI	Linear Matrix Inequality
LOE	Loss of Effectiveness
LPV	Linear Parameter Varying
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
MPC	Model Predictive Control
MRAC	Model Reference Adaptive Control
NAS	National Airspace System
OCP	Open Control Platform
PD	Proportional-Derivative

PID	Proportional-Integral-Derivative
PFTC	Passive Fault-Tolerant Control
PWM	Pulse-Width-Modulation
RLS	Recursive Least Square
RPV	Remotely Piloted Vehicle
RPH	Remotely Piloted Helicopter
R/C	Remote Control
SEC	Software-Enabled Control
SI	System Identification
SMC	Sliding Model Control
SEAD	Suppression of Enemy Air Defence
UAV	Unmanned Aerial Vehicle
UAS	Unmanned Aerial Systems
UA	Unmanned Aircraft

Chapter 1

Introduction

1.1 Motivation

High level of safety is the aviation's first priority. The year 2010 was a disappointing period for the airline safety. In this year, the number of airline accidents and passenger fatalities increased while insurance claims have exceeded premiums, according to aviation consultants Ascend Worldwide [1]. As reported in [1], the fatal accident rate worsened in 2010 to one per 1.3 million flights from one per 1.5 million flights in 2009 which was considered as the safest year in aviation history. Apart from 2009, only 2006 and 2007 produced better (safer) accident rates. There was 28 fatal accidents in 2010, compared with 23 fatal accidents in 2009. The number of deaths including passengers and the crews, rose 13% in 2010 to 828 people compared with 731 people in 2009. This year was 4% worse than the past decade average although it was 27% improvement to comparing 1990's average of 1128 deaths. The estimated passenger fatality rate for 2010 was one per 3.8 million passengers carried comparing with one per 4.5 million passengers carried in 2009. However, as stated in ICAO 2013 safety report (ICAO Annex 13), in 2012, around 2.9 billion people flew around the world with significant decreasing of accident numbers (21%) and the significant decreasing in global accident rate (3.2 accidents per million departures)

involving scheduled commercial operations. Compared to 2011, the number of fatalities decreased by 10% making 2012 the safest year with regard to fatalities since 2004 [1].

The statistics shows the considerable number of annual air travels, and the safety index of air transportation. Although aviation safety has the highest safety rank among all means of transportation, but still considerable number of air crashes are reported every year due to four major factors known as SHELL model shown in Fig. 1.1. Regardless of the human and environmental factors, the main purpose of fault-tolerant flight control systems is to deal with machine (hardware/software) to safely handle the faults and failures in order to avoid the crash. Accumulation of faults in actuator, sensor, computer or controller, can cause a major failure and lead into a catastrophic crash.

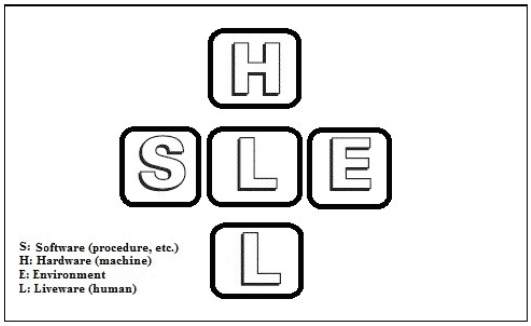

Figure 1.1: The SHELL model [6]

For example, the crash report of Air France flight 447 reveals that the airspeed sensor of the Airbus A330 (registered F-GZCP) shown in Fig. 1.2 was failed due to ice blockage followed by chain of machine and human faults. Since the airspeed data was cut off the autopilot, the autopilot was disengaged automatically after which the crew reacted incorrectly and ultimately led the aircraft to an aerodynamic stall. The plane crashed into the ocean and all 228 passengers and crew onboard were killed as reported in [2],[3] and [4].

The consideration of all possible faults during the flight with the capability of detection and diagnosis is the main motivation for fault-tolerant flight control systems.

Apart from the manned aviation the unmanned aircrafts are attracting more attention in civil and commercial applications in recent years. They can provide a low risk platform

Figure 1.2: The Airbus A330 crashed in June 2009

for various commercial applications with also a lower cost. As shown in [5] there are about 300 UAV commercial applications. This considerable number of UAV applications shows that quick actions should be taken for UAVs in term of flight control systems with more focus on fault-tolerant control systems, communication reliability improvement, proper integration of UAVs into the National Airspace System (NAS), etc.

1.2 Literature Review

In this section an overall review is provided based on the key issues of current researches in the field of fault-tolerant flight control of unmanned aerial vehicles.

The growing number of UAVs is considerable in the last decades. There are a wide variety of UAV shapes, sizes, configurations, and characteristics. UAVs are controlled from a remote location, or fly autonomously based on pre-programmed flight plans using more complex automated systems. In early years, UAVs were completely controlled by human operator from the ground, known as Remotely Piloted Vehicles (RPVs). However, the last decade has witnessed unprecedented interactions between technological developments in computing, control, and communications which made RPVs to fly autonomously in nowadays and more and more in the future. An autonomous control level trend for UAVs in the time scale is shown in Fig. 1.3 [7].

These developments led to the design and implementation of interacting dynamical systems such as networked unmanned multi-vehicle systems. Advances in sensor systems,

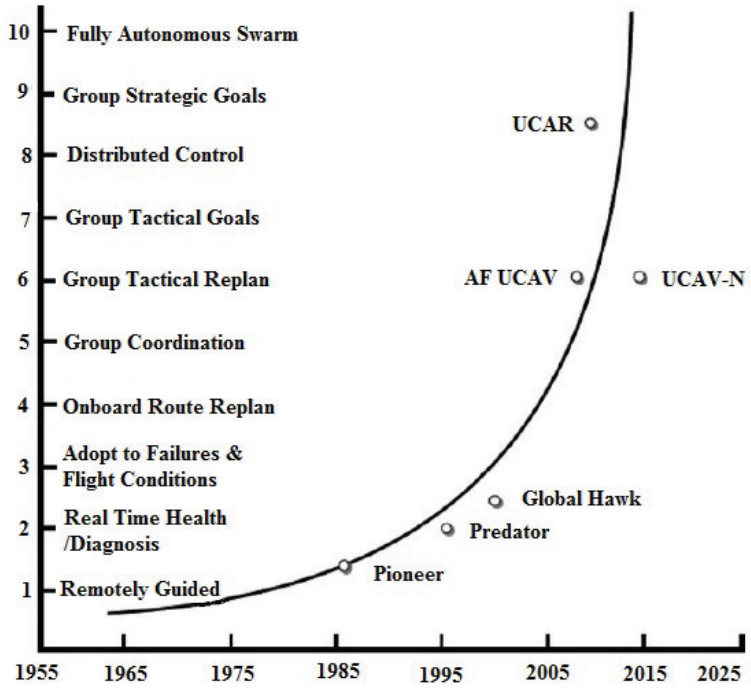

Figure 1.3: Autonomous control level trend [7]

onboard computational platforms, energy storage, and other enabling technologies have made it possible to build a huge variety of UAVs for a range of different mission scenarios from partial to fully autonomous control [8] and [9].

Also, UAVs are often preferred for missions that are too ”dull, dirty, or dangerous” for manned aircraft. In military applications, surveillance and reconnaissance, suppression of enemy air defence and high-value asset recovery scenarios can be considered. UAVs are also used in a small but growing number of civil applications, such as firefighting, surveillance of pipelines, telecommunication relay (hot spots), mining, crop dusting, power line monitoring, etc. In order to reduce the risk of human life, both emerging military and civilian application promote autonomous UAVs use. Many of the mission scenarios of interest, such as persistent surveillance, are inherently long-duration and require coordination of multiple cooperating UAVs in order to achieve the mission objectives. Thereby, the importance of UAV formation flight and cooperative control, currently is a challenging topic of research.

On the other hand, as of 2007, there are over 600 unmanned aerial systems of various sizes being manufactured by 250 companies in 42 nations, and this comprises solely commercial and government endeavours [10]. To realize the importance of fault-tolerant flight control of UAVs, it is worth mentioning that nearly half of the current-generation of unmanned surveillance aircrafts has been lost. This loss-rate is about 10 times worse than manned combat aircrafts. As stated in [11], a $250K police UAV crashed into Lake Conroe during an exercise over the lake. In [12], the report shows a commercial UAV that was being used to shoot a video for film industry crashed when it hits a building. Also a recent crash of Global Hawk UAV is reported in [13] due to an electrical malfunction. Also recently many quadrotor UAV (including hexacopter and octocopter UAV types) accidents are reported by the transportation authorities (see Fig. 1.4).

Although military and civilian researches and implementation results around the world have underscored the potential utility of UAVs, but yet most of their successes have occurred in a setting that allows a relatively large margin for error. Thereby, such sophisticated control systems should meet increased performance and safety requirements. In other words, the high demand of reliability, safety and fault-tolerance is required to avoid the consequences of a minor fault in a system component that can be catastrophic to any aircraft.

Figure 1.4: Octocopter UAV crash [14]

In order to investigate the existing UAV fault/failure researches, many different classifications can be conducted to address the existing problems in the field of FTFC. To give a more clear picture toward the review, the type of the UAV from its structural point of view (rotary-wing and fixed-wing) is the main classification parameter for this review.

1.2.1 Fault-Tolerant Flight Control of Rotary-Wing UAVs

There are different types of rotary-wing UAVs, such as conventional singlerotor, coaxial, tandem, quadrotor, etc. However, in most rotary-wing based researches, quadrotor and single-rotor UAVs are used as testbeds. The advantage of rotary-wing UAVs over fixed-wing UAVs, firstly, is the ability of vertical take-off and landing that brings the possibility of being used indoor or in the areas where there is no runway or airstrip. Also the capability of special operations such as low-speed payload carrying and deliverance is an important advantage of rotary-wing UAVs.

Quadrotor Helicopter UAV

Quadrotor UAV is a helicopter with different structure configurations such as $+$ shape, H shape or X shape, with four motors at the end of each rod. The idea of quadrotor dates back to 1920, but recently this configuration became a popular frame for researchers and R/C hobbyists thanks to its special configuration and associated dynamics which lead to relatively easy control of the helicopter with its actuators through PWM signals. Also, the new Li-Po battery technology along with powerful brushless motors makes the quadrotors easy to be built and used as testbeds. In Chapter 3 geometry and dynamics of the quadrotor helicopter is described.

Many control algorithms are applied on quadrotor UAVs mainly for fault-free case and in simulation environment. But, fault-tolerant flight control of quadrotor UAVs is gaining more and more attention in recent years.

In [15], MRAC together with combined/composite MRAC (CMRAC) are applied to

a quadrotor UAV in order to achieve higher performance as well as higher levels of robustness to the control of the quadrotor in presence of actuator faults. Results show that the CMRAC offers several benefits over the existing fixed-gain approach, particularly in the case of actuator failures. In an overall comparison, the CMRAC approach shows improved regulation and tracking performance over the regular MRAC approach. Also, the CMRAC allows for higher adaptive gains than the MRAC without exciting high frequency unmodelled dynamics. Such features contribute to the ability of CMRAC to respond quickly to a simulated actuator failure and return as closely as possible to nominal performance. A similar work in [15] shows the same approach based on adaptive control of quadrotor UAV in the presence of actuator uncertainties. In [16], the PD controllers are used for a mini quadrotor UAV along with FDI block based on Thau's observer developed for the nonlinear model of the quadrotor UAV.

At the Department of Mechanical and Industrial Engineering of Concordia University, with the financial support from NSERC (Natural Sciences and Engineering Research Council of Canada) through a Strategic Project Grant, a team of researchers has been working on fault detection and diagnosis and fault-tolerant control of quadrotor UAVs since 2007. In addition to the work that has been carried out for the multi-vehicle case, many fault-tolerant flight control (FTFC) strategies have been developed and applied to the quadrotor helicopter UAV system. The objective is to consider the actuator faults and to propose FTC methods to accommodate as much as possible the fault effects on the system performance. The proposed methods have been tested either in simulation, experiment or both frameworks where the experimental implementation has been carried out using Qball-X4 quadrotor helicopter UAV. Some of the developed approaches include: the Gain-Scheduled PID (GS-PID) [17], [18] and [19], Model Reference Adaptive Control (MRAC) [20], Sliding Mode Control (SMC) [21], Backstepping Control (BSC) [22] and flatness-based trajectory planning for actuator fault-tolerance capability improvement [23], etc.

Single Rotor Helicopter UAV

As mentioned before, helicopters have high manoeuvrability and hovering ability making them a suitable testbed for various missions. They are well suited to agile target tracking tasks, as well as inspection and monitoring tasks that require to maintain a position and to obtain detailed views. Furthermore, the vertical take-off and landing capabilities of helicopters is very desirable in many applications. Helicopters are inherently unstable and dynamically fast. Even with improved stability augmentation devices, a skilled, experienced pilot is required to control them during flight. Autonomous helicopter control is a challenging task involving a multivariable nonlinear open-loop unstable system with actuator saturation. Moreover, helicopters do not have the graceful degradation properties of fixed-wing aircrafts due to direct lift-to-power characteristics as well as the inherent instability. Thus, a failure in any part of the autonomous helicopter (actuators, sensors, control computer, etc.) can be catastrophic. Hence, fault-tolerant flight controllers are essential for such platforms.

Regarding conventional rotary UAV test-beds, the Yamaha-RMax UAV has attracted more attention due to the use of this UAV platform for their research work by Georgia Institute of Technology (Georgia Tech), thanks to its unique structure for different missions. The platform is designed and manufactured in 1983 by Yamaha Motor Company in Japan mainly for crop dusting based on the order received by the Ministry of Agriculture, Forestry and Fishery of Japan. Henceforth, about 2400 Yamaha Rmax helicopters are in service for spraying chemicals on rice farms. The RMax 50 helicopter is capable of carrying 20 kilogram of payload which makes this platform suitable for installation of required equipments for autonomous flight. In 1997 Georgia Tech purchased two Yamaha R-50 remotely piloted helicopters for their use in flight controls research under the Army/NASA sponsored Georgia Tech Center of Excellence in Rotorcraft Technology (CERT) program. The helicopter platform has been modified for autonomous flight purposes by Georgia Tech [24]. Flight control technologies, such as neural network and adaptive flight control, tested

Figure 1.5: Georgia Tech GTMax

on these testbeds have been transferred to Boeing for flight tests on the X-36 and Joint Direct Attack Munition (JDAM) programs as well as to NASA Ames and Marshal Space Flight Center (MSFC) for use in simulation studies. Georgia Tech then acquired a Yamaha RMax UAV with twice the payload of the R-50s, to be used in research and development programs. Georgia Tech has developed an open system UAV testbed based on this platform. This open system UAV testbed is referred to as the GTMax, as seen in Fig. 1.5, and includes four major elements. These are the basic Yamaha RMax helicopter, a modular avionics system, the SEC-OCP (developed jointly with Boeing Phantom Works) with baseline GNC components (software), and a set of simulation tools.

In [25], [26] and [27], FDI systems based on neural networks in the context of a reconfigurable flight control architecture for single-rotor helicopter UAVs are presented. The FDI system has been implemented and tested on GTMax autonomous helicopter. In this work, the diagnosis of actuator and sensor faults in autonomous helicopters is investigated and the design of a fault detection system is considered, using a model-based approach, with observer-based residual generation and the system is tested with real flight data.

1.2.2 Fault-Tolerant Flight Control of Fixed-Wing UAVs

Fixed-wing UAVs are useful testbeds for fault-tolerant flight control researches, thanks to their flight characteristics and stability factors. Fixed-wing UAVs are able to fly at higher speed and altitude and their flight range and endurance are more than rotary-wing UAVs.

However, fixed-wing UAVs are subject to some disadvantages in operation for research purposes, such as high speed take-off and landing, runway requirements, etc. Also, outdoor environment is subject to obstacles or disturbances such as cross wind and wind gust, that can highly degrade the performance of the UAVs. All of above-mentioned factors bring more operational challenges to fixed-wing UAVs comparing to rotary-wing UAVs for flight control and fault-tolerant flight control purposes.

Damage-Tolerant Flight Control of Fixed-Wing UAV

Structural damage of UAVs during combat operations is one of the scenarios that can be considered for FTC of fixed-wing UAVs. Rockwell Collins Inc. demonstrated a Damage-Tolerant Control (DTC) system for DARPAs Joint Unmanned Combat Aircraft system program using an unmanned F/A-18 subscale aircraft model with 60% of wing loss during flight [28]. In this interesting work, the autopilot formulation was shown to enable aerobatic maneuvers and automatic recovery from unusual attitudes. The model reference adaptive control module demonstrated the ability of damage-tolerant control to recover baseline control performance after losing 60% and 80% of one wing and primary flight control actuation (on the aileron, rudder, and/or elevator). The Automatic Supervisory Adaptive Control (ASAC) returned the aircraft with catastrophic wing damage to a re-trimmed and controllable flight within a few seconds, enabling the vehicle to complete its mission and perform an autonomous landing. The combination of these modules dramatically enhances the survivability of UAVs.

Two similar works are presented for damage-tolerant control by UAV Research Facility in Georgia Tech. In [29], the guidance and control of an airplane under severe structural damage is the main contribution of the works. It presents guidance and control algorithms for a Twinstar fixed-wing UAV under severe structural damage. The imposed damage includes 25% and 50% of asymmetric wing loss and the aircraft was able of successfully

Figure 1.6: Simulated wing damage of 60% in F/A-18 subscale UAV [28]

maintaining the stable autonomous flight and performing autonomous approach and landing. Key features of the presented algorithms are smart guidance techniques that ensure the aircraft maintains required flight speed to avoid stall and to use the available control surfaces to maintain autonomous flight. Closed-loop autonomous control is achieved using a PID type control technique with gain-scheduled switching technique based on measured airspeed. Adaptive control methods for the attitude control loop are also presented.

In the second DTC work of Georgia Tech, a real-time system identification is used for the same fixed-wing multi-engine aircraft testbed with structural damage [29]. Flight test data shows that the aircraft performs a special system identification manoeuvre with 25% left wing damage. The data was used to estimate a complete six-degree of freedom aircraft linear model with asymmetric stability derivatives using the recursive Fourier Transform Regression (FTR) method in frequency domain and the traditional least squares method in time domain. This experiment shows the feasibility of using the frequency domain FTR method for real-time parameter identification of damaged aircraft. The optimized multi-sine inputs were used for gathering rich system identification data for the twin engine aircraft with asymmetric lifting surface failure. Recursive implementation of the FTR method was used for parameter identification. Results indicate that a linear model with asymmetric stability derivatives included is able to approximately model the dynamics of

a transport category fixed-wing aircraft with asymmetric structural surface damage. The model identification data can be used for fault-tolerant control design.

Figure 1.7: Damage-tolerant control of DARPA subscale F/A-18 [30]

Figure 1.8: Georgia Tech Twinstar multi-engine testbed [31]

Furthermore, presented results indicate that various stability derivatives differ significantly from their nominal values when asymmetric damage is present. This justifies the use of online aerodynamic derivatives identification for health monitoring. The method of this work was analysed and has been implemented on the AFI FCS 20 autopilot onboard the Georgia Tech Twinstar UAV and the results indicate the feasibility of using the FTR method for real-time parameter identification and health monitoring on transport category fixed-wing aircraft. As mentioned previously, although this work is not directly related to FTC, but the results can be used to generate model-based FTC techniques.

1.3 General Problem Statement

From the existing works and above-mentioned motivation to this thesis work, the objective of this thesis is to develop and improve the controller that can stabilize the UAV when faults or failures occur in the actuators mainly in real application and flight test of UAVs. During recent years, various control techniques have been designed, developed and tested on UAVs, mainly in simulation environment. However, the results of a real application of FTC techniques may vary significantly from those in simulation. From this aspect, fixed-wing

UAVs are even more challenging to flight-test a fault-tolerant control techniques. Their high speed feature brings more difficulty in terms of launch and the recovery. The outdoor weather conditions such as wind, ambient air temperature trend and other outdoor environmental conditions along with other hardware factors such as GPS inaccuracy, compass calibration, telecommunication and telemetry delays are also additional difficulties.

1.4 Thesis Contribution

The contributions of this dissertation can be summarized in three categories as follows:

1. Design and development of application oriented methods:
   (a) Design and implementation of Gain-Scheduled Proportional-Integral-Derivative (GS-PID) control to fixed-wing UAV against rudder failures.
   (b) Design and implementation of GS-PID to quadrotor helicopter UAV for actuator loss-of-effectiveness (LOE) fault and propeller damage in height control as well as the trajectory tracking control.
   (c) Design and implementation of MRAC to quadrotor helicopter UAV for LOE faults.
   (d) Design and implementation of Control Allocation/Re-allocation technique to over-actuated helicopter UAV for propeller damage up to 100% of damage.
2. Theoretical design and development of flight control techniques
   (a) Design and development of Linear Parameter Varying (LPV) control technique for quadrotor helicopter UAV based on optimal state feedback and $H\infty$ approaches.
3. UAV testbeds development

(a) Development and system integration of fixed-wing and rotary-wing UAVs for NAVLAB of Concordia University.

1.5 Outline of the Thesis

This thesis is organized as follows:

Chapter 2 provides the literature review on recent works of fault-tolerant flight control of UAVs. The review is categorized based on two main types of UAVs: fixed-wing and rotary-wing UAVs.

Chapter 3 presents the GS-PID technique that is used as a FTC technique for a fixed-wing UAV known as HK Bixler. The platform is developed and integrated with ArduPilot APM 2.5 UAV autopilot. A brief introduction of this platform along with real flight test results are presented in the same chapter.

Chapter 4 compares different control techniques applied to the quadrotor helicopter UAV in real flight test. In this chapter, the GS-PID controller is used for the case of quadrotor actuator LOE fault, propeller damage and payload drop. The MRAC and Control Allocation and Re-allocation are also illustrated in this chapter for various fault tolerant control scenarios.

Chapter 5 introduces the Linear Parameter Varying (LPV) control technique that has been introduced as an alternative and more advanced gain-scheduling technique to switching control techniques such as GS-PID, in view of its advantage of stability guaranteed design in comparison with traditional gain-scheduling scheme. Two different LPV control approaches are used to generate the LPV controller for the flight control of Qball-X4 UAV.

Finally conclusion and future works are presented in Chapter 6.

1.6 Publications

- Journal Papers (Published/Submitted)

1. **Iman Sadeghzadeh**, Mahyar Abdolhosseini, and Youmin Zhang (2014), Payload Drop Application using an Unmanned Quadrotor Helicopter Based on Gain-Scheduled PID and Model Predictive Control, Unmanned Systems, 2(1), pp. 39-52.

2. **Iman Sadeghzadeh**, Ankit Mehta and Youmin Zhang, Fault-Tolerant Control of Quadrotor Helicopter Using Gain-Scheduled PID and Model Reference Adaptive Control, submitted to Journal of Unmanned System Technology (Under review).

3. Abbas Chamseddine, Didier Theilliol, **Iman Sadeghzadeh**, Youmin Zhang, and Philippe Weber (2014), Optimal Reliability Design for Over-actuated Systems Based on the MIT Rule: Application to an Octocopter Helicopter Testbed, Reliability Engineering & System Safety, 132:196-206.

- Conference Papers (Accepted/Presented)

1. **Iman Sadeghzadeh**, Abbas Chamseddine, Didier Theilliol and Youmin Zhang, Linear Parameter Varying Control Synthesis: State Feedback versus H∞ Technique with Application to Quadrotor UAV, in Proc. of the 2014 International Conference on Unmanned Aircraft Systems (ICUAS 2014), May 27-30, 2014, Orlando, FL, USA, pp. 1099-1104.

2. **Iman Sadeghzadeh** and Youmin Zhang, Actuator Fault-Tolerant Control Based on Gain-Scheduled PID with Application to Fixed-Wing Unmanned Aerial Vehicle, in Proc. of the International Conference on Control and Fault-Tolerant Systems (SysTol'13), October 9-11, 2013, Nice, France.

3. **Iman Sadeghzadeh**, Mahyar Abdolhosseini and Youmin Zhang, Payload Drop Application of Unmanned Quad-rotor Helicopter Using Gain Scheduled PID and Model

Predictive Control Techniques, Proc. of The 2012 International Conference on Intelligent Robotics and Applications (ICIRA12), October 3-5, 2012, Montreal, Quebec, Canada.

4. **Iman Sadeghzadeh**, Abbas Chamseddine, Youmin Zhang and Didier Theilliol, Control Allocation and Re-allocation for a Modified Quadrotor Helicopter against Actuator Faults, Proc. of the 8th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (Safeprocess12), August 29-31, 2012, Mexico City, Mexico.

5. Abbas Chamseddine, **Iman Sadeghzadeh**, Youmin Zhang, Didier Theilliol, and Ahmed Khelassi, Control Allocation for a Modified Quadrotor Helicopter Based on Reliability Analysis, in Proc. of AIAA Infotech@Aerospace 2012, Garden Grove, California, USA, June 19-21, 2012.

6. **Iman Sadeghzadeh**, Ankit Mehta, Abbas Chamseddine and Youmin Zhang, Active Fault-Tolerant Control of a Quadrotor UAV Based on Gain-Scheduled PID Control, in Proc. of the 25th IEEE Canadian Conference on Electrical and Computer Engineering, Montreal, QC, Canada, April 29 - May 2, 2012.

7. **Iman Sadeghzadeh**, Ankit Mehta, Youmin Zhang, and Camille-Alain Rabbath, Fault-Tolerant Trajectory Tracking Control of Quadrotor Helicopter Using Gain-Scheduled PID and Model Reference Adaptive Control, in Annual Conference of the Prognostics and Health Management Society 2011 (PHM2011), Sept. 25-29, 2011, Montreal, Quebec, Canada (Nominated as PHM2011 Best Paper).

8. **Iman Sadeghzadeh**, Ankit Mehta, and Youmin Zhang, Fault Tolerant Control of a Quadrotor Umanned Helicopter using Model Reference Adaptive Control, Presented at The Third Symposium on Small Unmanned Aerial Vehicle Technologies and Applications (SUAVTA'11) in ASME/IEEE International Conference on Mechatronic

and Embedded Systems and Applications (MESA2011), Washington, DC, USA, August 28-31, 2011.

9. Narendra Gollu, **Iman Sadeghzadeh**, and Youmin Zhang, Guidance Navigation and Control Algorithms for an Autonomous Landing of a Quad-Rotor on a Moving Rover, in AUVSI (Association for Unmanned Vehicle Systems International) Unmanned Systems North America 2011, Washington, DC, USA, August 16-19, 2011.

10. **Iman Sadeghzadeh**, Ankit Mehta, and Youmin Zhang, Model Reference Adaptive Fault/Damage Tolerant Control of a Quadrotor Unmanned Aerial Vehicle (UAV), in AIAA Guidance, Navigation, and Control Conference, Portland, Oregon, USA, August 8-11, 2011.

11. **Iman Sadeghzadeh** and Youmin Zhang, Review on Fault Tolerant Control for Unmanned Aerial Vehicles (UAVs), in AIAA InfoTech@Aerospace 2011: Unleashing Unmanned Systems, St. Louis, Missouri, USA, March 29-31, 2011 (AIAA-2011-1472).

Chapter 2

Background, Preliminaries and Definitions

2.1 Definition of Fault and Failure

Since the studied systems in this thesis are said to be fault-tolerant, it is worth to give a terminological distinction between a fault and a failure by the following definitions [32]:

Fault: "A fault is an unpermitted deviation of at least one characteristic property (feature) of the system from the acceptable, usual, standard condition" [32]. Based on this definition, any abnormal behavior of the system can be corresponded to a fault. Fault may or may not affect the overall functions of the system but may eventually lead to a failure. A fault may be small or hidden, and therefore undetectable.

Failure: "A failure is a permanent interruption of a system's ability to perform a required function under specified operating conditions" [32]. In other words, an actuator is declared failed when it can no longer be used in a controlled manner. Occurrence of one fault or a chain of faults can lead to termination of the functioning components in the system. In an aircraft, actuators are mainly used to deflect primary flight control surfaces such as ailerons, elevators, and rudders as well as secondary flight control surfaces. These

control surfaces may become ineffective and float at the zero moment position, stuck at any arbitrary intermediate position or reach and stay at the saturation position. Some types of faults and failures are shown in Fig. 2.1 [33].

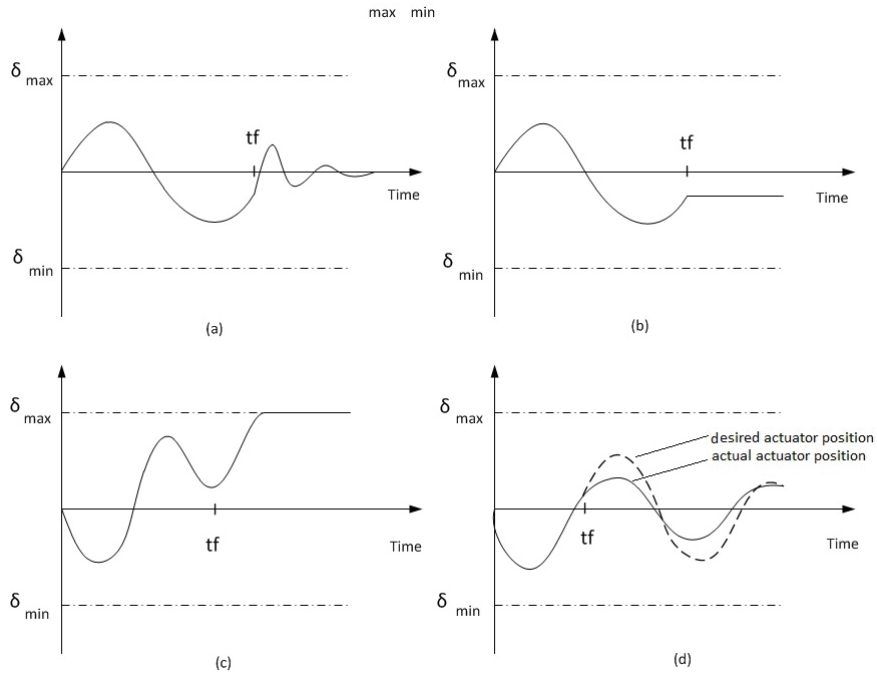

Figure 2.1: Several types of actuator failures: (a) floating around trim point; (b) locked-in-place; (c) hard-over; and (d) loss of effectiveness (actuator fault occurring after t_F) [33]

Mechanical and structural failures and damages may also happen to control surfaces and actuators due to object impact to control surfaces. There are many sources of possible irreversible damage to the aircraft that may be classified as structural failures. They correspond to the scenarios where a piece of the aircraft is missing, such as an aileron, a rudder, an elevator, or part of a wing [31] and [34]. In this thesis, an actuator fault corresponds to any abnormal behavior. This includes actuator damage, stuck or loss of effectiveness.

2.2 Types of Fault-Tolerant Control Systems

As stated in [35], a fault-tolerant control system is capable of controlling the system with satisfactory performance even if one or several faults, or more critically, one or several

failures occur in the system. Fault-tolerant control systems can be grouped into two main families: passive fault-tolerant control systems and active fault-tolerant control systems.

2.2.1 Passive Fault-Tolerant Control Systems

In a Passive Fault-Tolerant Control System (PFTCS), deviations of the plant parameters from their true values or deviations of the actuators from their expected position may be efficiently compensated by a fixed robust feedback controller [36], [37] and [38]. In fact, PFTCSs are designed to tolerate certain class of component faults without the need of on-line fault information. For such systems, the anticipated level of faults and the controller's structure and its parameters are fixed during the normal and faulty case operations. However, if these deviations become excessively large and exceed the robustness limits, the controlled plant can lead into instability. Also, if deviations occur at the sensor side, inevitable deviations from the reference command signals will happen.

Although PFTCSs are simple to implement, but it is difficult to handle the fault with magnitude larger than the robustness limits of the controller. A block diagram showing PFTCS structure is shown in Fig. 2.2.

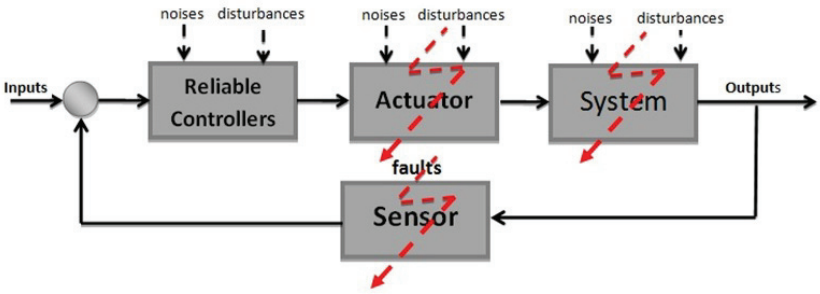

Figure 2.2: Passive fault-tolerant control block diagram

Active fault-tolerant control architecture is an alternative way to overcome some of the drawbacks of PFTCS and can be used in order to achieve extended fault-tolerance capability. The corresponding control system is referred as to Active Fault-Tolerant Control System (AFTCS).

2.2.2 Active Fault-Tolerant Control Systems

Active (reconfigurable) fault-tolerant control system is capable of detecting faults in the system (which are more difficult to detect than failures) and is able to adequately compensate for failures (which is more difficult than to only accommodate faults). An active fault-tolerant controller usually contains a separate module: an FDD (Fault Detection and Diagnosis) system that monitors the health of the system. The FDD system provides information to a supervisory module about the magnitude of the fault, failure or damage. Once the fault is detected, located and identified (the type and magnitude of the fault), appropriate actions can be taken using the fault-tolerant controller to handle the fault in the system. In aircraft example, the supervisory module may decide to reconfigure the flight controllers, the guidance system, and the navigation system based on the FTC type used in the system. There are also two types of FDD systems, namely passive FDD and active FDD systems. For the same example, the passive FDD systems "wait" until a fault or failure occurs, but in the active FDD systems, health-check flying maneuvers or test signals injections via actuator's commands can be used for health monitoring of the system. Different types of Kalman filter, for instance, can be used as a FDD block. However the focus of this thesis is mainly on fault-tolerant controllers but not FDD block.

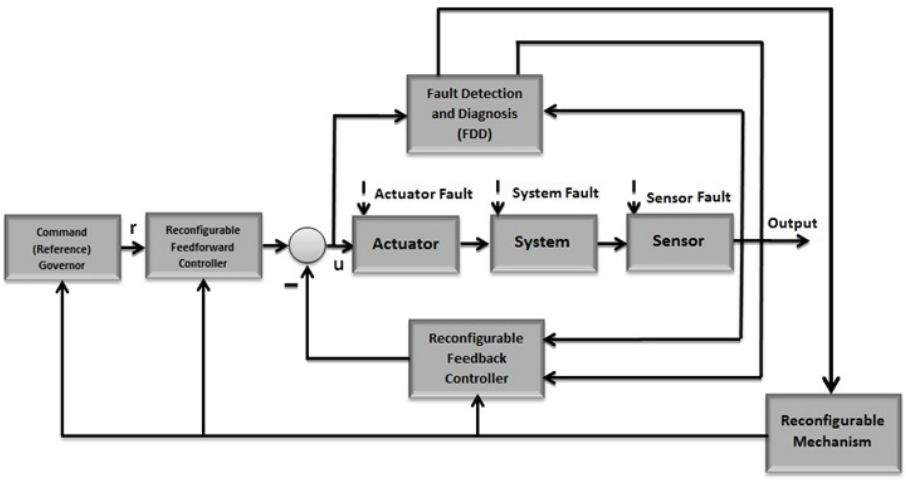

Figure 2.3: Active fault-tolerant control scheme [35]

2.3 Proportional-Integral-Derivative (PID) Control

Proportional-Integral-Derivative (PID) control is a widely-used closed-loop feedback control technique. PID controllers are the standard tool in current industrial automation experience thanks to their simplicity and flexibility making the PID controller capable of being used in many situations. Some of the advantages of PID controllers are: simplicity, easy to use, and robust against insensitive changes of plant parameters and disturbances. Many simple control problems can be handled very well by using PID control. The incremental or velocity algorithm is typically implemented in automation applications. In order to achieve the best performance, the gains of the PID controller must be tuned based on the nature of the plant. The tuning process is the adjustment of the control gains to the optimal values for the desired response. Although tuning of Single-Input-Single-Output (SISO) system PID gains are not very complicated thanks to several existing methods, the fine-tuning of PID gains for Multi-Input-Multi-Output (MIMO) systems is still challenging. There are some current existing tuning techniques for PID controllers. The Ziegler-Nichols (ZN) method is the most well-known method for tuning of PID controllers. The algorithm of the PID controller for discrete version is shown as following:

$$u(t_k) = u(t_{k-1}) + \Delta u(t_k) \tag{2.1}$$

where

$$\begin{aligned} \Delta u(t_k) = {} & K_p[e(t_k) - e(t_{k-1})] + \frac{K_p h}{T_i} e(t_k) \\ & + \frac{K_p T_d}{h}[e_f(t_k) - 2e_f(t_{k-1}) + e_f(t_{k-2})] \end{aligned} \tag{2.2}$$

where T_i and T_d denote the time constants of the integral and derivative terms respectively. u is the control variable, h is discretization rate, e is the tracking error defined as the difference between the reference signal and output of the system. K_p, $K_i = \frac{K_p}{T_i}$, and $K_d = K_p T_d$ are

controller gains associated with proportional (P), integral (I), and derivative (D) action, respectively.

2.3.1 Gain-Scheduled PID

In many control systems the dynamics of the plant changes with the operating conditions of the plant. If the plant's dynamic parameters varies without re-tuning the controller, the control system may get unstable or may become more sluggish. This problem with variable process dynamics can be solved using GS technique (GS-PID for instance). If the change of dynamics in a system/process with the operating condition is known, then it is possible to change the parameters of the controller by monitoring the operating conditions of the process. This approach is called gain-scheduling (GS) because the scheme was originally used to accommodate changes in process gain only. Generally GS is a proper way to compensate the variation of parameters or known nonlinearities of the plant but, the proper parameter tuning based on dynamic variations of the plant and the proper switching between those parameters plays a vital role in general control and stability of the plant. In most of Fly-By-Wire (FBW) aircrafts, the GS technique is used to control the aircraft under variation of altitude, airspeed, CG, weight, slat/flap position or the combination of these parameters. This allows pilot to operate the aircraft in a safer manner by protecting the flight envelope based on aforementioned parameter variations. However, it is worth mentioning that switching inherently can cause instability of the aircraft. In other words, the switching among different set of PID gains which are tuned for different portions of the flight envelope can cause the autopilot to send undesirable signal to the control surfaces that can be catastrophic. A basic scheme of GS technique is shown in Fig. 2.4.

In this thesis the GS-PID is used to control both fixed-wing and rotary-wing UAVs for different flight conditions including fault/failure in one or more actuators.

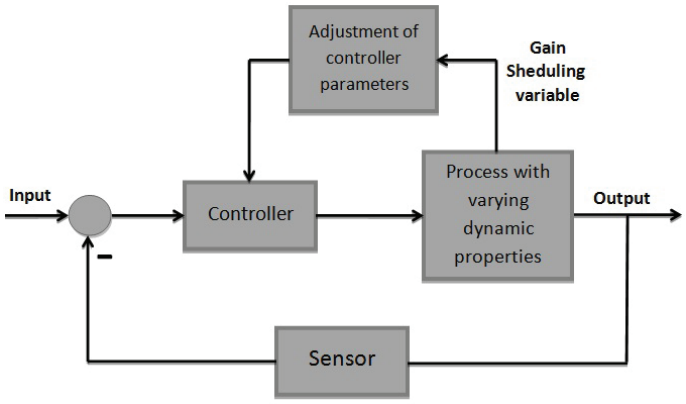

Figure 2.4: Control system for a process having varying dynamic properties. The GS variable expresses or represents the dynamic properties of the process

2.3.2 Auto-Tuning PID

As mentioned previously, PID controllers are the backbone of industrial systems especially in aerospace industry and the problem of their parameters determination is of great importance in the professional control domain. As described in [39], 97% of controllers are PID type but only 32% of the loops show "excellent" or "good" performance. Thereby, the auto-tuning techniques become very useful for online tuning of the PID controller gains in the case of any changes in the plant's dynamics or fault occurrence. It should be mentioned that a sufficiently skilled human with sufficient process knowledge, data and time available can outperform any auto-tuner in any situation. Auto-tuning can provide a tremendous improvement in setting up and maintaining control systems provided that it is viewed as an aid to human skill, not a substitute for it. This improvement can take place in several directions. Some of the existing PID tuning methods such as: Ziegler-Nicols (ZN), loop shaping method, analytical tuning methods, pole placement, dominant pole placement, step and frequency response may be used as the basic algorithms for auto-tuning techniques. However, some of these algorithms require data post-processing in order to find the best gains of the PID controller in auto-tuning manner.

There are different auto-tuning techniques for PID controllers which amongst relay-based self-oscillating technique is one of the most widely used techniques. In fact, the

auto-tuning algorithm makes the controller capable of measuring the system response to determine effective settings for Proportional, Integral, and Derivative control parameters. In other words, auto-tuning techniques attempt to observe and figure out the nature of what the controller is driving, then back-calculate tuning parameters from that.

Based on a relay element in the loop the auto-tuning feature reverts the controller to ON/OFF function and oscillates the system. The relay can be placed in the control loop as shown in Fig. 2.5.

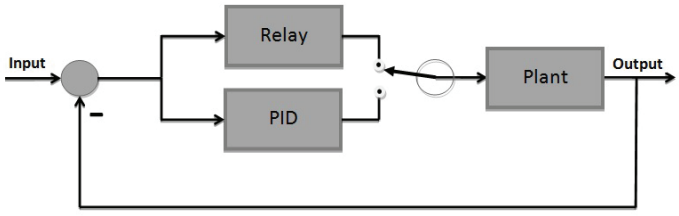

Figure 2.5: Structure of a relay-based auto-tuning PID controller

As shown in Fig. 2.5 the process begins with the steady-state for both steady input and output when the switch is closed for relay. Then, the output should be stepped in one direction by some distance L. When the input crosses a threshold (trigger) line, the output changes to the other direction by distance L. Based on the distance and amplitude of peaks in relation to the output changes, the auto-tuning module will determine the parameters for different types of plants. However, since in real application the data is noisy the proper set-up of the trigger line is challenging. Fig. 2.6 shows the input-output of a relay system.

If one define K_u as following:

$$K_u = \frac{2L}{\pi a} \tag{2.3}$$

then the following modified ZN equations can be used to determine the controller parameters.

For PI control type:

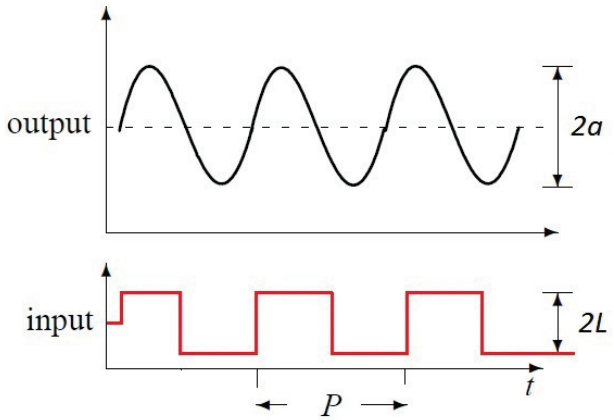

Figure 2.6: Plant oscillating under relay feedback with disabled PID

$$K_p = 0.4K_u \quad , \quad K_i = \frac{0.48K_u}{P} \quad and \quad K_d = 0 \tag{2.4}$$

and for PID controller:

$$K_p = 0.6K_u \quad , \quad K_i = \frac{1.2K_u}{P} \quad and \quad K_d = 0.075K_u.P \tag{2.5}$$

During the oscillation of the plant up to 180 degree of phase shifts between the output and the input occurs based on the oscillation frequency and the plant response. This phase shift is shown in Fig. 2.7

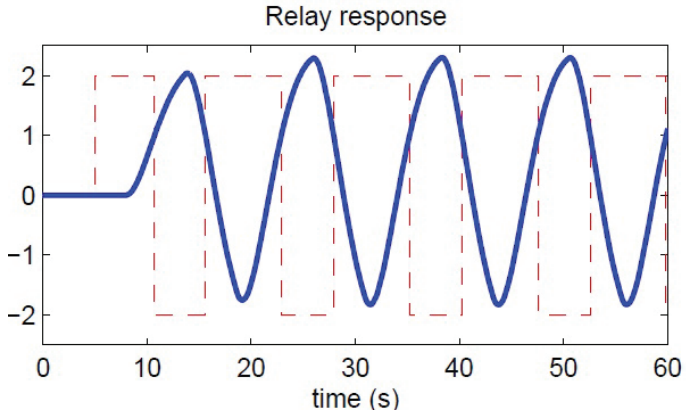

Figure 2.7: Phase shift between input and output in oscillation mode

As illustrated above, the relay-based technique is a very useful method for auto-tuning the PID controller. This technique forces the plant to oscillate in one step and by measuring the oscillation parameters the ZN-based algorithm can be used to calculate the PID gains. However the physical oscillation of some plants and parameters are not proper for such a technique.

2.4 Model Reference Adaptive Control

In recent years, Model Reference Adaptive Control (MRAC) has gained considerable attention and specially in the aerospace community. Research in adaptive control started in the early 1950s by designing autopilot for high-performance aircraft. Adaptive control technique was designed to deal with complex systems that have unpredictable parameter deviations and uncertainties. The main objective was to maintain consistent performance of a system in the presence of uncertainty and variations in plant parameters. Unlike robust control which has been designed to deal with disturbances, quickly varying parameters, and unmodeled dynamics, adaptive control deals with uncertainties in constant or slow-varying parameters.

In mid 1990s, many successful flight tests have been conducted and demonstrated the significant achievements in the theoretical framework, particularly with respect to stability and robustness. As previously described in section 1.2.2, in 2010 Rockwell Collins Inc. demonstrated a very interesting robustness of adaptive control technique on a subscale F/A-18 fighter jet UAV by injecting more than 60% of damage to its right wing and later by injecting damage in the tail section. In all of damage scenarios the airplane was capable of performing auto-landing based on adaptive-based control technique. These and many other recent researches show the importance of such control algorithm in aerospace systems.

In MRAC systems, the plant has a known structure but the parameters are unknown and the reference model specifies the desired response to the external command as shown

in Fig. 2.8. The controller is parameterized and provides tracking whiles the adaptation mechanism is used to adjust parameters in the control law. In MRAC controller, there are some parameters that need to be optimized based on the system response for achieving the desired output. The output of the reference model is considered as ideal output of the system and the adjustment mechanism adjusts (forces) the controller parameters to match the process output with the ideal reference model output.

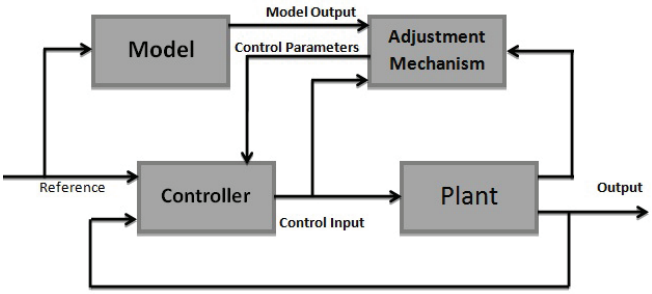

Figure 2.8: Model Reference Adaptive Control (MRAC) structure

The MRAC systems can be divided into two categories: Direct and Indirect methods. In indirect method the uncertain plant parameters are estimated and the controller "redesigned" online based on parameter estimation. In other words, the parameters of the plant should be estimated which brings the capability of computing the controller parameters with the aim of convergence of the estimated parameters to their true unknown values and the tracking error is forced to zero without considering the accuracy of parameter estimation. In direct mode however, there is no need for plant parameter estimation but only the controller parameters (gains).

There are different approaches to MRAC such as [40] and [41]:

- The MIT rule
- MRAC design based on Lyapunov stability theory;
- Hyperstability and passivity theory
- The error model

- Augmented error
- Model-following MRAC
- Modified MRAC (M-MRAC)
- Combined/Composite MRAC (C-MRAC)

Although the MIT rule is the original approach to MRAC but it can not guarantee the stability of the closed-loop system. However, the MRAC based on the Lyapunov stability theory can be considered as an alternative approach. As described in [42], in order to generate MRAC based on Lyapunov stability theory the differential equation of the error, which is the difference between the output of the model and the output of the system, should be derived which contains the adjustable parameters. In order to reduce the error to zero, a Lyapunov function and an adjustable mechanism should be established. The Lyapunov derivative function dV/dt is usually only negative semidefinite. Therefore, to define the parameter convergence, the persistent excitation and uniform observability should be conducted on the system and the reference signal. The Lyapunov stability theorem for time-invariant systems can be established as following: If there exists a function $V : R^n \rightarrow R$ being positive definite and its derivative:

$$\frac{dV}{dt} = \frac{\partial V^T}{dx}\frac{\partial x}{dt} = \frac{\partial V^T}{dx} f(x) = -W(x) \tag{2.6}$$

is negative semidefinite, then the solution $x(t) = 0$ to

$$\frac{dx}{dt} = f(x), f(0) = 0 \tag{2.7}$$

is stable. If dV/dt is negative definite the solution will be asymptotically stable. V denotes the Lyapunov function for the system. If:

$$\frac{dV}{dt} < 0 \quad \text{and} \quad V(x) \rightarrow \infty \quad \text{when} \quad ||x|| \rightarrow \infty \tag{2.8}$$

the solution is globally asymptotically stable. Therefore, the following procedure was realized:

The plant model:

$$\ddot{y_p} + a_1\dot{y_p} + a_2 y_p = bu \tag{2.9}$$

Reference model:

$$\ddot{y_r} + a_{1_r}\dot{y_r} + a_{2_r} y_r = b_r u_c \tag{2.10}$$

The control law:

$$u = \theta_1 u_c - \theta_2 y_p \tag{2.11}$$

and the error:

$$e = y_p - y_r \tag{2.12}$$

where e , y_p, y_r, u and u_c represent the error, process output, reference output, process input and controller input, respectively. Also θ_1 and θ_1 are adaptive parameter inside the controller and a_1, a_2 and b are unknown parameters. Then, the error dynamics is represented by:

$$\dot{e} = \dot{y_p} - \dot{y_r} = 1/a_1[bu - \ddot{y_p} - a_2 y_p] - 1/a_1 r - [bu - \ddot{y_r} - a_{2_r} y_r] \tag{2.13}$$

Substituting $y_r = y_p - e$ and $y_r = y_p - e$ from Eq. (2.13), following equation can be derived:

$$\dot{e} = \frac{1}{a_1} bu - \frac{1}{a_1}\ddot{y_p} - \frac{a_0}{a_1} y_p - \frac{1}{a_{r_1}} b_r u_c + \frac{1}{a_{1_r}}\ddot{y_p} + \frac{1}{a_{1_r}}\ddot{e} + \frac{a_{2_r}}{a_{1_r}} y_p - \frac{a_{2_r}}{a_{1_r}} e \tag{2.14}$$

Replacing $u = \theta_1 u_c - \theta_2 y_p$ in the above equation and placing the error terms in the left side of the equation, the following equation is obtained:

$$\frac{1}{a_{1_r}}\ddot{e} + \dot{e} + \frac{a_{2_r}}{a_{1_r}} e = \frac{1}{a_{1_r}} b_r \theta 1 u_c - \frac{1}{a_{1_r}} b_r u_c - \frac{1}{a_{1_r}} b_r \theta_2 y_p + \frac{a_{2_r}}{a_{1_r}} y_p - \frac{1}{a_{1_r}}\ddot{y_p} + \frac{1}{a_{1_r}}\ddot{y_p} \tag{2.15}$$

The Reference Model is equal to the Process Model if no fault occurs ($a_1 = a_{1_r}$, $a_2 = a_{2_r}$,

and $b = b_r$), then:

$$\frac{1}{a_{1_r}}\ddot{e} + \dot{e} + \frac{a_{2_r}}{a_{1_r}}e = \frac{1}{a_{1_r}}b_r\theta 1 u_c - \frac{1}{a_{1_r}}b_r u_c - \frac{1}{a_{1_r}}b_r\theta_2 y_p + \frac{a_{2_r}}{a_{1_r}}y_p - \frac{1}{a_{1_r}}\ddot{y_p} + \frac{1}{a_{1_r}}\ddot{y_p} \tag{2.16}$$

$$\frac{1}{a_{1_r}}\ddot{e} + \dot{e} + \frac{a_{2_r}}{a_{1_r}}e = \frac{1}{a_{1_r}}(b_r\theta_1 - b_r)u_c - \frac{1}{a_{1_r}}(b_r\theta_2)y_p \tag{2.17}$$

$$\frac{1}{a_{1_r}}\frac{d^2e}{dt} + \frac{de}{dt} + \frac{a_{2_r}}{a_{1_r}}e = \frac{1}{a_{1_r}}(b_r\theta_1 - b_r)u_c - \frac{1}{a_{1_r}}(b_r\theta_2)y_p \tag{2.18}$$

$$\frac{de}{dt} = -\frac{1}{a_{1_r}}\frac{d^2e}{dt} - \frac{a_{2_r}}{a_{1_r}}e + \frac{1}{a_{1_r}}(b_r\theta_1 - b_r)u_c - \frac{1}{a_{1_r}}(b_r\theta_2)y_p \tag{2.19}$$

The proposed Lyapunov function is quadratic in tracking error and controller parameter estimation error since it is expected that the adaptation mechanism will drive both types of errors to zero. From the equation error dynamics (see Eq. (2.19)) the proposed Lyapunov function is:

$$V(e, \theta_1, \theta_2) = -\frac{1}{2}(a_{1_r}e^2 + \frac{1}{\gamma b_r}(b_r\theta_1 - b_r)^2 + \frac{1}{\gamma b_r}(b_r\theta_2)^2) \tag{2.20}$$

where b_r, γ and $a_{1_r} > 0$. Equation (2.20) will be zero when the error is zero and the controller parameters are equal to the desired values. The above Lyapunov function is valid if the derivative of this function is negative. Thus, the derivative of Eq. (2.20) is:

$$\dot{V} = a_{1_r}e\frac{de}{dt} + \frac{1}{\gamma}(b_r\theta_1 - b_r)\frac{d\theta_1}{dt} + \frac{1}{\gamma}(b_r\theta_2)\frac{d\theta_2}{dt} \tag{2.21}$$

Substituting Eq. (2.19) in the above equation, and rearranging the similar terms, Eq. (2.22)

is obtained.

$$\begin{aligned}\dot{V} =& a_{1_r} e(-\frac{1}{a_{1_r}}\frac{d^2 e}{dt}^2 - \frac{a_{2_r}}{a_{1_r}} e + \frac{1}{a_{1_r}}(b_r\theta_1 - b_r)u_c - \frac{1}{a_{1_r}}(b_r\theta_2)y_p) \\ &+ \frac{1}{\gamma}(b_r\theta_1 - b_r)\frac{d\theta_1}{dt} + \frac{1}{\gamma}(b_r\theta_2)\frac{d\theta_2}{dt}\end{aligned} \tag{2.22}$$

$$\dot{V} = -e\frac{d^2 e}{dt^2} - a_{2_r} e^2 + (b_r\theta_1 - b_r)u_c e - (b_r\theta_2)y_p e + \frac{1}{\gamma}(b_r\theta_1 - b_r)\frac{d\theta_1}{dt} + \frac{1}{\gamma}(b_r\theta_2)\frac{d\theta_2}{dt} \tag{2.23}$$

$$\dot{V} = -e\frac{d^2 e}{dt^2} - a_{2_r} e^2 + (b_r\theta_1 - b_r)u_c e + \frac{1}{\gamma}(b_r\theta_1 - b_r)\frac{d\theta_1}{dt} - (b_r\theta_2)y_p e - \frac{1}{\gamma}(b_r\theta_2)\frac{d\theta_2}{dt} \tag{2.24}$$

$$\dot{V} = -e\frac{d^2 e}{dt^2} - a_{2_r} e^2 + \frac{1}{\gamma}(b_r\theta_1 - b_r)(\frac{d\theta_1}{dt} + \gamma u_c e) + \frac{1}{\gamma}(b_r\theta_2)(\frac{d\theta_2}{dt} - \gamma y_p e) \tag{2.25}$$

Therefore, the adaptation parameters are selected to be updated as:

$$\frac{d\theta_1}{dt} = -\gamma u_c e \tag{2.26}$$

$$\frac{d\theta_2}{dt} = -\gamma y_p e \tag{2.27}$$

therefore

$$\dot{V} = -e\frac{d^2 e}{dt^2} - a_{2_r} e^2 \tag{2.28}$$

It can be seen that Eq. (2.28) is negative semidefinite which implies $V(t) \leq V(0)$. This ensures that e, θ_1 and θ_2 are bounded. Since $a_{1_r} > 0$, $a_{2_r} > 0$ and u_c is bounded then y_r is bounded and therefore $y_p = e + y_r$ is bounded as well. From the boundedness and

convergence set theorem it can be concluded that the error e will go to zero [42].

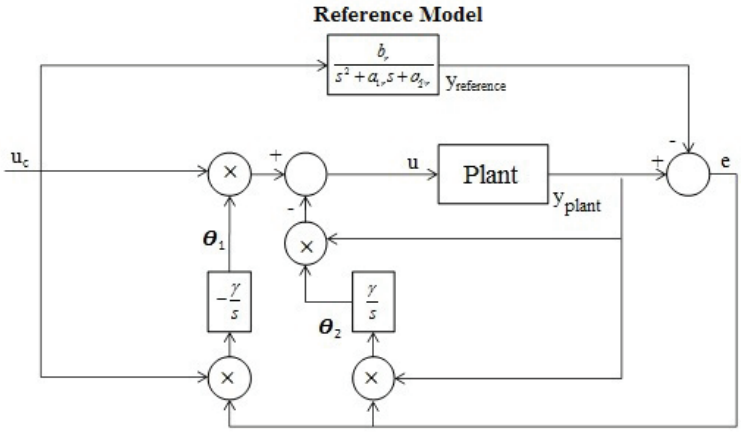

Figure 2.9: MRAC structure based on Lyapunov theory

2.5 Control Allocation and Re-allocation

Control Allocation and Re-allocation (CA/RA) is one of the most useful control algorithms in aerospace industry. Many recently developed fly-by-wire flight control system designs are based CA/RA technique for different flight modes based on aircraft configurations. Most of modern aircrafts are so-called over-actuated systems due to several control surfaces such as: inboard/outboard ailerons, multifunctional spoilers, ground spoilers and high lift surfaces (slats and flaps) that make them to have redundant actuators for control of all six-degree of freedom in flight. These redundancies bring more attention to flight control designers to the use of CA/RA technique. A conceptual representation of CA/RA block diagram is illustrated in Fig. 2.10.

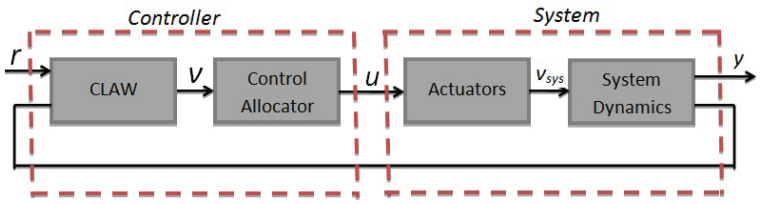

Figure 2.10: Control allocation block diagram

From the reconfigurable flight control point of view, CA/RA technique can be used to handle the fault without modifying the baseline controller. As shown in Fig. 2.11, the CA/RA block is located between the controller and the actuators. For fault-tolerance purpose, the system uses the control allocation block when the system operates in fault-free conditions and the control reallocation block at the time of fault occurrence in the actuators. Different constrained optimization based algorithms such as least-squares, linear programming and quadratic programming can be implemented for CA/RA purpose. The pseudo-inverse, fixed-point, direct control allocation and weighted least square are just few algorithms to name.

As described in [43], the control system is made up of a controller, specifying the total control effect, v should be produced and a control allocator, which distributes this control demand among the individual actuators, u is produced. In this system, actuators

generate a total control effect v_{sys}, which determines the system behavior. If the control allocation is successful, then $v_{sys} = v$.

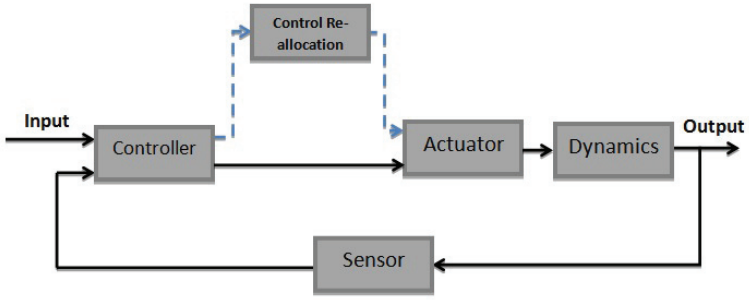

Figure 2.11: Control allocation/re-allocation structure

As described in [44], the CA/RA technique in recent years is in the center of attention of aerospace control, marine vessel control and UAV control. In this thesis, the CA/RA technique is used for fault-tolerant flight control of a quadrotor helicopter UAV against actuator faults/propeller damages.

2.6 Linear Parameter Varying Control

As described in previous section, many control systems are based on gain scheduling systems. As stated in [46], "Machines that walk, swim, or fly are gain-scheduled". Generally, the GS technique contains various methods that try to solve the challenging problem of nonlinear control. For this purpose the local linear system theory is mainly used to obtain a non-local controller. However, the local descriptions of the nonlinear system cannot capture the non-local behavior. Hence, the GS methods may not result in a controller that meets the specifications of the feedback system. As shown in Fig. 2.12, most of GS controllers have been designed and tuned within a specific zone that the Linear Time-Invariant (LTI) model is closed-loop stable.

The Linear Parameter Varying (LPV) control as an alternative control method can be used for the controller synthesis in a systematic way of obtaining the nonlinear controller

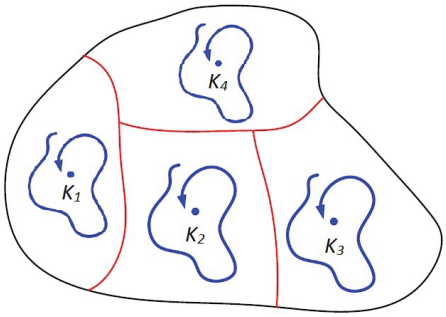

Figure 2.12: Gain-scheduling operating range

for nonlinear systems in a linear-like fashion. If the standard linear system with state-space description is as following:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{2.29}$$

then the LPV system can be defined as:

$$\begin{aligned} \dot{x}(t) &= A(\rho(t))x + B(\rho(t))u \\ y(t) &= C(\rho(t))x \end{aligned} \tag{2.30}$$

This LPV system can be considered as standard linear system with respect to input, state and output with time-varying system matrices and the parameter vector as following:

$$\rho(t) = (\rho_1(t) + \rho_2(t) + ... + \rho_n(t)) \tag{2.31}$$

With such a control synthesis method, the controller parameters get interpolated based on the plant's selected varying parameter which is not known in advance but can be measured in real-time. One or more parameters can be selected as the varying parameters based on the plant's nature. This kind of gain-scheduling technique guarantees the stability and the performance of the system. In Chapter 5, two methods are used to synthesize the LPV controller and have been applied to a quadrotor UAV as the testbed.

Chapter 3

Fault-Tolerant Flight Control of a Fixed-Wing UAV

A fixed-wing aircraft has both advantages and drawbacks in comparison with rotary-wing aircrafts. A properly designed fixed-wing aircraft tends to be stable in the air which brings more forgiving flying characteristics in the face of both piloting and technical errors. Thanks to their capability of gliding with no power, engine failure or power loss, fixed-wing aircraft has less effect on the aircraft to compare to singlerotor or quadrotor helicopter. Fixed-wing aircrafts are able also to carry heavier payloads for longer distances on less power. The range and endurance capabilities of fixed-wing UAVs make them suitable platforms for the missions that require less power consumption in long range flights such as fire-fighting, pipeline monitoring, aerial monitoring and mapping. A small UAV with about 1 kg may be able to fly for about twenty minutes at a speed of 100 km/h depending on size and power system of the UAV. This brings the necessity of FTC system integration to such UAVs. Any fault or failure in an airplane's hardware, software, telecommunication or the telemetry system can cause the loss of airplane. That is the reason why the role of fault-tolerant controller is inevitable for fixed-wing UAVs.

Several research works have been conducted toward the fixed-wing UAV mainly focused on the control and navigation. In [47], a PID controller is used to control the airspeed and altitude of a fixed-wing UAV. The control structure provides a low level control necessarily for mission planning and execution. In [48] a hierarchical approach for real-time motion planning is used for a small fixed-wing UAV. The proposed approach divides the trajectory generation into four tasks: waypoint path planning, dynamic trajectory smoothing, trajectory tracking, and low-level autopilot compensation. In [49], an optimization-based tuning for PID controllers is presented in order to minimize the requirement of in-flight "tuning" and substantially reduce the risks and costs involved in flight tests. Also, a design of a robust PID controller scheme that combines the deadbeat response, robust control, and model reduction techniques is described in [50] to enhance the performance and robustness of PID controller. The design scheme is applied for the pitch control of a fixed-wing UAV.

This chapter focuses on the application of fault-tolerant control to a fixed-wing UAV based on widely used PID control technique. The PID controller is used to control all six variables separately, which are Roll, Pitch, Yaw (Euler angles), X, Y and Z (Navigation) for both fault-free and faulty scenarios. For fault cases, the Gain-Scheduled PID (GS-PID) control strategy is used to handle the hard-over failure injected to UAV with split rudders. Different sets of pre-tuned PID gains are designed for different parts of flight envelope and fault cases. The results are presented at the end of the chapter.

3.1 Bixler UAV testbed and ArduPilot Mega 2.5 Autopilot

3.1.1 The HK Bixler UAV Testbed

For this work, a dynamically stable airframe known as "HK Bixler" with pusher propeller configuration and brushless DC motor is considered as the testbed. The pusher motor located in the back of the fuselage making the plane safer to operate. Also it makes the autopilot integration easier and eliminates the turbulent flow caused by the propeller. This

helps the barometric sensor and the pitot-static tube to face a laminar airflow with less measured airspeed error. Such a configuration will also be significantly helpful for installing cameras and other payloads for the purpose of UAV applications such as forest fire monitoring and detection with necessary remote sensing capability. The above mentioned testbed is shown in Fig. 3.1.

Figure 3.1: The HK Bixler UAV testbed with equipped avionics

The dihedral angle in winglet form of the wing brings more lateral stability to the airframe. This makes the UAV to somehow compensate the unwanted lateral movements by wind gusts and makes the plane more stable. The specifications of the HK Bixler airframe are presented as follows:

- Wingspan: 1400 mm
- Fuselage Length: 925 mm
- Wing Area: 26 dm^2
- Wing Loading: 25 g/dm^2
- Flying Weight: 650 g
- Motor: 2620-1900 kv
- Battery: 2600 mAh 3 cell Li-Po

- Electrical Speed Controller (ESC): 20 *A*

It should also be mentioned that the original prototype of this airframe has no flaps and the rudder and it was in one piece (that is, only single rudder control surface available originally as a RC model airplane). Since the actuator redundancy brings more controllability to the system and such a hardware redundancy is a necessary condition for achieving fault-tolerant control, two additional flaps are added to the roots of the wing in the leading edge section and the rudder is splitted into two pieces, i.e. upper and lower rudder surfaces, as shown in Fig. 3.2. These modifications in UAV airframe help the airplane to be suitable for the main objective of this research and development work.

Figure 3.2: Splitted rudder for added/enhanced hardware redundancy

As shown in Fig. 3.2, the rudder is splitted into the upper rudder and the lower rudder parts. Each section of rudder can be controlled by separate servo motor and autopilot channel and associated controller. In this work, the upper rudder is designed for hard-over failure injection. Flaps are also designed to generate more lift in take-off and more drag and higher glide angle with low approaching speed in landing portion of the flight. Such a configuration also helps to reduce the stall speed of the airplane in slow flights.

3.1.2 ArduPilot Mega 2.5 Autopilot

To implement flight control law, in particular fault-tolerant control law for the developed fixed-wing UAV testbed, low cost and reliable onboard microcomputer/microcontroller is

necessary as a controller hardware and software environment. For such a purpose, an open source and low cost autopilot system, ArduPilot Mega 2.5 (APM 2.5) is selected and integrated to the HK Bixler UAV testbed. Such an autopilot has the capability of performing programmed GPS missions for waypoints following. It consists of a very high resolution Inertial Measurement Unit (IMU) as well as the barometric and dynamic airspeed sensor. The APM 2.5 board along with its sensors and components is shown in Fig. 3.3.

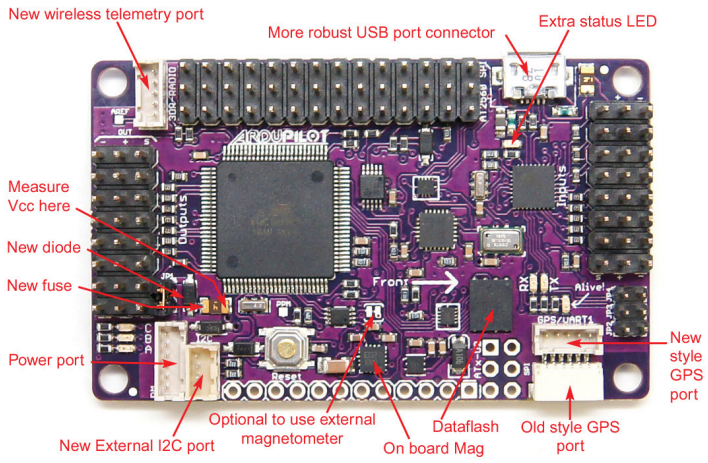

Figure 3.3: The ArduPilot Mega 2.5 autopilot [51]

The APM 2.5 autopilot is also capable of communicating with mission planner embedded in the Ground Control Station (GCS) computer via 915 MHz or 433 MHz radio telemetry module as well as 2.4 GHz Xbee. The GCS unit collects all necessary data receiving from UAV via telemetry module during flight for further analyses. This data contains the UAV coordinates, heading, air speed, ground speed, vertical speed (climb and sink rate), roll angle, pitch angle, yaw angle and many other useful information. In addition, operator is able to define waypoints on the map, and even to re-shape the flight pattern in flight by changing the coordinates of waypoints using GCS. The screen shot of Mission Planner (MP) with map and some of the data transmitted to GCS computer by radio telemetry module in circle flight pattern is shown in Fig. 3.4.

The integration of APM 2.5 autopilot with other components is illustrated in the block diagram of Fig. 3.5. As it can be seen from Fig. 3.5, the APM 2.5 autopilot is located

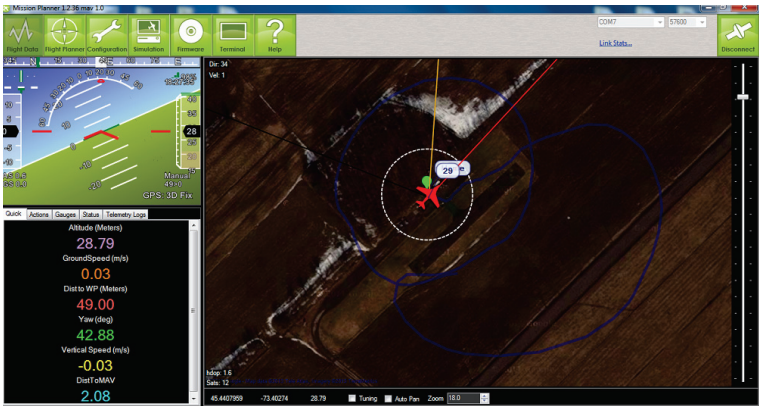

Figure 3.4: Mission planner in ground control station with a circle flight track

between the radio control receiver and servos motors which are connected to actuators. Also the Pulse-Width-Modulation (PWM) output signals can be used for ESC to power the electric motor for electrically powered UAVs such as HK Bixler or other platforms. This type of autopilot integration allows the pilot to switch between different flight modes.

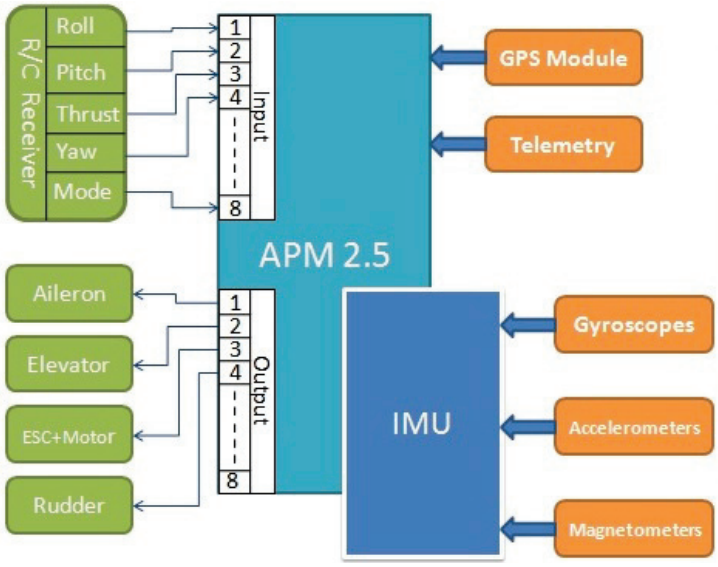

Figure 3.5: Block diagram of the APM 2.5 integration

It makes also the pilot capable of injecting fault for FTC test scenarios during the flight. Other sensors such as IMU, airspeed sensor, radio telemetry and external magnetometer are also plugged into APM 2.5 autopilot as feedback data inputs. These sensors, in cooperation with radio telemetry module, allow the operator to track the UAV on the live

Google map.

There are two ways of data logging for APM 2.5. Data either can be collected by the embedded onboard 16 MB flash memories or by the help of telemetry module and GCS computer which allows more data to be collected. The radio telemetry module along with UBLOX LEA-6 GPS unit and airspeed sensor (with pitot-static tube) is shown in Fig. 3.6.

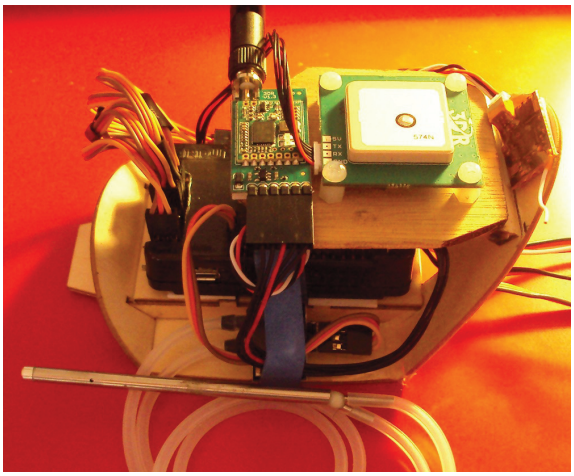

Figure 3.6: The custom made APM 2.5 avionics rack equipped with GPS, pitot-static tube and radio telemetry module

3.2 Gain-Scheduled PID Controller for Bixler UAV

As explained previously PID and GS-PID control strategies are widely used in aerospace and many industrial applications. The algorithm of the PID controller in time domain is shown as following:

$$u(t) = K_p e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de(t)}{dt} \tag{3.1}$$

where u is the control variable and e is the error defined as the difference between reference signal and the real output y of the system. K_P, K_I, and K_D are controller gains associated with proportional (P), integral (I), and derivative (D) actions, respectively [42].

Generally, if the change of dynamics in a system/process with the operating condition

is known, then it is possible to change the parameters of the controller by monitoring the operating conditions of the process. In this work, for GS-PID controller, a set of pre-tuned gains are applied to the controllers under fault-free and hard-over rudder failure of the HK Bixler UAV.

Description of experimental testing results is presented in next section.

3.3 Experimental Results

In this section the results of the above discussed controllers (PID and GS-PID) which are applied to the HK Bixler fixed-wing UAV for FTC purpose are illustrated. To demonstrate effectiveness of the control algorithms, a number of experiments are set up. As mentioned previously, every channel of the HK Bixler UAV is controlled by separate PID controller as shown in Fig. 3.7.

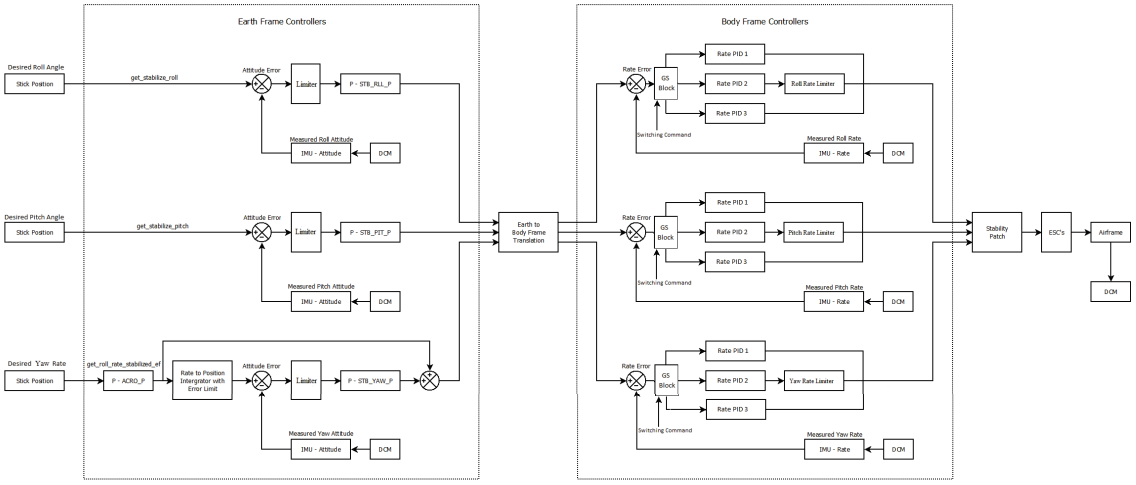

Figure 3.7: Illustrative GS-PID controller block diagram for HK Bixler UAV

In the first set of experiments, focus is on the single PID controller to take over control of the UAV for all phases of flight (except take-off and landing) especially for fault injection to upper rudder by about 30 degrees of deflection. As shown in Fig. 3.8, a rectangular flight trajectory is designed using four waypoints. However, more than four waypoints for

designing a trajectory is preferable to avoid steep turn in autonomous waypoint trajectory tracking.

In the first experiment, the rudder hard-over failure is injected to upper rudder in downwind portion of flight pattern. As shown in Fig. 3.9, the PID controllers of roll and yaw channels along with navigation PID controllers (X, Y and Z channels) were able to bring the UAV back to its trajectory in not a very short time.

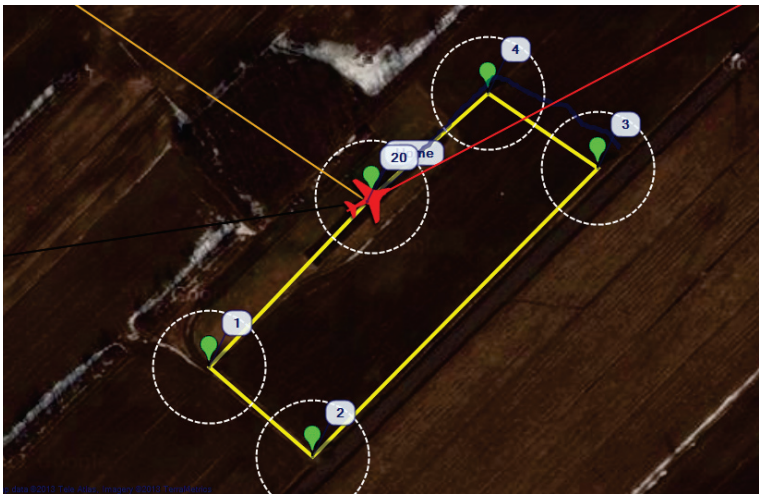

Figure 3.8: Assigned flight trajectory using four waypoints

Figure 3.9: Fault injection in downwind leg using single PID controller

In the second set of the experiments, the GS-PID is used for the same flight scenario to test the fault-tolerant capability of the GS-PID in comparison with the single PID. For

this objective, a set of pre-tuned PID gains are used to bring back and keep the UAV in its assigned track or to immediately recover it from its deviation of the assigned track at the time of fault injection. Since the magnitude of the fault as well as its occurrence time is assumed to be provided in real-time by either an fault detection and diagnosis module or a special smart sensor installed on the airframe (if it is available), the PID controller of yaw and roll channels are pre-tuned and scheduled with different set of gains for the time of the fault. It should be mentioned that separated single PID controllers are used for UAV navigation controllers.

Figure 3.10: Fault injection in downwind leg using GS-PID controller

At the time of fault injection in the upper rudder, the UAV abruptly changes the heading followed by a steep turn to left. Unlike the PID controller, the GS-PID controller controls the heading change and the roll motion by switching to a new set of gains which have been adjusted in advance for roll and yaw channels. After switching to the new set of PID gains, the deflections of ailerons and the lower rudder surface in opposite direction makes the UAV to deviate less from its desired trajectory. Such a control action causes an unwanted angle between the flight track of the UAV and its heading (sideslip angle) which is normal due to rudder fault. On the other hand, the sideslip angle increases the drag mainly on the fuselage of the airplane which causes the descent of the UAV over the time. However the PID controller in altitude increases the throttle to compensate for it. The

result for the GS-PID controller as well as the altitude hold control block diagram taken form [51] are shown in Fig. 3.10 and Fig. 3.11 respectively.

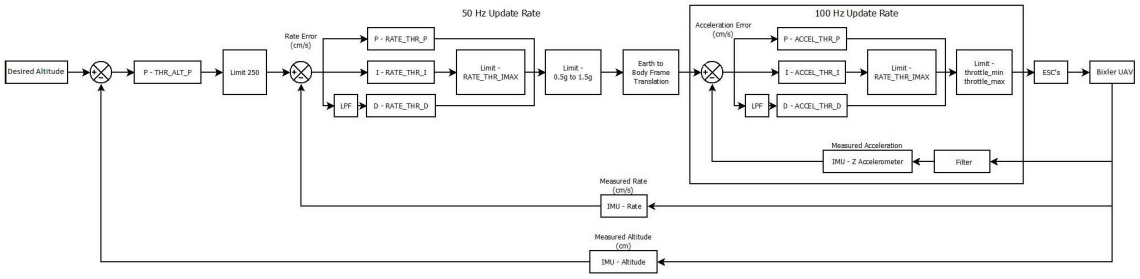

Figure 3.11: HK Bixler PID altitude control block diagram

Compared with Fig. 3.9 for the single PID controller case, GS-PID controller shows much improved response of compensation from overall deviation of designed trajectory. Such a flight testing result demonstrates the capability and effectiveness of such a simple fault-tolerant control strategy based on GS-PID control technique together with setup of hardware redundancy for achieving fault-tolerance of this UAV testbed and other more general, larger or smaller fixed-wing type of airplanes and UAVs. Such a work is important and useful in the sense to demonstrate such a design philosophy and technique towards practical fault-tolerant control systems design in general by combining hardware and analytical redundancies in a coherent way, not alone.

As mentioned previously, at the time of the fault in split rudder, a sudden change in heading occurs which forces the UAV to deviate from its assigned trajectory. However, the roll controller acts in opposite direction in order to bring the UAV back to the flight trajectory. That is the reason why the roll angle appears in sensor readings. Readings of all three sensors, i.e. roll, pitch and yaw angles are shown in Fig. 3.12 and Fig. 3.13 respectively.

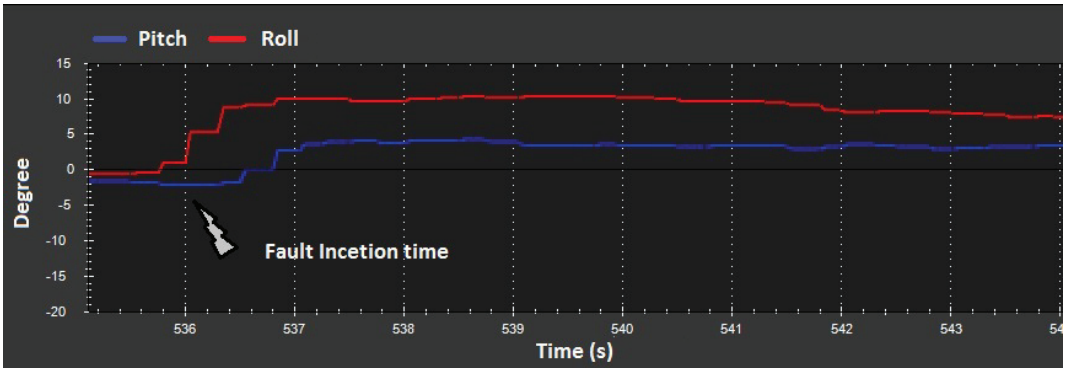

Figure 3.12: Fault effect on roll and pitch sensors

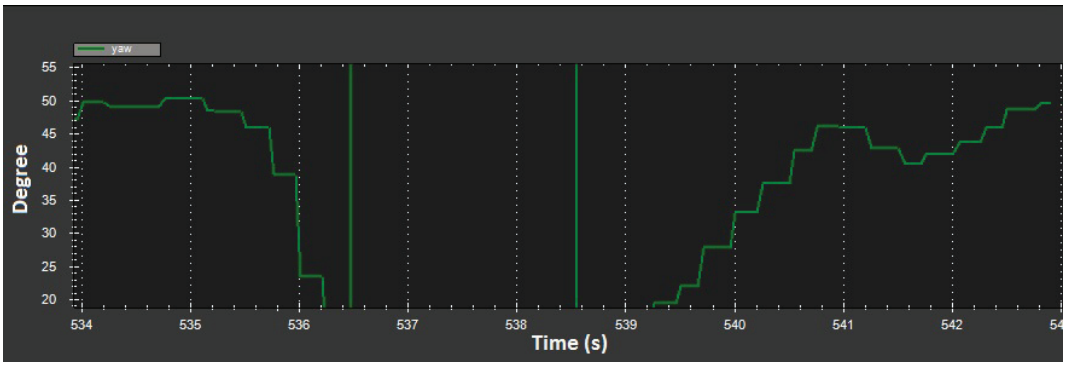

Figure 3.13: Fault effect on yaw sensor

Chapter 4

Fault-Tolerant Flight Control of a Quadrotor Helicopter UAV

In this chapter several fault-tolerant control techniques applied to the quadrotor helicopter UAV known as Qball-X4 are presented. Model Reference Adaptive Control (MRAC), Gain-Scheduled PID (GS-PID) and finally Control Allocation and Re-allocation (CA/RA) are the control algorithms used as fault-tolerant controllers for the Qball-X4.

Before presenting the applied control techniques and their results, it is worth to give a brief introduction of Qball-X4 quadrotor UAV system and its components in the next section.

4.1 Description and Dynamics of the Qball-X4 Quadrotor UAV Testbed

The Qball-X4 UAV (Fig. 4.1) available at the Network Autonomous Vehicle (NAV) Lab in the Department of Mechanical and Industrial Engineering of Concordia University is a prototype quadrotor testbed, designed and manufactured by Quanser [52].

The quadrotor UAV is enclosed within a protective carbon fiber ball-shape cage

Figure 4.1: The Qball-X4 quadrotor UAV [52]

(therefore a name of Qball-X4) to ensure safe operation. It uses four 10×4.7 inch propellers and AC brushless motors and speed controllers. It is equipped with the Quanser Embedded Control Module (QECM), which is comprised of a Quanser HiQ aero data acquisition card and a QuaRC-powered Gumstix embedded computer. The Quanser HiQ provides high-resolution accelerometer, gyroscope, and magnetometer IMU sensors as well as servo outputs to drive four motors. The onboard Gumstix computer runs QuaRC (Quansers real-time control software), which allows to rapidly develop and deploy controllers designed in MATLAB/Simulink environment to real-time control of the Qball-X4. The controllers run onboard the vehicle itself and runtime sensors measurement, data logging and parameter tuning are supported between the host computer (ground station) and the target vehicle. The block diagram of the entire UAV system is illustrated in Fig. 4.2. It is composed of three main parts. The first part represents the ESCs + the motors + the propellers in a set of four. The input to this part is $u = [u_1\ u_2\ u_3\ u_4]^T$ which are PWM signals. The output is the thrust vector $T = [T_1\ T_2\ T_3\ T_4]^T$ generated by four individually-controlled motor-driven propellers. The second part is the geometry that relates the generated thrusts to the applied lift and torques to the system. This geometry corresponds to the position and orientation of the propellers with respect to the center of mass of the Qball-X4. The third part is the dynamics that relate the applied lift and torques to the position (P), velocity (V) and acceleration (A) of the Qball-X4 [53].

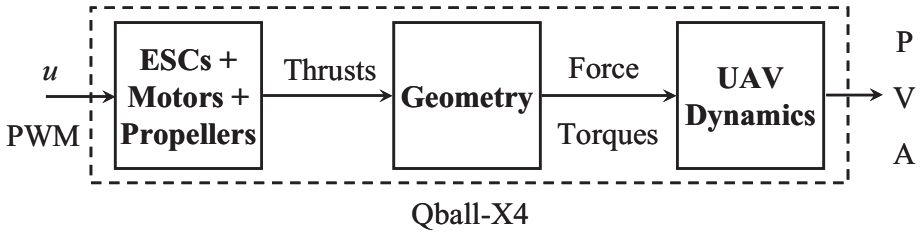

Figure 4.2: The UAV system block diagram

The subsequent sections describe the corresponding mathematical model for each of the blocks in Fig. 4.2.

4.1.1 OptiTrack Motion Tracking System for Localization

A set of twenty four V100:R2 cameras which offer integrated image capture, processing, and motion tracking in a compact package constitute the OptiTrack's optical motion tracking system. The capability of customizing cameras with user-changeable M12 lenses, and OptiTrack's exclusive Filter Switcher technology has let V100 cameras deliver one of the world's premier optical tracking value propositions. Each V100:R2 camera is capable of capturing fast moving objects with its global shutter imager at 100 FPS capture speed. By maximizing its 640×480 Video Graphic Array (VGA) resolution through advanced image processing algorithms, the V100:R2 can also track markers down to sub-centimeter movements with maintainable accuracy [54].

A variety of V100:R2 settings are customized with any of OptiTrack's software applications such as the one employed in this study, i.e. Tracking Tool, for greater control over what cameras capture and what information they report to the personal computer set up as the ground station. Available settings include: image processing type, frame rate, exposure, threshold, illumination, filter switching, and status LED control. OptiTrack's application software, named Tracking Tool, interfaces with MATLAB via specific blocks inside MATLAB/Simulink within the library of QuaRC. Fig. 4.3 illustrates one of the six cameras employed constituting the system of OptiTrack.

Figure 4.3: V100:R2 camera used for Optitrack positioning system

4.1.2 ESCs, Motors and Propellers

The motors of the Qball-X4 are out-runner brushless motors. The generated thrust T_i of the i^{th} motor is related to the i^{th} PWM input u_i by a first-order linear transfer function:

$$T_i = K\frac{\omega}{s+\omega}u_i \tag{4.1}$$

where $i = 1,...,4$ and K is a positive gain and ω is the motor bandwidth. K and ω are theoretically the same for the four motors but this may not be the case in practice. It should be noted that $u_i = 0$ corresponds to zero thrust and $u_i = 0.05$ corresponds to the maximal thrust that can be generated by the i^{th} motor.

4.1.3 Geometry

A schematic representation of the Qball-X4 is given in Fig. 4.4. The motors and propellers are configured in such a way that the back and front (1 and 2) motors spin clockwise and the left and right (3 and 4) spin counterclockwise. Each motor is located at a distance L from the center of mass o and when spinning, a motor produces a torque τ_i which is in the opposite direction of that of the motor as shown in Fig. 4.4. The origin of the body-fixed frame is the system's center of mass o with the x-axis pointing from back to front and the

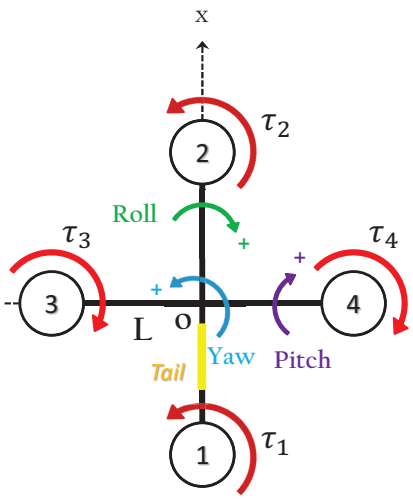

Figure 4.4: Schematic representation of the Qball-X4

y-axis pointing from right to left. The thrust T_i generated by the i^{th} propeller is always pointing upward in the z-direction in parallel to the motors rotation axis. The thrusts T_i and the torques τ_i result in a lift in the z-direction (body-fixed frame) and torques about the x, y and z axis. The relations between the lift/torques and the thrusts are:

$$\begin{aligned} u_z &= T_1 + T_2 + T_3 + T_4 \\ u_\theta &= L(T_1 - T_2) \\ u_\phi &= L(T_3 - T_4) \\ u_\psi &= \tau_1 + \tau_2 - \tau_3 - \tau_4 \end{aligned} \tag{4.2}$$

The torque τ_i produced by the i^{th} motor is directly related to the thrust T_i via the relation of $\tau_i = K_\psi T_i$ with K_ψ as a constant. In addition, by setting $T_i = Ku_i$ from Eq. (4.1), the relations (4.2) can be written in a compact matrix form as:

$$\begin{bmatrix} u_z \\ u_\theta \\ u_\phi \\ u_\psi \end{bmatrix} = \begin{bmatrix} K & K & K & K \\ KL & -KL & 0 & 0 \\ 0 & 0 & KL & -KL \\ KK_\psi & KK_\psi & -KK_\psi & -KK_\psi \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \tag{4.3}$$

where u_z is the total lift generated by the four propellers and applied to the quadrotor UAV in the z-direction (body-fixed frame). The parameters u_θ, u_ϕ, and u_ψ are respectively the applied torques in θ, ϕ, and ψ directions (see Fig. 4.4). As mentioned before, L is the distance from the center of mass to each motor.

4.1.4 UAV Dynamics

A commonly employed quadrotor UAV model is [53]:

$$\begin{aligned} m\ddot{x} &= u_z(\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \; ; & J_1\ddot{\theta} &= u_\theta \\ m\ddot{y} &= u_z(\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) \; ; & J_2\ddot{\phi} &= u_\phi \\ m\ddot{z} &= u_z(\cos\phi \cos\theta) - mg \; ; & J_3\ddot{\psi} &= u_\psi \end{aligned} \tag{4.4}$$

where x, y and z are the coordinates of the quadrotor UAV center of mass in the earth-fixed frame. Also θ, ϕ, and ψ are the pitch, roll and yaw Euler angles respectively and m is the mass. If we fix the yaw angle to zero ($\psi = 0$) and consider the roll and pitch angles very small then a simplified linear model can be obtained in hovering conditions ($u_z = mg$ in the x and y directions). Therefore the linear model that can be used for control design is given

by:

$$\begin{aligned} \ddot{x} &= \theta g \; ; & J_1 \ddot{\theta} &= u_\theta \\ \ddot{y} &= -\phi g \; ; & J_2 \ddot{\phi} &= u_\phi \\ \ddot{z} &= u_z/m - g \; ; & J_3 \ddot{\psi} &= u_\psi \end{aligned} \tag{4.5}$$

4.2 Model Reference Adaptive Fault/Damage Tolerant Control of the Qball-X4 Quadrotor UAV

An adaptive controller is a controller with adjustable parameters and a mechanism for adjusting the parameters. Among all adaptive control techniques the Model Reference Adaptive Systems (MRAS) are important adaptive systems which were originally derived for deterministic continuous-time systems [42]. Extension to discrete-time systems and systems with stochastic disturbances has also been carried out. As shown in Fig. 2.8 the system has an ordinary feedback loop composed of the plant and the controller and another feedback loop that changes the controller parameters. The parameters are changed on the basis of feedback from the error, which is the difference between the output of the system and the output of the reference model. The ordinary feedback loop is called inner loop, and the parameter adjustment loop is called the outer loop. In other words, as stated in section 2.4, MRAC is concerned with forcing the dynamic response of the controlled system to asymptotically approach that of a reference system represented by a reference model, despite parametric uncertainties in the plant. The mechanism for adjusting the parameters in model reference adaptive system can be obtained by using a gradient method or by applying stability theory [42]. As mentioned in section 2.4 two major subcategories of MRAC are those of indirect methods, in which the uncertain plant parameters are estimated and the controller redesigned online based on the estimated parameters, and direct methods, in which the tracking error is forced to zero regardless of parameter estimation accuracy

(though under certain conditions related to the level of excitation in the command signal, the adaptive laws often can converge to the proper values). There are different approaches to MRAC such as [40], [60] and [41]. In [60] a comparison is made for different MRAC approaches applied to the NASA Generic Transport Model (GTM) which is a subscale model of Boeing 757 fixed-wing UAV. Also in [61] and [62] a multivariable MRAC is applied to the same test-bed. Another example for MRAC with application to civil aviation application for a real scale Bonanza fly-by-wire airplane is presented in [63]. In [64], a modified MIT rule is designed for a second order system and results have been shown in simulation environment. A similar work for a second order system is described in [65]. A MRAC fuzzy controller design and application to an automatic gauge control system is presented in [66]. For fast parameter adaptation of MRAC based a optimal control technique is used in [67]. In [68] a new direct model reference fuzzy adaptive control of SISO continuous-time nonlinear systems based on an adaptive Takagi-Sugeno (TS) fuzzy system is developed. Finally, a combination of feedback linearization and with indirect MRAC technique is presented in [69].

The MRAC technique used in this chapter is based on the MIT rule. The MIT rule is the original approach to MRAC. The name is derived for the fact that it was developed at the Instrumentation Laboratory at Massachusetts Institute of Technology (MIT) for aerospace applications. The MIT rule is used to control the height in hovering as well as the trajectory tracking of the Qball-X4.

To explain the MIT rule, consider the following second order system [20]:

$$\ddot{y} = -a_1\dot{y} - a_2 y + u \tag{4.6}$$

where a_1, a_2 and are unknown plant parameters, and $\dot{y}$ and y are available for measurement. The reference model to be matched by the closed-loop plant is given by:

$$\ddot{y}_m = -2\dot{y}_m - y_m + r \tag{4.7}$$

where r is the reference command. The control input u can be defined as follows:

$$u = \theta_1 \dot{y} + \theta_2 y + r \tag{4.8}$$

In order to obtain a perfect model following one can choose θ_1 and θ_2 as:

$$\theta_1 = a_1 - 2,\ \theta_2 = a_2 - 1 \tag{4.9}$$

The objective of the MIT rule is to minimize the cost function by adjusting the parameters θ_1 and θ_2. This cost function can be chosen for example as follows:

$$J = \frac{1}{2}e^2 \tag{4.10}$$

where e is the error showing the difference between the system output and the reference model output i.e. $e = y - y_m$. It is reasonable to adjust the parameters in the direction of the negative gradient of J:

$$\frac{d\theta_i}{dt} = -\gamma \frac{\partial J}{\partial \theta_i} = -\gamma e \frac{\partial e}{\partial \theta_i} = -\gamma e \frac{\partial y}{\partial \theta_i} \tag{4.11}$$

where $\gamma > 0$ is the adaptation rate and $\partial e / \partial \theta_i$ is known as sensitivity derivative of the system and is evaluated under the assumption that θ_i varies slowly. After calculation one can obtain:

$$\frac{dy}{d\theta_1} = \dot{y}/(p^2 + (a_1 - \theta_1)p + (a_2 - \theta_2)) \tag{4.12}$$

and

$$\frac{dy}{d\theta_2} = y/(p^2 + (a_1 - \theta_1)p + (a_2 - \theta_2)) \tag{4.13}$$

where the p is the differential operator. By using the matching Eq. (4.9) it is possible to approximate Eq. (4.12) and Eq. (4.13) as follows:

$$\frac{dy}{d\theta_1} \approx \dot{y}/p^2 + 2p + 1 \tag{4.14}$$

and

$$\frac{dy}{d\theta_2} \approx y/p^2 + 2p + 1 \tag{4.15}$$

Finally, the adaptation of θ_1 and θ_2 is:

$$\frac{d\theta_1}{dt} = -\gamma e \frac{\dot{y}}{p^2 + 2p + 1} \tag{4.16}$$

and

$$\frac{d\theta_2}{dt} = -\gamma e \frac{y}{p^2 + 2p + 1} \tag{4.17}$$

It should be noted that MRAC designed based on MIT rule is locally stable if γ is small and initial conditions $\theta_i(0)$ are close to the nominal values of θ_i. Otherwise the aforementioned MIT rule may lead to instability and unbounded signal response.

For Qball-X4 quadrotor UAV the MIT rule is generated using the same procedure for all axes. Based on Eq. (4.5) one can assume:

$$\hat{U}_x = g\theta \tag{4.18}$$

$$\hat{U}_y = -g\phi \tag{4.19}$$

$$\hat{U}_z = \frac{U_z}{m} - g \tag{4.20}$$

Considering Eq. (4.5), $\ddot{x}$, $\ddot{y}$ and $\ddot{z}$ can be written as follows:

$$\ddot{x} = \hat{U}_x \tag{4.21}$$

$$\ddot{y} = \hat{U}_y \tag{4.22}$$

$$\ddot{z} = \hat{U}_z \tag{4.23}$$

For the X axis Eq. (4.21) is considered as the plant model. As described in Eq. (4.7) the reference model is as follows:

$$\ddot{x}_m = -2\dot{x}_m - y_m + r \tag{4.24}$$

and the control input (Eq. (4.8))

$$u = \theta_1 \dot{y} + \theta_2 y + r \tag{4.25}$$

To obtain a perfect model following one can choose $\theta_1 = -2$ and $\theta_2 = -1$. However, since the there is no a_1 and a_2 parameters as in the general MIT formulation, fixing θ_1 and θ_2 as -2 and -1 respectively is not the best solution due to simplified model which is not matched with the real model. Therefore, θ_1 and θ_2 are adapted using MIT rule and the final solution is the same as equations Eq. (4.16) and Eq. (4.17). The same formulation is used to generate MRAC parameters for Y and Z axes.

Experimental Results

For the FTC purpose based on MRAC, hovering control and trajectory tracking control scenarios with injected fault are applied to Qball-X4 and the experimental testing results are shown in Figs. 4.5, 4.6 and 4.7. In Fig. 4.5 a square trajectory of 0.5 m$\times$ 0.6 m is tracked by Qball-X4 under fault-free condition. Also, in Fig. 4.6 the trajectory tracking

performance under a fault scenario with faults injected only to forward (rotor 1) and left (rotor 3) actuators with a 12% control effectiveness loss is presented. As can be seen from Fig. 4.6 the real trajectory of the Qball-X4 derivates from the desired path immediately after the occurrence of the fault.

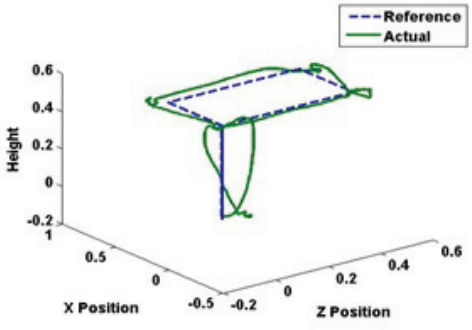

Figure 4.5: Square trajectory in fault-free condition with MRAC

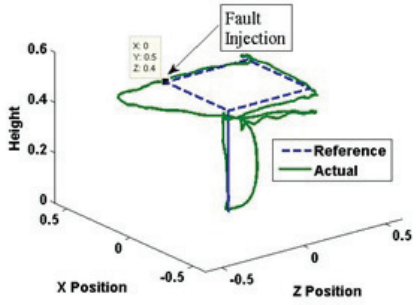

Figure 4.6: Square trajectory in fault condition with MRAC

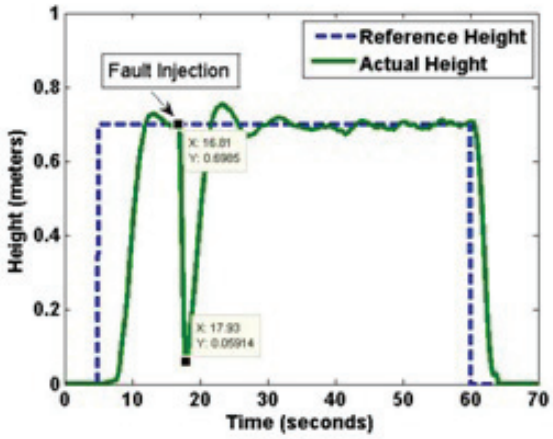

Figure 4.7: MRAC with 18% of fault on all motors

In Fig. 4.7 the height control of Qball-X4 is shown under 18% of loss of effectiveness (LOE) in all actuators.

4.3 Control Allocation and Re-allocation for a Modified Quadrotor Helicopter against Actuator Faults

Redundancy in sensors, actuators and all other essential components of an airplane plays a vital role in increasing of the flight safety in the case of fault/failure occurrence in those components. From the actuator redundancy point of view, a conventional airliner is designed to fly using redundant actuators in combination with it's primary control surfaces. Almost all modern military aircrafts and the new generation of civil aircrafts such as Boeing 777 and Airbus A320/330/340/380 have triplex or quadruplex redundant actuation systems, flight control computers and data bus systems, air data and motion sensor systems. Such redundancies are implemented in both hardware and software [35]. In fault conditions the pilot is able to use flaps, slats, spoilers and the thrust vector in the case of fault or failure in elevator to control the pitch of the plane. Some of these actuators can be used in the case of stuck in place or floating rudder or aileron. Hence, over-actuated systems have benefits over under-actuated systems in both fault-free and fault conditions.

The quadrotor also is an under-actuated system. Thereby, this section investigates the modification of Qball-X4 quadrotor UAV by adding two actuators to the helicopter for increased hardware redundancy and enhanced FTC capability. This redundancy improves the helicopters safety margin and on the other hand, it extends the FTC methods that can be flying-tested on the quadrotor with more severe faults/damages. For illustration, control re-allocation is employed as a fault-tolerant control method in the presence of actuator faults. Some experimental results using Qball-X4 are given to illustrate the capabilities of the modified UAV system. It should be noted however that the Qball-X4 quadrotor UAV, as mentioned before, is an under-actuated system. As a consequence, a complete loss of one of the actuators will crash the quadrotor and this limits the number of approaches and the fault amplitudes that can be tested on the UAV system. The importance of hardware redundancy and the limited capability of the quadrotor UAV as a testbed for FTC demonstration was

the motivation of the design and apply the hardware redundancy to the Qball-X4 quadrotor UAV by adding two new brushless motors to the system. In the first, the Qball-X4 quadrotor helicopter was modified by adding two additional redundant motors. From one hand, this redundancy increases the safety margins of Qball-X4 UAV where a complete loss of an actuator can still be accommodated due to the presence of a redundant one. On the other hand, it extends the FTC approaches as well as the fault amplitudes that can be tested on the system. It turns out that redundancy makes from the Qball-X4 quadrotor an affordable aerial vehicle to test more FTC methods in the case of major faults which is not possible with the conventional quadrotor. In the second part, as a direct consequence of the increased redundancy, the application of the Pseudo- Inverse method for control allocation is investigated. It also uses control re-allocation for FTC in the presence of actuators faults. Control allocation and re-allocation which are not applicable to the conventional quadrotor have been implemented and experimentally tested on the modified Qball-X4 UAV testbed. Finally in the third part, two types of faults are considered for the UAV testbed: simulated faults and real damage of a propeller. In the former case, the fault is injected by limiting the control input to the faulty actuator. In the latter case, the fault is injected by breaking the propeller and thus limiting the thrust that can be generated by the actuator.

4.3.1 Modifying Qball-X4 into Qball-X6

The quadrotor UAV is a relatively simple, easy to fly and affordable aerial vehicle. This makes it an excellent testbed to develop, implement and flying-test several technologies in control, telecommunication and mechatronics engineering. However, as stated above, the Qball-X4 is an under-actuated system where a major fault in one of the motors/propellers will result in a crash. As an example, it has been found through experimentations that it is not possible to maintain the stability of the Qball-X4 for a damage over 20% of one propeller. This limits the FTC methods that can be tested on such a conventional quadrotor UAV. To address this issue, two more actuators are added to Qball-X4 quadrotor UAV.

The first additional actuator (which is the 5th actuator in the system) is put underneath the 4th one. The second additional actuator (which is the 6th actuator in the system) is put underneath the 2nd one. The redundant 4th and 5th (respec. the 2nd and 6th) actuators are now aligned on the same axis where both rotate in the same direction and generate thrusts upwards in the body frame. Figures 4.8 and 4.9 show the redundant actuators after modification. In the sequel, the modified Qball-X4 will be referred to as MQ-X4.

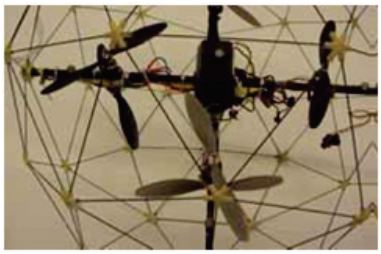

Figure 4.8: View of the redundant motors

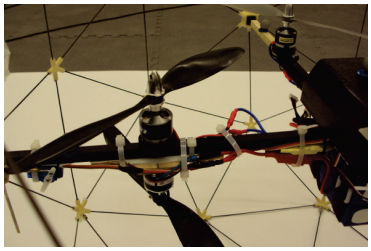

Figure 4.9: The 4th (top) and 5th (bottom) actuators

It should be noted that some quadrotor-type helicopters with actuator redundancy are already existing in the market. However, most of these are designed for manual control through a remote controller and with the objective of being used for surveillance application with very high payload capabilities due to redundant actuators and not for automatic/autonomous control purposes. Draganflyers X6 (Fig. 4.10) and X8 (Fig. 4.11) can be named for example. These helicopters are capable to carry high resolution cameras for high quality of video capturing in both indoor and outdoor flying environments. The Eye Droid 8 Rotor Octocopter UAV shown in Fig. 4.12 is another example of quadrotor-type UAV with actuator redundancy.

Figure 4.10: The Draganflyer X6 [55]

Figure 4.11: The Draganflyer X8 [55]

Figure 4.12: The Eye Droid octocopter [56]

After upgrading the Qball-X4 to the MQ-X4, the mathematical model described in section 4.1.3 remains the same with some changes in the total mass m and the moments of inertia J_1, J_2 and J_3. However, with the two additional actuators, the relation given in Eq. (4.3) becomes:

$$\begin{bmatrix} u_z \\ u_\theta \\ u_\phi \\ u_\psi \end{bmatrix} = \begin{bmatrix} K & K & K & K & K & K \\ KL & -KL & 0 & 0 & 0 & -KL \\ 0 & 0 & KL & -KL & -KL & 0 \\ \overline{K} & \overline{K} & -\overline{K} & -\overline{K} & -\overline{K} & \overline{K} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} \tag{4.26}$$

where $\overline{K} = KK_\psi$. It can be seen that with the MQ-X4, six actuators are used to produce four movements (the lift u_z and the three torques u_θ, u_ϕ, and u_ψ). The system has then actuator redundancy and therefore it can better tolerate actuator faults. In addition, control allocation and re-allocation techniques can be tested on the system which was not possible to do with the Qball-X4.

4.3.2 The Principles of Control Allocation and Re-allocation

The control allocation and re-allocation methods are applied to the MQ-X4. Control re-allocation can be used to accommodate actuator faults. The main advantage of control re-allocation is that fault-tolerant control can be achieved without the need to reconfigure/restructure the baseline controller and that only the control duties are redistributed

among the redundant healthy actuators while taking into consideration the limited capabilities of the damaged ones. Consider the following continuous-time system:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{4.27}$$

where $A \in \Re^{n\times n}$, $B \in \Re^{n\times q}$ and $C \in \Re^{m\times n}$ are the state, the control and the output matrices, respectively. $x \in \Re^n$ is the state vector, $u \in \Re^q$ is the control input vector, and $y \in \Re^m$ represents the system outputs. Control allocation is generally used for over-actuated systems where the number of operable control is greater than the controlled variables, i.e. when the matrix $B \in \Re^{n\times q}$ is column rank deficient $(rank(B) = n < q)$. In order to solve the control law problem, a virtual system variable $v(t) = Bu(t)$ is synthesized in order to define an equivalent representation of the plant such as:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + v(t) \\ y(t) &= Cx(t) \end{aligned} \tag{4.28}$$

The control law $v(t)$ is designed so that to attain the desired objectives. The control allocation aims then to distribute the desired control inputs $v^d(t)$ over the available actuators. Such a problem has been extensively studied in recent years and several numerical procedures have been proposed in [57] and [58]. One of the simplest control allocation methods is the Pseudo-Inverse approach. It can be shown that the optimization problem:

$$\min_u \; u(t)^T W^{-1} u(t) \tag{4.29}$$

under the constraints:

$$Bu(t) = v^d(t) \tag{4.30}$$

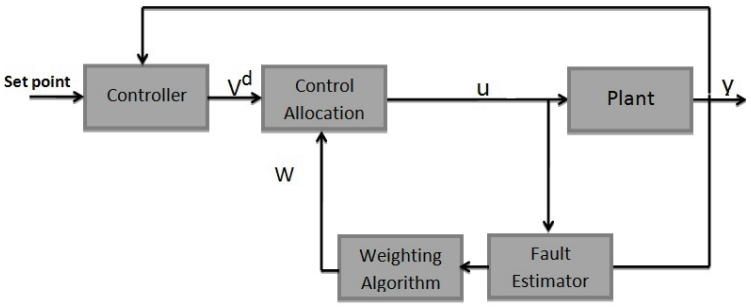

Figure 4.13: Control allocation/re-allocation strategy

has the explicit solution [59]:

$$u(t) = WB^T \left(BWB^T\right)^{-1} v^d(t) \tag{4.31}$$

with $W = diag([w_1, w_2, ..., w_q]^T)$ is a positive definite weighting diagonal matrix. If W is chosen as an identity matrix, then all actuators have the same priority and Eq. (4.31) then becomes:

$$u(t) = B^T \left(BB^T\right)^{-1} v^d(t) \tag{4.32}$$

where $B^T \left(BB^T\right)^{-1}$ is the pseudo-inverse of matrix B. In [44], control re-allocation is employed as a fault-tolerant method and applied to the ADMIRE (Aero-Data Model In Research Environment) benchmark aircraft model in the presence of actuator faults. In [45], this is achieved by adjusting the weighting matrix W to penalize the faulty actuators and reduce their duties and consequently increase the duties of the healthy actuators. It can be seen that when $w_i \to 0$, then $1/w_i \to \infty$ and according to Eq. (4.29) the i^{th} element of $u(t)$ is highly penalized. The adjustment of matrix W can be done in function of the fault amplitude, i.e. the higher the fault amplitude is in the i^{th} actuator, the lower w_i is the i^{th} diagonal element of W. For instance, for a complete loss of the i^{th} actuator, one can set $w_i = 0$.

The control allocation/re-allocation strategy is illustrated in Fig. 4.13 [45]. In the fault-free case, W is set to the identity matrix. However, control re-allocation requires an

information about the occurred fault and thus an FDD module is needed to detect, isolate and identify faults. This point is beyond the scope of this work and is not considered here. It is then assumed that an FDD module is present to provide the necessary information about the fault for control re-allocation.

4.3.3 Experimental Results

In all the experiments, the system is required to hover at an altitude of 0.8 m, the faults are taking place at time instant $t = 25$ s and the system lands at $t = 50$ s. Two types of faults are considered: simulated and real faults. In the former case, the maximal limit of the PWM input of the faulty actuator is suddenly reduced. This limits the PWM input that the Gumstix on-board computer can send to the corresponding ESC and simulate a loss of actuator control effectiveness. In the latter case, a part of the propeller is ejected using a servo-controlled mechanism to simulate propeller damages which may occur in real life with the purpose of testing the fault-tolerant control strategies. All the faults are taking place in the 5th actuator. The baseline controller is LQR and the controller gains are kept constant during all experiments for a fair comparison.

A) Fault-free Case: In the fault-free case, the weighting matrix $W \in \Re^{6\times 6}$ is chosen as the identity matrix. Hence all the six actuators have the same priority/expectation. The 3D evolution of the MQ-X4 is shown in Fig. 4.14 and the corresponding PWM inputs are given in Fig. 4.15. One can see in Fig. 4.14 that the system hovers and lands with a horizontal tracking error of less than 0.15 m.

It should be noted that all the actuators are active during the nominal fault-free condition. Regarding the PWM inputs in Fig. 4.15, two things can be noticed. First, the redundant 2nd and 6th actuators (respectively the 4th and 5th ones) have the same PWM inputs. This is because all actuators have the same priority and thus redundant actuators will take the same duties in counteracting the system weight and keeping it in hovering

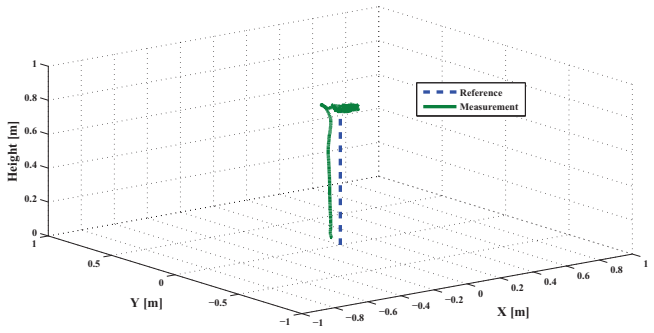

Figure 4.14: Fault-free case based on MRAC

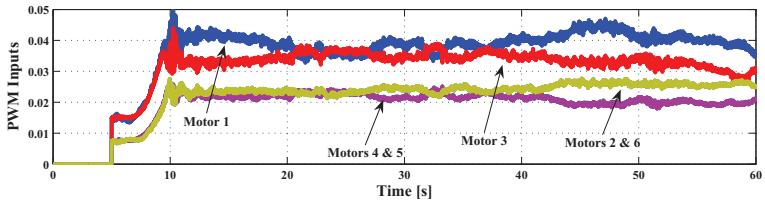

Figure 4.15: PWM signal for fault-free case

position. This leads to the second remark where the PWM inputs to the redundant actuators are almost the half of those of the non-redundant ones. This means that the redundant actuators are giving half the thrust that is usually needed without redundancy.

B) Simulated Actuator Fault Case: This kind of faults may take place due to a software fault onboard the computer or in the ESC. For the simulated actuator fault in the 5th actuator, it is assumed that the PWM input limit which equals to 0.05 suddenly drops to 0.018 thus limiting the PWM inputs that the Gumstix can send to the corresponding ESC and consequently reducing the maximal thrust that can be generated by the actuator. This corresponds to 64% loss of control effectiveness. Without control re-allocation the system goes unstable as shown in Fig. 4.16.

The correspondent PWM inputs are illustrated in Fig. 4.17. One can see that at 25 s, the PWM input to the 5th motor seems to be stuck at 0.018.

This can be explained as follows: without control re-allocation, the controller (unaware of the occurred fault) continues to have the same expectations from the faulty actuator as before fault occurrence and thus it continues to ask it for the same pre-fault duties.

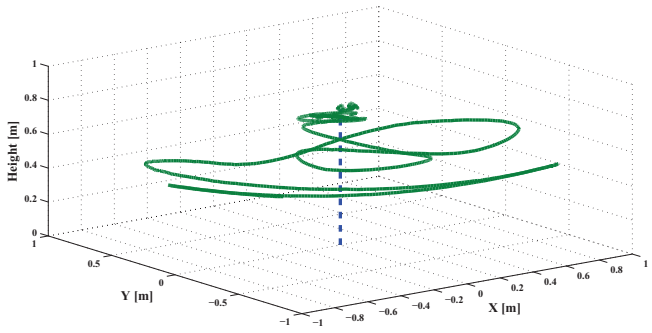

Figure 4.16: Simulated loss of control effectiveness of 64% without control re-allocation

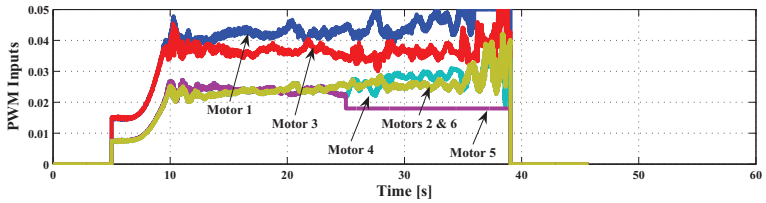

Figure 4.17: PWM signal for simulated fault without control re-allocation

Due to the occurred fault, the 5th actuator is not able to generate the required thrust anymore. The controller responds by increasing the PWM inputs to the 4th and 5th motors to compensate for the increasing tracking error. This leads to the saturation of the PWM input to the 5th motor at 0.018 whereas the PWM input to the 4th motor continues to increase but not fast enough to keep system's stability. To prevent system's damage and for operators safety, the system is stopped which explains the sudden cut of the PWM inputs at about 39 s.

The control re-allocation is applied at time instant $t = 26\ s$ where $1\ s$ is assumed to be taken by an FDD module to detect, isolate and identify the occurred fault. The control re-allocation is achieved by setting w_5 to 0.3 and thus reducing the expectations from the faulty actuator and consequently increasing those of the redundant healthy one (the 4th actuator). In this case, the system experiences a small drift but it goes back in short time to the hovering position and lands safely at $t = 50\ s$ as shown in Fig. 4.18.

The PWM inputs are shown in Fig. 4.19. The vertical line represents the instant where the control re-allocation is taking place. As expected, the PWM input to the 5th

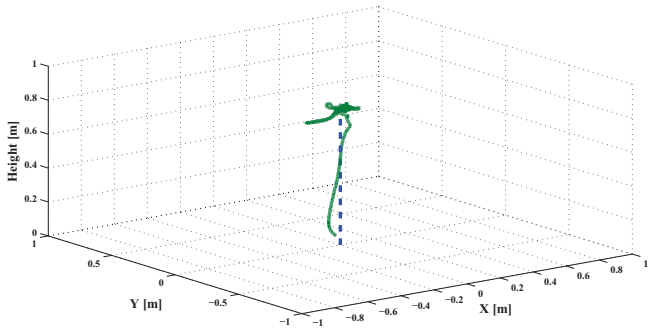

Figure 4.18: Simulated loss of control effectiveness of 64% with control re-allocation

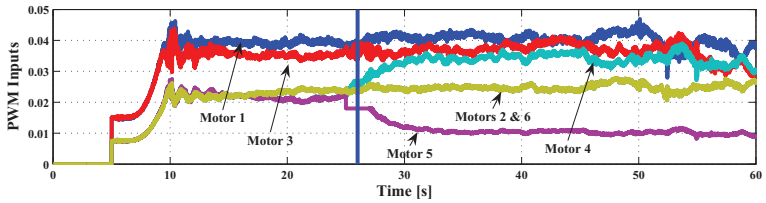

Figure 4.19: PWM signal for simulated fault with control re-allocation

motor drops down to about 0.01 while the PWM input to the 4th motor quickly increase to replace the faulty one. It turns out that control re-allocation can keep system's stability and the PWM input to the 4th actuator is close to those of the non-redundant 1st and 3rd actuators.

C) Damaged Propeller Fault Case: For this case of real actuator damage, the propeller is cut and glued again and then the fault is injected by using a servo-controlled mechanism to eject the propeller tips during flight. Fig. 4.23 shows a broken propeller where the broken part represents 70% of the overall length of the propeller. Due to the actuator redundancy, the MQ-X4 can tolerate certain level of actuator faults without the need for fault-tolerant control. In contrast to the original Qball-X4 where it was impossible to maintain system's stability in the presence of a propeller damage of 20%, Figures 4.20 and 4.21 show that the MQ-X4 can maintain its stability when a propeller damage of 35% is injected without control re-allocation.

A larger propeller damage of 70% is investigated next for both with and without control re-allocation cases. The damage percentage of the propeller is measured by using the

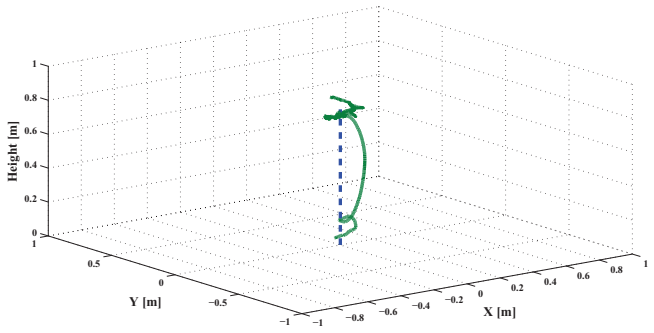

Figure 4.20: Propeller damage of 35% without control re-allocation

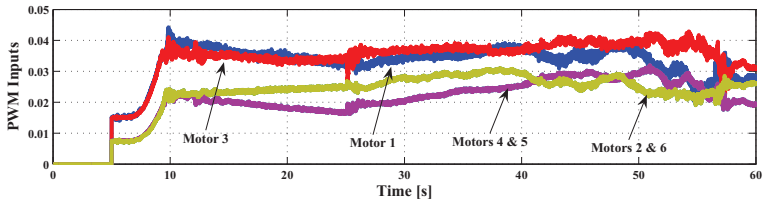

Figure 4.21: Propeller damage of 35% without control re-allocation

thrust stand in the way that the force generated by the healthy propeller first was measured and compared to the force generated by the damaged propeller using the same stand and in the same test conditions (Fig. 4.22).

As before, the fault is injected at $t = 25\ s$ and Fig. 4.24 shows that the system is not able to keep its stability without control re-allocation. Since the 5th propeller is not able to generate the required thrust anymore, Fig. 4.25 shows that the controller gradually increases the PWM inputs to both 4th and 5th actuators but fails to react appropriately and fast enough to maintain stability.

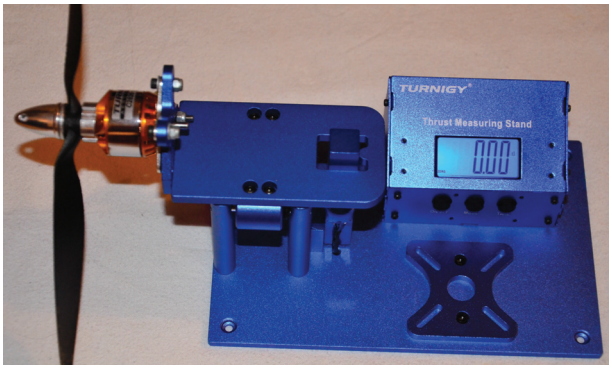

Figure 4.22: Turnigy thrust measuring stand

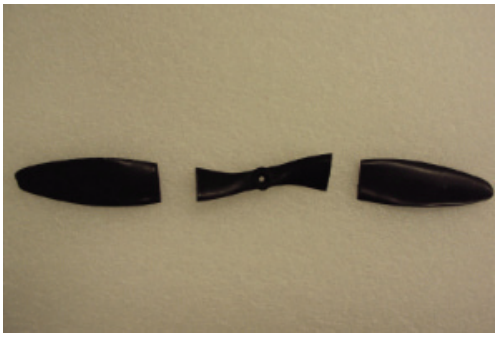

Figure 4.23: A 70% damaged propeller

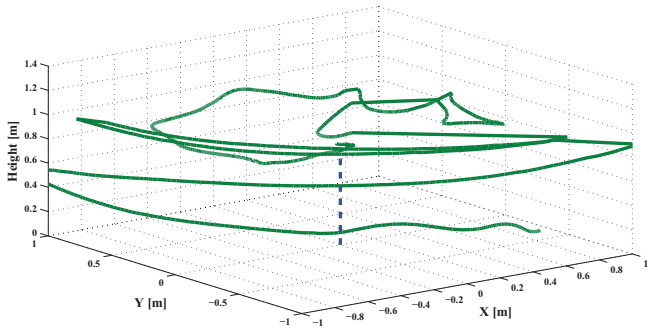

Figure 4.24: Propeller damage of 70% without control re-allocation

The same experiment above is repeated but this time with control re-allocation. The control re-allocation is taking place at $t = 26\ s$ where w_5 is set to 0.1 to penalize the faulty actuator. Fig. 4.26 shows that once the fault occurs, the system deviates from the hovering position but goes back to it after short time and lands safely at $t = 50\ s$. Fig. 4.27 shows that, as in the simulated fault case, the control re-allocation allocates less duties to the 5th actuator whose PWM input drops to about 0.005. In the same time, the control re-allocation allocates more duties to the 4th actuator whose PWM input rapidly increases to maintain system's stability. One can also see that before re-allocation, the PWM inputs to the 4th

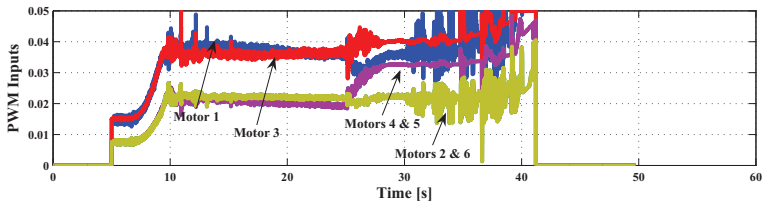

Figure 4.25: PWM signal for propeller damage of 70% without control re-allocation

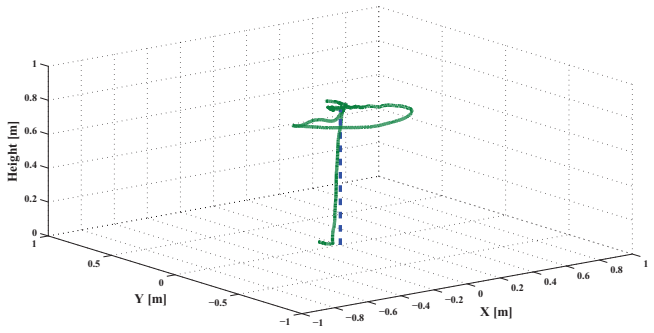

Figure 4.26: Propeller damage of 100% with control re-allocation

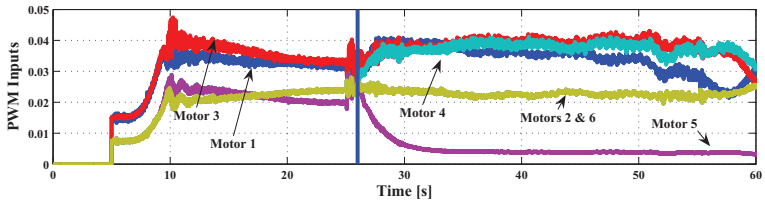

Figure 4.27: Propeller damage of 100% with control re-allocation

and 5th actuators are the same which is not the case after re-allocation (which is indicated by the vertical line in Fig. 4.27). It should be noted that during fault injection and due to the aggressive collision between the propeller and the servo-controlled mechanism, the 5th actuator was completely ejected from its mount which resulted in a 100% fault instead of the 70% planned. It turns out that control re-allocation allows to maintain system's stability with 100% damage whereas without re-allocation, the system stability is lost with a damage of 70%.

D) Discussion: The MQ-X4 shows that existence of actuator redundancy allows the quadrotor-type helicopter to accommodate actuator faults to some level without the need for FTC. The modified system also extends the FTC methods that can be applied and flying-tested. However, additional motors can bring many difficulties in hardware which makes the work not very easy to accomplish. These difficulties include the extra payload and the faster drain of electrical current from the onboard Lithium-Polymer batteries which has a direct effect on flight time.

4.4 Fault-Tolerant Control of Qball-X4 Quadrotor UAV Based on Gain-Scheduled PID Control

As mentioned in section 3.2, PID controllers are the most common controllers used in the industry. PID controller is a feedback controller that helps to attain a set point irrespective of disturbances or any variation in characteristics of the plant. It calculates its output based on the measured error and the three controller gains; proportional gain K_P, integral gain K_I, and derivative gain K_D. The proportional gain simply multiplies the error by a factor K_P. This reacts based on how big the error is. The integral term is a multiplication of the integral gain and the sum of the recent errors. The integral term helps in getting rid of the steady state error and causes the system to catch up with the desired set point. The derivative controller determines the reaction to the rate of which the error has been changing. The equation describing the PID control technique is presented in section 3.2. The effect of each gain on control output characteristics can be described as in Fig. 4.28.

Parameter Increase	Rise Time	Overshoot	Settling Time	Steady-state Error
K_P	↓	↑	Small Change	↓
K_I	↓	↑	↑	Great Reduce
K_D	Small Change	↓	↓	Small Change

Figure 4.28: The effect of PID gains on the the control response

Gain scheduling techniques such as GS-PID, can be used when the gains and the time constants vary based on varying parameters of the plant. In aerospace systems, gain scheduling techniques are very popular and useful. Fon example, one can tune a gain sheduling controller for different sections of the flight envelope of an aircraft. As aircraft parameters vary over the time (speed, altitude, etc.) the gain scheduler mechanism switches the parameters based on flying conditions and time. In quadrotor case, the same technique can be used for FTC purpose. In this section, a control strategy by using gain-scheduled based PID (GS-PID) controller is used for fault-tolerant control of the Qball-X4 testbed

which has been described previously in section 4.1.

4.4.1 Passive Fault-Tolerant Flight Control of Qball-X4 Quadrotor UAV Based on Gain-Scheduled PID Control

In this section two PID-based control techniques, PID and GS-PID are applied to Qball-X4 for fault-free and fault scenarios. The fault is injected at the same time in all four motors in hover.

As mentioned in section 2.2.1, a passive fault-tolerant controller is designed to handle a certain type and magnitude of fault. However the term "passive", in this section, mainly is used to show that there is no FDD block used for the fault-tolerant controller, since it is assumed that the time, magnitude the location of the fault are known in advance. Based on this information the controller is pre-tuned and has been set up to switch (interpolate) between the PID gains on the proper time.

For better presentation of the switching effect in the GS-PID controller for height control of Qball-X4, a single PID controller is tuned for fault-free and hovering condition and is applied to Qball-X4. As it can be seen in Fig. 4.29 the single PID controller which is tuned well for normal take-off and hovering, is not able to handle the fault condition that is set to 18% of power loss in all motors. However, as shown in Fig. 4.30, the GS-PID can handle the fault and result in better height hold than the single PID controller even with certain time delays in control gains switching (Fig. 4.31 and Fig. 4.32). The best performance with GS-PID controller is presented in Fig. 4.30.

As results show, the fault detection time is vital for the stability and the performance of the Qball-X4. Comparisons with time delay of 0.5s and 1s are shown in Fig. 4.31 and Fig. 4.32 respectively. A Better performance with a shorter time delay of 0.5s has been achieved which verified the importance of fast and correct fault detection and control

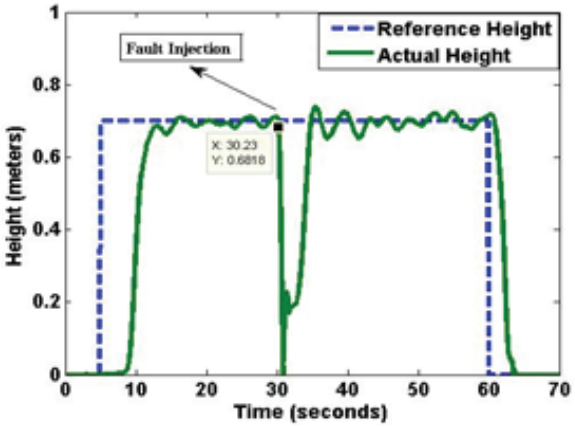

Figure 4.29: Single PID controller in the presence of 18% of fault on all actuators

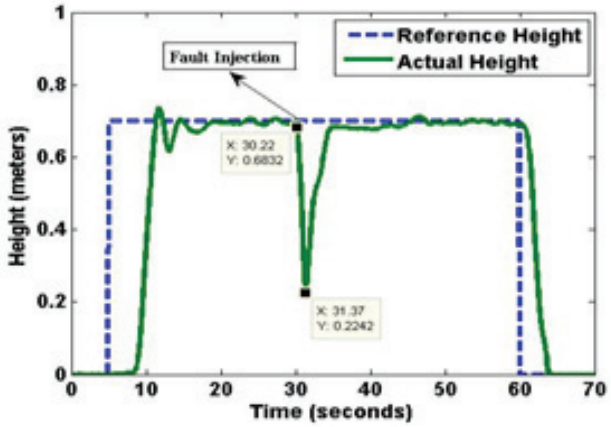

Figure 4.30: GS-PID controller in simultaneous fault injection and gains switching in the presence of 18% of fault on all actuators

switching (reconfiguration) after fault occurrence. If the fault occurrence and the switching of controller gains are set at the same time, i.e. with the perfect fault detection and diagnosis, the best result will be achieved for the GS-PID as shown in Fig. 4.30. It should be mentioned that since only height control is implemented by GS-PID, a LQR with Integral Action (LQR-IC) controller is also used to control the pitch and roll motion of the unmanned quadrotor helicopter during the experimental tests.

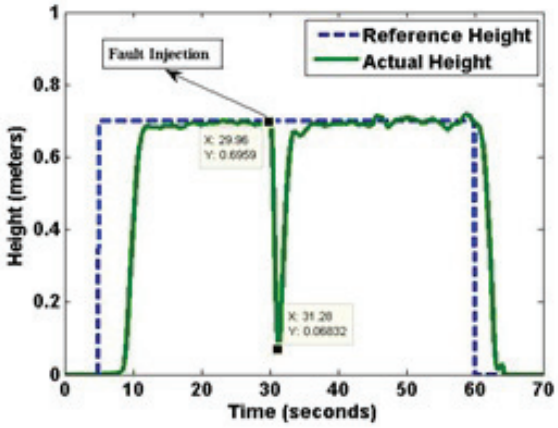

Figure 4.31: GS-PID control with 0.5s time-delay in controller gains switching after the fault occurrence

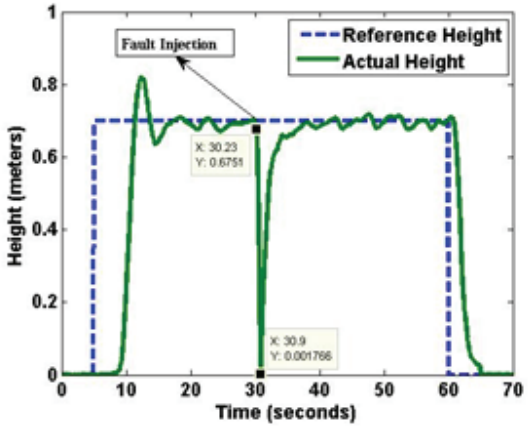

Figure 4.32: GS-PID control with 1s time-delay in controller gains switching after the fault occurrence

4.4.2 Active Fault-Tolerant Control of Qball-X4 Quadrotor UAV Based on Gain-Scheduled PID Control

In this section the same GS-PID control structure as previous work is used, based on a set of pre-tuned gains of controllers for handling the simulated fault conditions. What differs this work from the previous one is the presence of a Fault Detection and Diagnosis (FDD) block which is the essential block for active fault-tolerant control method. In order to obtain the best stability and performance of Qball-X4 under fault conditions, as mentioned before, the switching action from one set of pre-tuned PID gains to another set is highly coupled with the (FDD) block. In fact, in previous work, it has been assumed that the time and magnitude of the fault is known in advance and the switching is set to the lowest transient time to achieve the best performance. But in active way, there is no information about the time of fault occurrence and it should be detected using FDD block. The FDD block is designed to detect the fault as quick as possible in order to get the best possible performance.

The FDD function is based on the residual generated by the difference between real height (measured by OptiTrack visual system) of the Qball-X4 and the desired height (set point). Since during the flight there exist also disturbances, it was essential to define a threshold for FDD scheme not to be affected by the disturbances. Tuning of the afore-mentioned threshold was one of the main challenges of this work since it should be set very precisely for quick fault detection in FDD block as well as switching of the GS-PID controller at the minimum possible time. There is a direct relation between the threshold boundaries with the fault detection followed by gain switching action. In this experiment the tightest designed threshold was able to detect the fault and switch the gain in 0.23 second after the fault occurrence. However, in take off and landing phases different PID controllers are used and the FDD block was disabled since the fault is not assumed to occur during take off and landing.

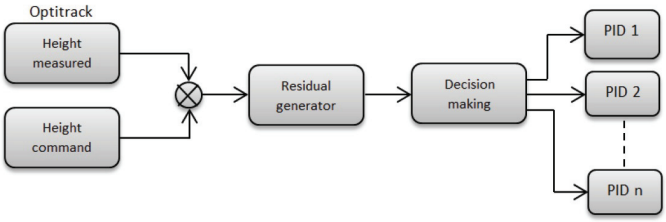

Figure 4.33: FDD block integrated with GS-PID controller

Experimental Results

Like the "Passive" case in previous section, for better comparison purpose and as a baseline controller of the Qball-X4 under normal flight conditions the same single PID controller which is well-tuned for taking-off, hovering and landing scenario under normal flight condition, is designed first. Such a controller is used also for fault scenario with the same magnitude of fault in the previous experiment (18% LOE in all actuators).

As shown in Fig. 4.34, the single PID was not able to maintain the desired height of the Qball-X4 when the fault occurred at 30 second and the Qball-X4 almost touched the ground after fault occurrence, although the Qball-X4 is still able to keep the desired height finally benefited by the outstanding feature of PID controller.

As shown in Fig. 4.35, unlike the single PID, the gain-scheduled PID was able to handle the fault by preventing the Qball-X4 from hitting the ground due to fast reaction of FDD block (0.23s) for the FDD block to detect the fault and switch the PID gains on the Qball-X4 actuators.

4.4.3 Payload Drop Application of Unmanned Quadrotor Helicopter Based on Gain-Scheduled PID Control

In continuation of GS-PID control technique, the GS-PID and Model Predictive Control (MPC) techniques are applied separately to Qball-X4 helicopter in order to control the height and while carrying a payload weighing one-fourth of its total weight as well as dropping the payload at a predetermined time.

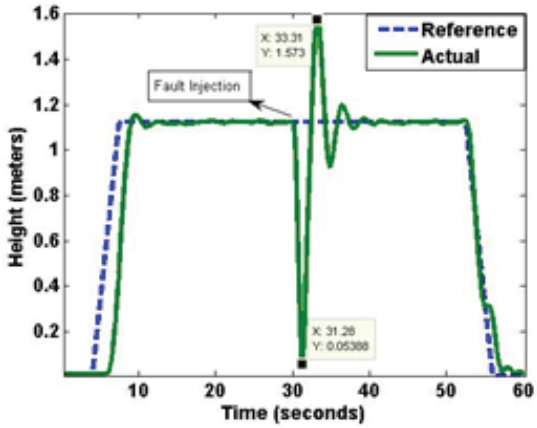

Figure 4.34: Single PID controller in the presence 18 percent of fault on all actuators

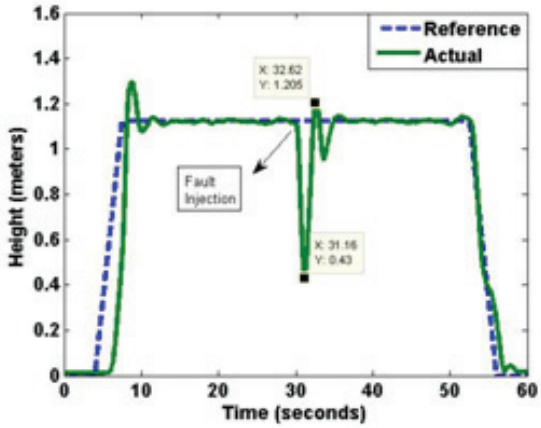

Figure 4.35: GS-PID controller in the presence 18 percent of fault on all actuators

Airdrop is a very useful and common manoeuvre for flying vehicles for different civil and military applications such as: firefighting, delivery of supplies to ground forces and also flight test of hypersonic and glider-type experimental airplanes which need to be mounted on another flying vehicle and to be released in the air. During the recent earthquake in Japan, military helicopters were dumping seawater on a stricken nuclear reactor in north-eastern Japan for cooling the overheated fuel rods inside its core. Also U.S. Joint Forces Command continues to develop the Joint Precision Airdrop System (JPAS) with new ways of delivering supplies to ground forces while minimizing risks to soldiers. A joint military utility assessment team recently observed and rated airdrops of cargos of 6,000 to 10,000 pounds at Yuma Proving Ground, Ariz [71].

The payload drop scenario has the same effect as fault has on Qball-X4 since the mass of Qball-X4 suddenly changes due to payload drop and it needs the gains to be scheduled for handling the Qball-X4 and to avoid the overshoot of the helicopter and crash. The performance of these two control techniques are compared in the next section. Also the payload drop mechanism is shown in Fig. 4.36.

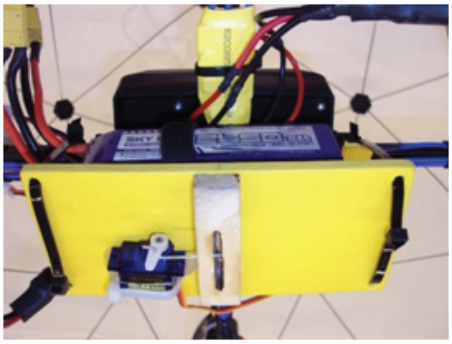

Figure 4.36: Servo based payload releasing mechanism

Experimental Results

As mentioned above, in this work two control techniques are applied and implemented in real time to a Qball-X4 quadrotor UAV: GS-PID and MPC. In order to demonstrate effectiveness of the control algorithms and prove their stability, two experiments have been set

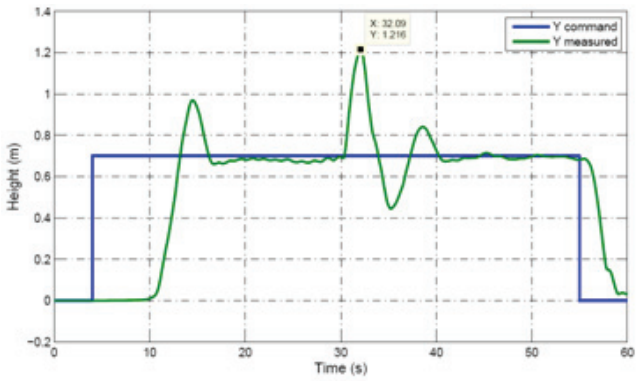

Figure 4.37: Height control based on single PID control

up. In the first set of experiments, focus is on the PID control, starting with a single PID controller to take over control of the quadrotor over the phases of take-off, flight with payload onboard, payload drop, and finally landing. Although the single PID controller was capable of keeping the desired height, it was not able to eliminate undesired overshoot at the moment of payload drop (see Fig. 4.37). Hence, the single PID controller is replaced by the GS-PID controller to improve performance in both main aforementioned phases (see Fig. 4.38). For the same objective to evaluate performance of Qball-X4 helicopter the third experiment was conducted and flight tested using a MPC controller. The take-off and payload carrying flight phases were better than a single PID and GS-PID in terms of offset-free tracking and takeoff overshoot, but compared to GS-PID the overshoot was 3.4% more at the time of payload drop as shown in Fig. 4.39. This can be reduced more by proper tuning of the controller using the controllers design parameters such as the prediction horizon, the control horizon, and control change penalizing parameter in the cost function. For the case of a single PID controller the Qball-X4 did maintain the desired height but was not satisfactory in performance since 73 % of overshoot happened over the course of payload drop. On the other hand, the MPC controller noticeably improved systems reaction to payload drop, eliminating vertical jerk at the instant of payload dump with 17 % of overshoot at the moment of payload drop. The best performance is achieved using GS-PID with 13.6% overshoot at the time of release.

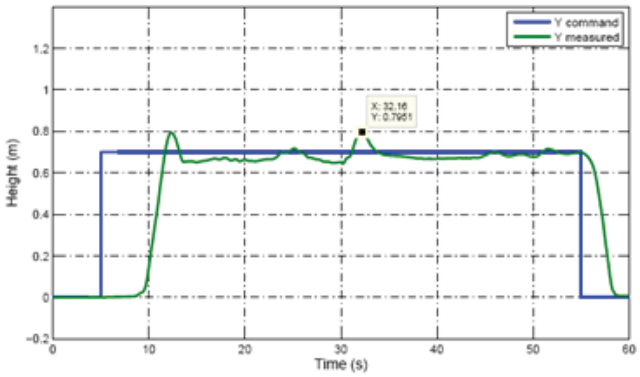

Figure 4.38: Height control based on GS-PID control

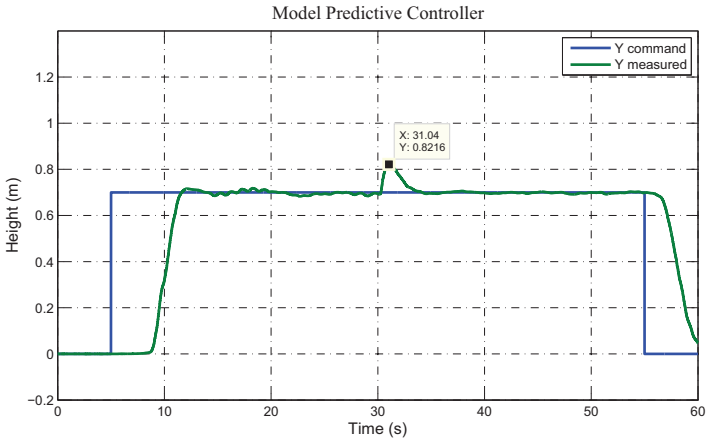

Figure 4.39: Height control based on MPC

Among all the three controllers applied in this work, GS-PID is suggested for the scenario of payload drop. But fine tuning of gains as well as healthy and fully charged Li-Po batteries are two essential factors affecting systems behavior. For instance, a mid-charged battery could have a direct effect on fine tuned controller gains and deviate them from the desired values. Due to the nature of non model-based switching techniques such as the gain scheduling PID there is no guarantee of system stability.

Chapter 5

Linear Parameter Varying Control of the Quadrotor Helicopter UAV

This chapter focuses on the synthesis of Linear Parameter Varying (LPV) control based on two different LPV control structures. In the first structure the H∞ self gain-scheduling control technique is used to obtain the LPV controller and in the second method, the composite quadratic Lyapunov function and the quadratic cost function are used to find the optimal state feedback gain. Finally, the Qball-X4 quadrotor helicopter is used as an illustrative plant to compare the results of both LPV control structures.

Aerospace systems have been benefited from a variety of modern multivariable feedback control techniques over the past few decades. As explained in previous chapter, most of these controllers are designed based on various operating points of the linearized plant and the gain scheduling technique is used for interpolation between the controller gains for each point. The interpolation process is mainly designed based on parameter variables of the plant. However, these ad-hoc methods are not capable of guaranteeing the stability, performance and the robustness of the controller other than the design points. Therefore,

design and application of scheduling multivariable controllers is a complex and time consuming task. To overcome these difficulties, LPV control technique is introduced as an alternative gain scheduling method. As it can be understood from its name, the LPV systems are based on linear structure with a set of varying parameters over time. In other words, the model and the controller are linear but the dynamics of the plant model and the controller are dependent to certain time-varying parameters. The value of time-varying parameter is based on plants operational conditions, and can be measured in real time. When the plants dynamics vary over the time based on operating conditions, the appropriate pre-designed controller can be selected based on weighting functions to guarantee the performance, stability and robustness of the plant. A comprehensive survey about LPV control applications validated by experiments or high-fidelity simulations is presented in [72].

The LPV systems can be represented in input-output or state-space form and either in continuous or discrete-time. The continuous representation of the LPV system is shown in the following form [73]:

$$\begin{aligned} \dot{x}(t) &= A(\rho(t))x + B(\rho(t))u \\ y(t) &= C(\rho(t))x + D(\rho(t))u \end{aligned} \tag{5.1}$$

where x and y represent the states and output vectors and u and ρ represent the control input and varying parameter respectively. A general scheme of the controlled LPV system is presented in Fig. 5.1 where d is the exogenous input containing the desired set point

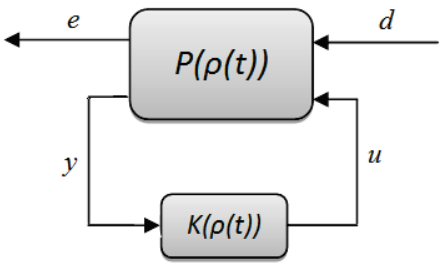

Figure 5.1: The controlled LPV system

and disturbances and e is the error between the set point and the real output of the plant.

Also u and y are manipulated variables and measured variables (output) respectively. In order to obtain the LPV model from a physical representation of the nonlinear model, three methods can be used: Jacobian Linearization, State transformation and Substitution Function method [74]. The main objective of using these methods is to distribute (hide) the nonlinearity of the system into varying parameter. Also one can use the Least Square (LS) or the Recursive Least Square (RLS) methods to obtain the model from experimental data [75]. As mentioned in the beginning of this chapter, the objective of this chapter is to present two different LPV control structures synthesized based on different techniques and to compare the performance of both control structures using Qball-X4 helicopter UAV.

5.1 Qball-X4 UAV Nonlinear Dynamic Model

As described in section 4.1, the thrust (lift) vector generated by four propellers is $T = [T_1\ T_2\ T_3\ T_4]^T$. Also twelve states are considered for the quadrotor as follows [76]:

$$\mathbf{x} = [x\ \ y\ \ z\ \ \dot{x}\ \ \dot{y}\ \ \dot{z}\ \ p\ \ q\ \ r\ \ \phi\ \ \theta\ \ \psi]^T \tag{5.2}$$

The dynamic model of the quadrotor can be represented as follows:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{5.3}$$

where

$$\begin{aligned}
f_1(\mathbf{x}, \mathbf{u}) &= \dot{x};\\
f_2(\mathbf{x}, \mathbf{u}) &= \dot{y};\\
f_3(\mathbf{x}, \mathbf{u}) &= \dot{z};\\
f_4(\mathbf{x}, \mathbf{u}) &= \ddot{x} = (\sin\psi\ \sin\phi + \cos\psi\ \sin\theta\ \cos\phi)\, U_1/M;\\
f_5(\mathbf{x}, \mathbf{u}) &= \ddot{y} = (\sin\psi\ \sin\theta\ \cos\phi - \cos\psi\ \sin\phi)\, U_1/M;\\
f_6(\mathbf{x}, \mathbf{u}) &= \ddot{z} = -g + (\cos\theta\ \cos\phi)\, U_1/M;\\
f_7(\mathbf{x}, \mathbf{u}) &= \dot{p} = \frac{J_{yy} - J_{zz}}{J_{xx}} q\, r - \frac{J_{TP}}{J_{xx}} q\, \Omega + U_2/J_{xx};\\
f_8(\mathbf{x}, \mathbf{u}) &= \dot{q} = \frac{J_{zz} - J_{xx}}{J_{yy}} p\, r + \frac{J_{TP}}{J_{yy}} p\, \Omega + U_3/J_{xx};\\
f_9(\mathbf{x}, \mathbf{u}) &= \dot{r} = \frac{J_{xx} - J_{yy}}{J_{zz}} p\, q + U_4/J_{zz};\\
f_{10}(\mathbf{x}, \mathbf{u}) &= \dot{\phi} = p + \sin\phi\ \tan\theta\, q + \cos\phi\ \tan\theta\, r;\\
f_{11}(\mathbf{x}, \mathbf{u}) &= \dot{\theta} = \cos\phi\, q - \sin\phi\, r;\\
f_{12}(\mathbf{x}, \mathbf{u}) &= \dot{\psi} = \frac{\sin\phi}{\cos\theta} q + \frac{\cos\phi}{\cos\theta} r;
\end{aligned} \tag{5.4}$$

where p, q and r are angular rates in body-fixed frames and J_{TP} is the total rotational moment of inertia around the propeller axis. The linearized LPV model can be achieved

by applying Jacobian linearization to Eq. (5.4). It should be mentioned that the hovering condition is considered for linearization.

5.2 Optimal State Feedback Linear Parameter Varying Control

For this method, the yaw angle is defined as varying parameter and its range change between $[-\pi, \pi]$. The reason for choosing only one varying parameter (yaw angle) is to simplify the LPV controller synthesis.

$$\rho(t) = \psi(t), \qquad \forall t > 0 \tag{5.5}$$

It is expected that the varying parameter (the yaw angle) appears in state matrix after linearization. The general LPV model can be re-written as follows:

$$\dot{\mathbf{x}} = A(\rho(t))\mathbf{x}(t) + B(\mathbf{u}(t) - u^*) \tag{5.6}$$

where u^* is equal to the total amount of rotor thrusts in hovering flight condition:

$$u^* = \frac{mg}{4}[1\ 1\ 1\ 1]^T \tag{5.7}$$

By applying the Jacobian linearization, the linearized state and input matrices can be presented as follows:

$$A(\rho(t)) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & g\sin\psi & g\cos\psi & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g\cos\psi & g\sin\psi & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{5.8}$$

$$B = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1/M & 0 & 0 & 0 \\ 0 & 1/J_{xx} & 0 & 0 \\ 0 & 0 & 1/J_{yy} & 0 \\ 0 & 0 & 0 & 1/J_{zz} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{5.9}$$

As it can be seen from Eq. (5.8), the state matrix is parameter dependant but nonlinear with respect to the varying parameter i.e. yaw angle. In order to transform the state matrix into a linear affine function, one can introduce the following polytope form.

$$A(\rho(t)) = \sum_{i=1}^{4} \alpha_i(\rho(t)) A_i \tag{5.10}$$

There is an ad-hoc relation between each element of A_i and $A(\rho(t))$. The nonlinear parts are equal to a_1, a_2, a_3 and a_4 for A_1, A_2, A_3 and A_4 respectively.

$$a_1 = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} a_2 = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} a_3 = \begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix} a_4 = \begin{bmatrix} -1 & 1 \\ -1 & -1 \end{bmatrix} \tag{5.11}$$

The four weighting functions $\alpha_i(\rho(t))$ are calculated as follows:

$$\begin{aligned} \alpha_1 &= (\frac{1+\cos(\rho(t))}{2})(\frac{1+\sin(\rho(t))}{2}) \\ \alpha_2 &= (\frac{1+\cos(\rho(t))}{2})(\frac{1-\sin(\rho(t))}{2}) \\ \alpha_3 &= (\frac{1-\cos(\rho(t))}{2})(\frac{1+\sin(\rho(t))}{2}) \\ \alpha_4 &= (\frac{1-\cos(\rho(t))}{2})(\frac{1-\sin(\rho(t))}{2}) \end{aligned} \tag{5.12}$$

In order to synthesize the LPV controller based on optimal control method, one can compute a parameter varying state feedback controller with the standard infinite positive horizon quadratic cost function. However, the composite quadratic Lyapunov function can be used to obtain the non-conservative result. Hereafter, the final goal is to find the optimal feedback gain by solving an optimization problem.

The general LPV system can be represented as follows:

$$\dot{x}(t) = A(\alpha(t))x(t) + B(u(t)) \tag{5.13}$$

where

$$A(\alpha(t)) = \sum_{i=1}^{N} \alpha_i(t) A_i, \;\; \alpha_i(t) \geq 0 \tag{5.14}$$

and

$$\sum_{i=1}^{N} \alpha_i(t) = 1 \;\; \forall t > 0 \tag{5.15}$$

And the infinite positive horizon quadratic cost function is illustrated by following equation:

$$J = \int_{0}^{\infty} (x^T Q x + u^T R u) dt \tag{5.16}$$

Also, the composite quadratic Lyapunov function introduced in [76] can be utilized for a set of positive definite matrices. Let $P_1, ..., P_N \in R^{n \times n}$

$$F_i = P_i^{-1}, \;\; i \in I[1, N] \tag{5.17}$$

Define $\gamma \in R^N$

$$F(\gamma) := \sum_{i=1}^{N} \gamma_i F_i, \;\; P(\gamma) := F^{-1}(\gamma) \tag{5.18}$$

and

$$\Gamma = \{\gamma \in R^N : \sum_{i=1}^{N} \gamma_i = 1, \;\; \gamma_i \geq 0, \;\; i \in I[1, N]\} \tag{5.19}$$

It is obvious that $F(\gamma), P(\gamma) > 0$ for all $\gamma \in \Gamma$ and the composite quadratic function can be defined as:

$$V_C(x) := \min_{\gamma \in \Gamma} x^T p(\gamma) x \tag{5.20}$$

Based on Theorem 1 in [76], if $\gamma^*(x)$ is considered as an optimal γ such that

$$x^T P(\gamma^*(x))x = \min_{\gamma \in \Gamma} x^T p(\gamma)x \tag{5.21}$$

then

$$\frac{\partial V_c}{\partial x} = 2P(\gamma^*(x))x \tag{5.22}$$

The feedback control u is defined as:

$$u = K(\gamma^*(x))x \tag{5.23}$$

The time derivative of Lyapunov function is obtained as:

$$V_c(x) = 2x^T P(\gamma^*(x))(A\alpha(t) + BK(\gamma^*(x)))x \tag{5.24}$$

The following optimization problem which is composed in [77] satisfies the Lyapunov conditions and minimizes the cost function in Eq. (5.16).

$$\begin{aligned} &\min_{p(\gamma),K(\gamma)} tr(P(\gamma)) \\ &s.t. \{(A\alpha) + BK(\gamma))^T p(\gamma) + P(\gamma)A\alpha + BK(\gamma))\} < -Q - K(\gamma)^T RK(\gamma) \end{aligned} \tag{5.25}$$

A new matrix $V(\gamma)$ can be defined such that $V(\gamma) = K(\gamma)F(\gamma)$. By applying Schur compliment, an equivalent optimization problem will be as follows:

$$\begin{aligned} &\max_{F(\gamma),V(\gamma)} tr(F(\gamma)) \\ &s.t. = \begin{bmatrix} \{-A(\alpha)F(\gamma) + BL(\gamma) - A(\alpha)F(\gamma) + BL(\gamma)^T\} & * & * \\ F(\gamma) & Q^-1 & * \\ V(\gamma) & 0 & R^-1 \end{bmatrix} > 0 \end{aligned} \tag{5.26}$$

By varying the parameters of the system, the above Bilinear Matrix Inequality (BMI) is transformed to Linear Matrix Inequality (LMI) and solved for each instant of α i.e. F and V at each instant of γ. Finally the feedback gain is achieved as follows:

$$K(\gamma) = V(\gamma)F(\gamma)^{-1} \tag{5.27}$$

Finally, the LPV controller gains $K_1, ..., K_4$ are computed as follows:

$$K_1 = \begin{bmatrix} 23.17 & 23.17 & -16.87 & 15.80 & 15.80 & -35.28 & -0.00 & 35.24 & -24.14 & -89.04 & 89.04 & -49.95 \\ -23.17 & 23.17 & -16.87 & -15.80 & 15.80 & -35.28 & -35.24 & 0.00 & 24.14 & -89.04 & -89.04 & 49.95 \\ -23.17 & -23.17 & -16.87 & -15.80 & -15.80 & -35.28 & -0.00 & -35.24 & -24.14 & 89.04 & -89.04 & -49.95 \\ 23.17 & -23.17 & -16.87 & 15.80 & -15.80 & -35.28 & 35.24 & -0.00 & 24.14 & 89.04 & 89.04 & 49.95 \end{bmatrix} \tag{5.28}$$

$$K_2 = \begin{bmatrix} 23.17 & -23.17 & -16.87 & 15.80 & -15.80 & -35.28 & -0.00 & 35.24 & -24.14 & 89.04 & 89.04 & -49.95 \\ 23.17 & 23.17 & -16.87 & 15.80 & 15.80 & -35.28 & -35.24 & 0.00 & 24.14 & -89.04 & 89.04 & 49.95 \\ -23.17 & 23.17 & -16.87 & -15.80 & 15.80 & -35.28 & 0.00 & -35.24 & -24.14 & -89.04 & -89.04 & -49.95 \\ -23.17 & -23.17 & -16.87 & -15.80 & -15.80 & -35.28 & 35.24 & -0.00 & 24.14 & 89.04 & -89.04 & 49.95 \end{bmatrix} \tag{5.29}$$

$$K_3 = \begin{bmatrix} -23.17 & 23.17 & -16.87 & -15.80 & 15.80 & -35.28 & -0.00 & 35.24 & -24.14 & -89.04 & -89.04 & -49.95 \\ -23.17 & -23.17 & -16.87 & -15.80 & -15.80 & -35.28 & -35.24 & 0.00 & 24.14 & 89.04 & -89.04 & 49.95 \\ 23.17 & -23.17 & -16.87 & 15.80 & -15.80 & -35.28 & -0.00 & -35.24 & -24.14 & 89.04 & 89.04 & -49.95 \\ 23.17 & 23.17 & -16.87 & 15.80 & 15.80 & -35.28 & 35.24 & -0.00 & 24.14 & -89.04 & 89.04 & 49.95 \end{bmatrix} \tag{5.30}$$

$$K_4 = \begin{bmatrix} -23.17 & -23.17 & -16.87 & -15.80 & -15.80 & -35.28 & -0.00 & 35.24 & -24.14 & 89.04 & -89.04 & -49.95 \\ 23.17 & -23.17 & -16.87 & 15.80 & -15.80 & -35.28 & -35.24 & -0.00 & 24.14 & 89.04 & 89.04 & 49.95 \\ 23.17 & 23.17 & -16.87 & 15.80 & 15.80 & -35.28 & -0.00 & -35.24 & -24.14 & -89.04 & 89.04 & -49.95 \\ -23.17 & 23.17 & -16.87 & -15.80 & 15.80 & -35.28 & 35.24 & -0.00 & 24.14 & -89.04 & -89.04 & 49.95 \end{bmatrix} \tag{5.31}$$

5.3 H∞ Self Gain-Scheduling Linear Parameter Varying Control

In this section the $H\infty$ self gain-scheduling control technique is used to obtain the LPV controller. In modern control theory, $H\infty$ methods are mainly used to achieve stabilization with guaranteed performance. In order to synthesize the $H\infty$ controller, the control problem shall be formed into a mathematical optimization problem and to solve the optimization for determining the controller. This control technique mainly can be used for the systems containing multivariate with cross-coupling characteristics. However, to design such a controller a very high level of mathematical understanding is needed to apply them successfully. Also having an accurate model of the plant is essential for a good controller performance. Such advantages for $H\infty$ control technique bring a strong motivation of designing the LPV control technique based on $H\infty$ technique. In fact, the natural expansion of the $H\infty$ control technique to gain-scheduling leads into LPV control design which guarantees the stability and performance of the designed LPV controller.

In this method the linearized equations of motion of the quadrotor is used to synthesize the LPV controller. Unlike the previous method the yaw angle is not playing any role as the varying parameter, but the state matrix is the function of roll and pitch angles. This way of parameter selection brings broader flight envelop to quadrotor and makes the controller possible to control the forward flight as well. It should be noted that for each varying parameter a certain operating range is determined. However, it is very important to define the range for varying parameters properly in order not to overload the calculation burden of the controller. On the other hand, increasing the safety margin for quadrotors manoeuvrability is tightly coupled with the range selection of the varying parameters. The selected range (parameter space) for each of the two varying parameters is: [-0.4 0.4], [-0.4 0.4] with discretized sampling grid of 300×300 respectively. To synthesize the LPV controller for this method, the Bounded Real Lemma (BRL) with the notion of quadratic $H\infty$

performance γ is used as the main tool. For a given LTI system with state space realization $G(s) = D + C(sI - A)^{-1}B$, the BRL map for a symmetric matrix P and positive scalar γ can be written from [77] and [78] as follows:

$$B^S_{[A,B,C,D]}(P,\gamma) = \begin{pmatrix} A^TP + PA & PB & C^T \\ B^TP & -\gamma I & D^T \\ C & D & -\gamma I \end{pmatrix} \tag{5.32}$$

Given a closed-loop LPV system having state space matrices $A(\rho)$, $B(\rho)$, $C(\rho)$ and $D(\rho)$, the system has quadratic $H\infty$ performance γ if and only if there exists a single positive definite matrix P such that:

$$\begin{pmatrix} A^T(\rho)P + PA & PB(\rho) & C^T(\rho) \\ B^T(\rho)P & -\gamma I & D^T(\rho) \\ C(\rho) & D(\rho) & -\gamma I \end{pmatrix} < 0 \tag{5.33}$$

is admissible for all values of the parameter vector ρ. Then the Lyapunov function $V(x) = x^TPx$ can be applied to initiate the global (asymptotic) stability and the L_2 gain is bounded by γ between the input and the output that is,

$$\|y\|_2 < \gamma \|u\|_2 \tag{5.34}$$

is applicable for all possible trajectories of ρ and the u is control input vector. Therefore the $H\infty$ performance requires the existence of a fixed quadratic Lyapunov function for the entire operating range [78]. The estimation of the controller is based on solving the LMI presented in Eq. (5.33) by using the concept of affine Polytopic LPV in [79]. Finally, by using the ”hinfgs” tool in MATLAB, the LPV controller gain $K(\rho(t))$ can be obtained as follows:

$$K(\rho(t)) = \sum_{i=1}^{5}\sum_{i=1}^{5} \omega_{1,i}\omega_{2,j} \times K_r \tag{5.35}$$

where

$$K_r = \begin{pmatrix} A_{k_{i,j}} & B_{k_{i,j}} \\ C_{k_{i,j}} & D_{k_{i,j}} \end{pmatrix} \tag{5.36}$$

and $\omega_{1,i}$ and $\omega_{2,j}$ are the two weighting functions for interpolation of LPV controller gains with 25 (5×5) LTI vertices in a polytopic LPV form and the LPV parameter dependent LPV controller is defined as:

$$\begin{aligned} \dot{x}_K &= A(\rho(t))x_K + B(\rho(t))e \\ u &= C(\rho(t))x_K + D(\rho(t))e \end{aligned} \tag{5.37}$$

where u is the control input and e is the error between reference and the real output signals. As it can be seen from Eq. (5.37), all matrices are parameter dependent and in the gain-scheduling process, both plant and the controller interpolate automatically by getting updated based on operating conditions. This ensures the stability, performance and the robustness of the system in the defined range of varying parameter [78].

5.4 Simulation Results

In this section the results for both applied methods are presented. For the first method the quadrotor set-point is set to [1 1 1 0 0 180] for X, Y, Z, ϕ, θ and ψ respectively. As it can be seen, all six parameters are following the reference accurately.

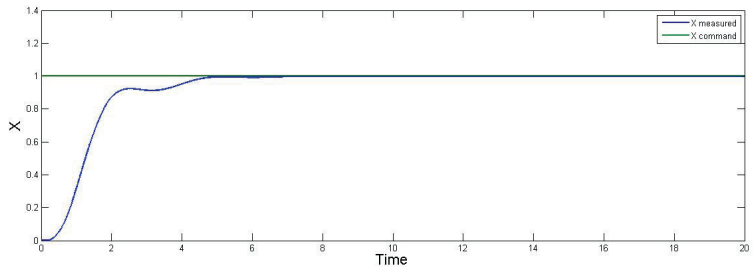

Figure 5.2: X position control based on optimal state feedback method

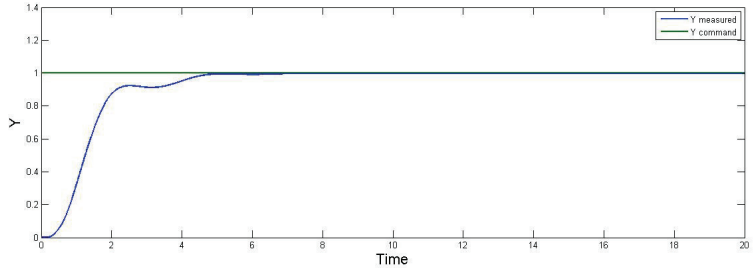

Figure 5.3: Y position control based on optimal state feedback method

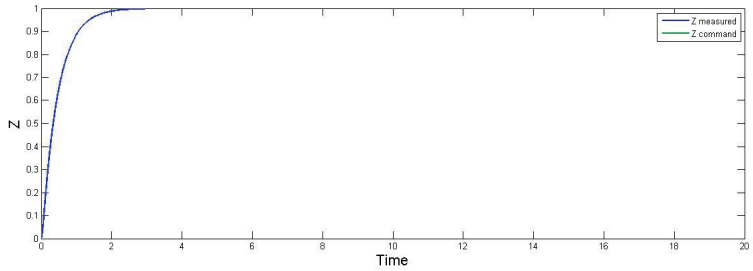

Figure 5.4: Z position control based on optimal state feedback method

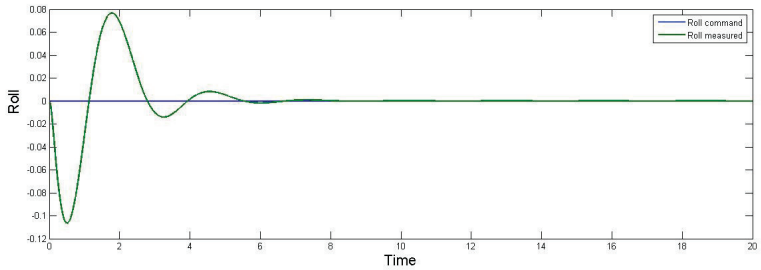

Figure 5.5: Roll control based on optimal state feedback method

In the second method where the gain-scheduling $H\infty$ LPV controller is used, the same parameters are used to demonstrate the performance of the controller.

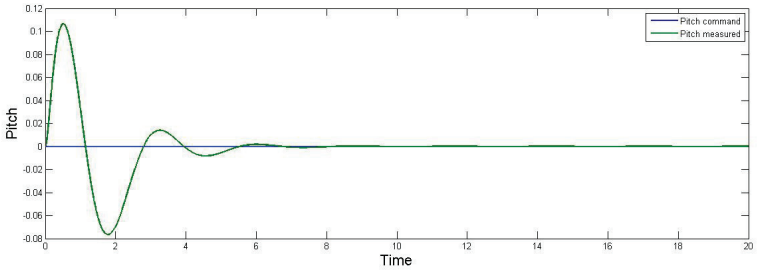

Figure 5.6: Pitch control based on optimal state feedback method

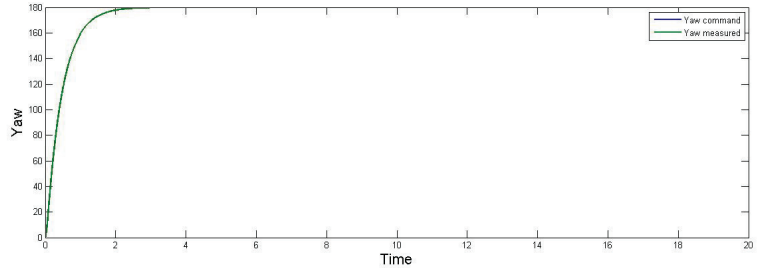

Figure 5.7: Yaw control based on optimal state feedback method

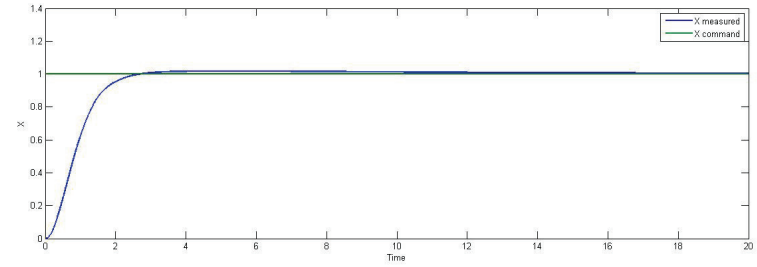

Figure 5.8: X position control using H∞-based LPV method

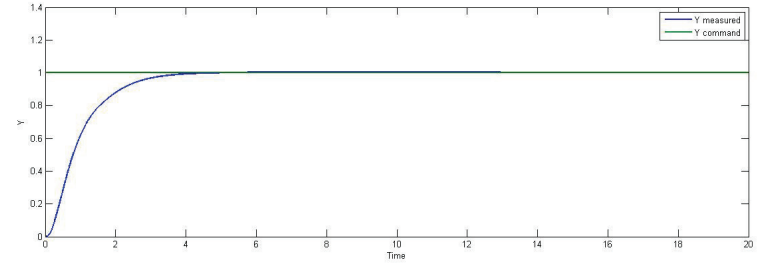

Figure 5.9: Y position control using H∞-based LPV method

For the second method, the set-point for yaw angle is set to zero since the yaw angle is not considered as the varying parameter and it can be varied in any selected range. By comparing the result of both techniques it can be concluded that both controllers are

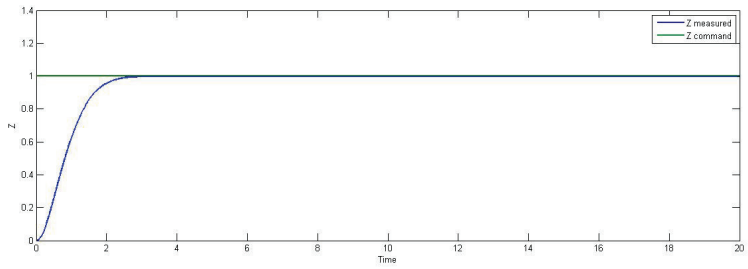

Figure 5.10: Z position control using H∞-based LPV method

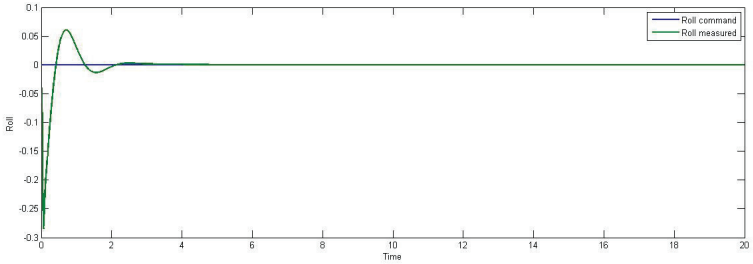

Figure 5.11: Roll control using H∞-based LPV method

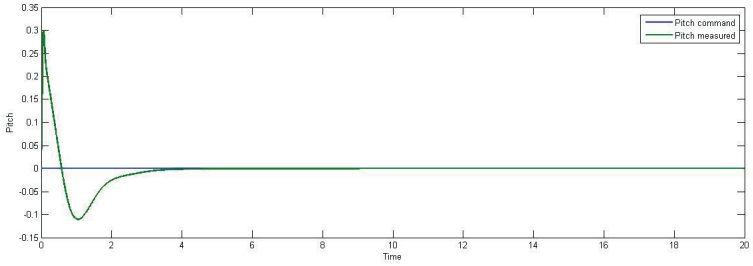

Figure 5.12: Pitch control using H∞-based LPV method

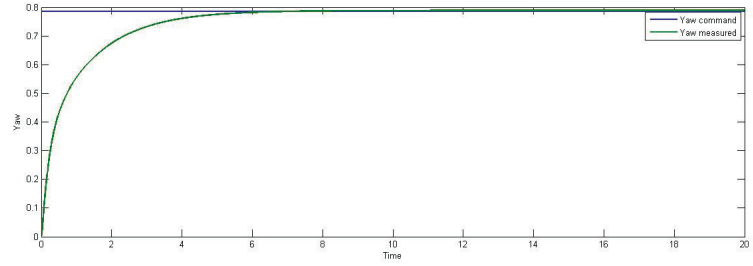

Figure 5.13: Yaw control using H∞-based LPV method

performing very promising for all assigned parameters to be controlled. However, the H∞ method showed to be more accurate and robust specially in the control of X and Y positions. This accurate performance can be due to increased number of LTIs considered in

control synthesis of the LPV controller comparing state feedback optimal control. In other words, LTIs (control regions) which shape the vertices of the LPV polytopic hull, help the controller to switch in more accurate manner among the designed local controllers. On the other hand, it should be mentioned that a huge number of LTIs (polytopic vertices) brings a huge burden that can be considered as a drawback to this performance augmentation. The number of LTIs should be in balance with the computation power of the controller.

The LPV control technique showed to be a proper alternative to most of approaches based on gain-scheduling technique such as GS-PID in terms of robustness, stability and performance. However, using the LPV control technique as a fault-tolerant controller is still challenging since the fault can bring infeasibility to LMIs and the convexity of the operational range. On the other hand, from the real flight-test point of view, since the LTIs can not be decreased to less than certain numbers due to stability and performance degradation of the system, the onboard computer should be powerful enough to handle the burden of the control algorithm. In the Qball-X4 case, the Gumstix onboard computer was not able to handle any reduced size of the LPV controller. This can be considered as a challenging potential future work.

The main motivation for using the LPV technique was to develop this control technique as a further extension of gain scheduling method for potential applications of UAV testbeds investigated in this thesis with a goal for achieving guaranteed stability and performance. However, due to the above mentioned reasons it can be concluded that LPV can be considered as a challenging technique for such a purpose.

As it can be understood from the literature, this LPV controller design approach "is well known to only provide rigorous stability and performance guarantee for sufficiently slow parameter variation" [74]. Also as stated in [72] "while theoretically well founded and despite its introduction over 25 years ago, the LPV methodology appears to be still not widely used in industrial applications. It is also stated that LPV methods are difficult to apply to plants of industrial complexity due to considerable computational burden". These

descriptions can justify the failure of LPV control technique for agile systems such as quadrotor UAV specially for FTC purposes since the quick varying parameters either in the quadrotor dynamics in the case of fault/failure occurrence is in contrast with slow variation of varying parameter in the LPV system.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis some important key issues of fault-tolerant flight control of unmanned aerial vehicles (UAVs) are presented. A comprehensive literature review is provided and several control techniques have been applied for fault-free and faulty scenarios. The PID, GS-PID, MRAC and CA/RA are mainly used for FTC and the LPV is used for fault-free case of quadrotor UAV and the results are presented at the end of each chapter. All of the results in this thesis are based on the faults/failures occurred only in actuators. The sensor or the telecommunication faults/failures were not considered although they have the same level of importance for FTC systems.

Finally, some hardware works toward the fixed/rotary-wing UAV testbed development have been conducted in the NAVLAB of the Department of Mechanical and Industrial Engineering of Concordia University. A list of such hardwares along with a brief description of their characteristics is presented in Appendix A of this thesis.

6.2 Future Work

Since this thesis is mainly UAV application-oriented, the future work can be considered based on the available rotary-wing and fixed-wine UAV testbeds.

As described in Chapter 3, the GS-PID techniques is used for the HK Bixler UAV for actual fault in rudder (stuck rudder). On the other hand, the Bixler-2 UAV (described in Appendix 1) is equipped with wing split mechanism that makes it a proper testbed for wing damage scenario. The same GS-PID controller can be used for this purpose. Some other scenarios such as engine failure and aileron stuck also can be applied to this testbed. In terms of control technique, the idea of control allocation and re-allocation can be used for handling the same fault scenarios based on PID technique since the mathematical model of HK Bixler is not currently available.

In terms of rotary-wing helicopter UAVs, the Qball-X8 quadrotor can be used for more FTC techniques which require actuator redundancy. Also the Align T-rex 800 single rotor helicopter UAV has the potential of being used for tail-rotor or the swash-plate fault scenarios as a potential future work.

Although it was originally planned for investigating the feasibility of using LPV technique to FTC applications of the Qball-X4 UAV, it was not successful due to the numerical infeasibility issue of the problem formulation in a LPV framework when actuator faults are introduced. As another future work of this thesis, it is worthwhile for further investigation of this control technique based on the advantages and disadvantages as mentioned earlier in the thesis.

Appendix A

UAV Testbeds Development

As mentioned in Chapter 2, there are few publications for real flight-tests for the fixed-wing UAV, since fixed-wing UAVs are more challenging to compare with rotary-wing UAVs in terms of required skills and outdoor flight-test facilities for the operation. This brings more demand for building and development of fixed-wing UAV platforms.

On the other hand, since the quadrotor helicopters are so-called "underactuated" systems, the need of more actuators (redundant rotors) is essential for some of the control algorithms that requires overactuated systems such as CA/RA techniques. Hence, the development and flight-test of an octocopter was inevitable. For this reason at the NAVLAB, one of the Qball-X4 quadrotors has been converted first to MQ-X4 with six actuators as described in section 4.3.1 and later upgraded to an octocopter known as Qball-X8 with eight rotors.

In this section, the focus is on brief illustration of some hardware works that have been accomplished toward development and tests of fixed-wing and rotary-wing UAVs to make them suitable and reliable testbeds for other researchers and students in the NAVLAB. Fig. A.1 shows some of the currently developed UAVs along with available UGVs in the NAVLAB.

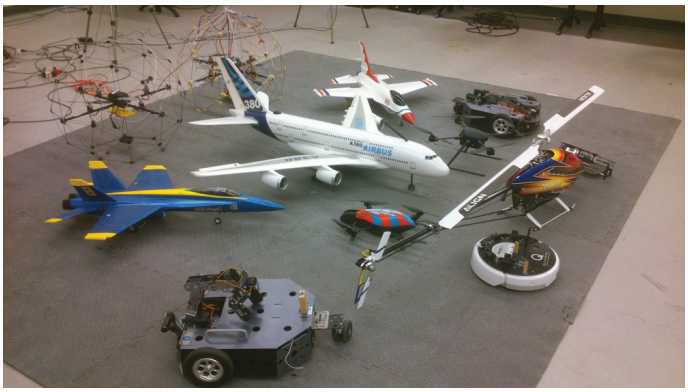

Figure A.1: NAVLAB UAV/UGV testbeds

A.1 Airbus A380 Testbed

For the need of subscale flight test of UAVs for FTFC purposes, a subscale Airbus A380 (see Fig. A.2) is built and flight-tested. The A380 is made of EPO material with a wingspan of 1520 mm and four 56 mm ducted fans connected to four powerful 25A brushless EDF system. The A380 is powered by 5,000 mAh 3S Li-Po battery. The plane is equipped with MP2128H autopilot system by MicroPilot Inc. The airplane performance characteristics as well as handling qualities showed to be very efficient in all phases of the flight. The specifications of this airframe are presented as follows:

- Wingspan: 1520 mm
- Fuselage Length: 1410 mm
- Flying Weight: 1800 g
- Motor: 4× 2820-1300kv (with 56 mm EDF)
- Battery: 5000 mAh 3 cell Li-Po
- Speed Controller: 4× 25A

The MicroPilot MP2128H autopilot is a highly functional high speed UAVs through backpack UAVs to handheld micro UAVs. The MicroPilot Autopilot is capable for fully

Figure A.2: Airbus A380 fixed-wing UAV testbed

autonomous operation from launch through recovery. Capabilities include airspeed hold, altitude hold, turn coordination, GPS navigation as well as autonomous launch and recovery. Extensive data logging and manual overrides are also supported, as is a highly functional command buffer. All feedback loop gains and flight parameters are user programmable and feedback loops are adjustable in flight. The MP2128H autopilot integrated to Airbus A380 UAV is shown in Fig. A.3.

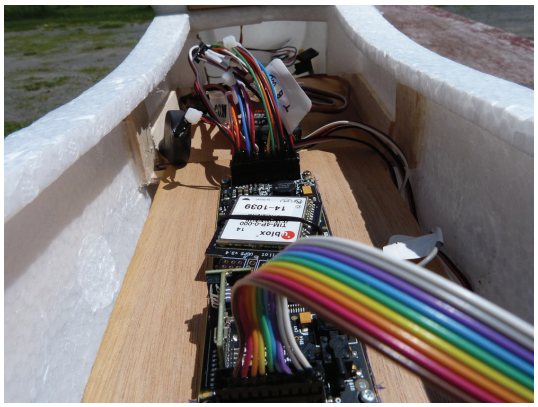

Figure A.3: MP2128H autopilot installed on A380 UAV

The MicroPilot MP2128H autopilot also includes the HORIZON ground control software which is shown in Fig. A.4 for mission creation, parameter adjustment, flight monitoring and mission simulation. This autopilot has many useful features and capabilities such as:

- 1,000 programmable waypoints or commands

- Controls up to 24 servos or relays
- Autonomous recovery methods include runway landing, parachute recovery, and deep stall landing
- Supports flaps, flaperons, elevons, v-tail, x-tail, split rudders, split ailerons, and flap/aileron mixing
- User programmable error handlers for loss of GPS signal, loss of RC signal, engine failure, loss of data link, and low battery voltage, etc. All these capabilities makes MP2128H a suitable autopilot for fault-tolerant flight control of both fixed-wing and rotary-wing UAVs

Figure A.4: Horizon ground control station software for MP2128H

The MP2128H autopilot is equipped with pitot tube and the airspeed sensor, two way telecommunication system for real-time data telemetry, sonar module and its sensor for auto take-off and auto landing and finally the high precision UBLOX GPG module and antenna.

A.2 HK Bixler-2 UAV Testbed

As shown in Chapter 3, the HK Bixler R/C airplane has been upgraded to a fully autonomous UAV with the help ArduPilot APM Mega 2.5 autopilot. A more recent version

of this R/C airplane known as Bixler-2 is assembled and flight-tested for the NAVLAB. This airframe is shown in Fig. A.5.

Figure A.5: Bixler 2 UAV with the split wing mechanism

The platform is equipped with the same APM 2.5 autopilot in order to be used for aerial surveillance, UAV formation flight control and finally for the purpose of FTFC. For the latter case, a split wing mechanism is designed and installed on the right wing for UAV wing damage scenarios. The damage percentage can be changed based on FTC scenario requirements.

The following airplane's specifications show the differences between the HK Bixler UAV which has been described in Chapter 3, with updated HK Bixler-2 platform. However, as mentioned before, the autopilot of this platform is the same as HK Bixler UAV in Chapter 3.

- Wingspan: 1500 mm
- Fuselage Length: 963 mm
- Flying Weight: 760g
- Motor: 2620-1300kv
- Battery: 2600 mAh 3 cell Li-Po
- ESC: 20A

A.3 Align T-rex 800 Single Rotor Helicopter Testbed

The Align T-rex 800 is a single rotor helicopter designed and manufactured specifically for aerial photography and cinematography by Align company, in Taiwan. The long 800 mm main blades provides superior flight stability and high payload capacity. The helicopter uses direct flight control flybarless (DFC-FBL) main rotor system, the extreme low CG design effectively lowers aerodynamic resistance and improves flight stability characteristics. The carbon fiber main frame plates as well as the tail boom, makes the airframe very durable for different purposes of use including acrobatic flights, payload carrying, etc.

Figure A.6: Align T-rex 800 single-rotor helicopter testbed

Figure A.7: Align G800 aerial gimbal system

Also the camera gimbal 3X G800 from the same manufacturer has been purchased and assembled to be used as a reliable platform for mounting up to 2 *kg* camera for aerial surveillance, photography and cinematography. In Fig. A.7, this camera gimbal platform is shown.

The gimbal is equipped with GS800 gimbal controller with high power computing capability, and various control modes to precisely lock into the shooting object, as well as meet the needs of various other aerial photography needs, allowing optimal imagery to be obtained easily [83]. The T-rex 800 anlong with the gimbal camera system can be converted to full autonomous UAV using the MP2128H or APM 2.5 autopilot units. The specifications of this UAV platform are presented as follows:

- Length: 1490 mm

- Height: 400 mm
- Main Blade Length: 800 mm
- Main Rotor Diameter: 1780 mm
- Tail Rotor Diameter: 310 mm
- Flying Weight: 5400 g
- Motor: 750MX brushless 450 KV
- ESC: Castel ICE2 HV 120 A
- Battery: 2×6S Li-Po 5200 mAh

A.4 Align T-rex 450 Sigle Rotor Helicopter Testbed

One of the main research interest in the field of UAVs is System Identification (SI). One of the main challenges for such a purpose is to collect flight data directly from the flying vehicle since the results obtained from flight simulator can be far from the real values recorded by a UAV. For this purpose, an Align T-rex 450 single rotor helicopter which has a high manoeuvrability capabilities is converted to a UAV by building and installing a special autopilot platform along with well-mounted ArduPilot APM 2.5 autopilot unit and its equipments as shown in Fig. A.8.

As it can be seen from Fig. A.8, the UAV is smaller and lighter than the Align T-rex 800 UAV which bring more agility to this platform and makes it very suitable for SI required maneuvers such as frequenct sweep, doublet, 3211 etc. This platform also can be used for FTFC purposes. The specifications of this platform are presented as follows:

- Length: 634 mm
- Height: 205 mm

Figure A.8: T-rex 450 single rotor helicopter built for SI purposes

- Main Blade Length: 450 mm
- Main Rotor Diameter: 715 mm
- Tail Rotor Diameter: 158 mm
- Flying Weight: 584 g
- Motor: 450MX Brushless Motor (3400 KV)
- ESC: Focus Shot 15, 35 A
- Battery: 3S 11.1V 2200 mAh Li-Po

A.5 F-18 and F-16 Subscale UAV Platforms

Apart from the following UAV testbed developments, two EDF R/C models (the F-16 and F-18) models have been assembled and flight tested. These airframes have been equipped with APM 2.5 autopilot which makes them capable of full autonomous UAVs. Both planes are powered by 30A ESCs and ducted fan brushless powerplant system. The F-18 specifications are as follows:

- Wing Span: 686mm

- Length: 1032mm
- Height: 257mm
- Flying Weight: 640g
- Motor: B2025 Brushless Inrunner (4300KV) with EDF 64mm
- ESC: 30A
- Battery: 1300mAh 3S Li-po

Figure A.9: F-18 UAV testbed

Figure A.10: F-16 UAV testbed

The specifications of the F-16 are as follows:

- Wing span: 1030 mm
- Fuselage Length: 840 mm
- Weight: 650 g
- ESC : 30A with BEC
- Motor: Brushless outrunner 2826/3200KV with 70 mm EDF unit
- Battery: 3S 1600 mAH

These UAV testbeds can be used for damage-tolerant control as well as the scenarios which require high speed airspeed. The low drag aerodynamical shape of these testbeds makes them able to fly at high speed.

Bibliography

[1] ICAO safety report, available online at http://www.icao.int/safety/Documents/ICAO_2013-Safety-Report_FINAL.pdf,2013 (Accessed on 2014-04-11).

[2] "Final report on the accident on 1st June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France flight AF 447 Rio de Janeiro-Paris," Bureau d'Enquetes et d'Analyses pour la securite de l'aviation civile, July 2012, pp. 1-223.

[3] "Interim report no. 3: on the accident on 1 June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France flight AF 447 Rio de Janeiro-Paris," Bureau d'Enquetes et d'Analyses pour la securite de l'aviation civile, 29 July 2011.

[4] C. Nicola, "Report on Air France crash points to pilot training issues," The New York Times, 29 July 2011.

[5] http://www.uxvuniversity.com/careers.

[6] available online at http://www.atlasaviation.com/AviationLibrary (Accessed on 2014-03-8).

[7] G. Vachtsevanos, T. Liang, G. Drozeski, and L. Gutierrez, "From mission planning to flight control of unmanned aerial vehicles: strategies and implementation tools," Annual Reviews in Control, Vol 29, 2005, pp. 101-115.

[8] American Institute of Aeronautics and Astronautics, "Worldwide UAV roundup," Available online at http://www.aiaa.org/images/PDF/WilsonChart.pdf,2007 (Accessed on 2014-04-11).

[9] Office of the Secretary of Defense, "Unmanned aircraft systems roadmap," available online http://www.acq.osd.mil/usd/RoadmapFinal2,2005 (Accessed on 2014-04-11).

[10] K. Keerti, D.U. Bhamidipati, and N. Neogi, "Engineering safety and reliability into UAV systems: mitigating the ground impact hazard," University of Illinois, Urbana-Champaign, Urbana, IL, 2008.

[11] http://www.chron.com/neighborhood/woodlands/article/250K-police-drone-crashes-into-Lake-Conroe-5435343.php#photo-1711069 (Accessed on 2014-04-11).

[12] http://www.cbc.ca/news/canada/british-columbia/drone-crash-prompts-vancouver-to-review-film-industry-use-1.2671977 (Accessed on 2014-10-10).

[13] http://www.uasvision.com/2014/07/09/electrical-malfunction-caused\-global-hawk-crash/?utm_source=Newsletter&utm_medium=email&utm_campaign=e295980635-RSS_EMAIL_CAMPAIGN&utm_term=0_799756aeb7-e295980635-297548553 (Accessed on 2014-10-5).

[14] http://www.reddit.com/user/jonsimo (Accessed on 2014-02-21).

[15] Z.T. Dydek, and M.A. Anuradha, "Adaptive control of quadrotor UAVs in the presence of actuator uncertainties," AIAA Infotech@Aerospace 2010, 20-22 April 2010, Atlanta, Georgia.

[16] A. Freddi, S. Longhi, and A. Monteriu, "A Model-based fault diagnosis system for a mini-quadrotor," 7th workshop on Advanced control and Diagnosis, 19-20 November, 2009, Zielona Gora, Poland.

[17] A. Bani Milhim, Y.M. Zhang, and C.A. Rabbath, "Gain scheduling based PID controller for fault tolerant control of a quadrotor UAV," AIAA Infotech@Aerospace 2010 20 - 22 April 2010, Atlanta, Georgia.

[18] M.H. Amoozgar, A. Chamseddine, and Y.M. Zhang, "Fault-tolerant fuzzy gain-scheduled PID for a quadrotor helicopter testbed in the presence of actuator faults," in Proc. of IFAC Conference on Advances in PID Control, March 28-30, 2012, Brescia, Italy.

[19] I. Sadeghzadeh, A. Mehta, Y.M. Zhang, and C.A. Rabbath, "Fault-tolerant trajectory tracking control of quadrotor helicopter using gain-scheduled PID and model reference adaptive control," Annual Conference of the Prognostics and Health Management Society 2011, September 25-29, 2011, Montreal, Quebec, Canada.

[20] A. Chamseddine, Y.M. Zhang, C.A. Rabbath, C. Fulford, and J. Apkarian, "Model reference adaptive fault tolerant control of a quadrotor UAV," AIAA InfoTech@Aerospace 2011: Unleashing Unmanned Systems, March 29-31, 2011, St. Louis, Missouri, USA.

[21] F. Sharifi, M. Mirzaei, B.W. Gordon, and Y.M. Zhang, "fault tolerant control of quadrotor using sliding mode control," Proc. of the Int. Conference on Control and Fault-Tolerant Systems (SysTol'10), October 6-8, 2010, Nice, France.

[22] T. Li, Y.M. Zhang, and B.W. Gordon, "Investigation, flight testing, and comparison of three nonlinear control techniques with application to a quadrotor unmanned aerial vehicle," AIAA Guidance, Navigation, and Control Conference, 13-16 August, 2012, Minneapolis, Minnesota, USA.

[23] A. Chamseddine, D. Theilliol, D. Sauter, and Y.M. Zhang, "Flatness-based trajectory planning to improve actuator fault-tolerance capability," Mediterranean Conference of Complex Systems Engineering, 27-28 May, 2011, Agadir, Morocco.

[24] E.N Johnson, and D.P Schrage, "System integration and operation of a research unmanned aerial vehicle," Journal of Aerospace Computing, Information, and Communication Vol. 1, January 2004.

[25] G. Drozeski, S. Bhaskar, and G. Vachtsevanos, "Fault detection and reconfigurable control architecture for unmanned aerial vehicles," in Proceedings of the IEEE Aerospace Conference, March 5-12, 2005, Big Sky, MT, USA.

[26] G. Drozeski, S. Bhaskar, and G. Vachtsevanos, "Fault tolerant architecture for an unmanned rotorcraft," in Proceedings of the AHS International Specialists' Meeting on Unmanned Rotorcraft, January 18-20, 2005, Chandler, AZ, USA.

[27] G. Drozeski, and G. Vachtsevanos, "Fault-tolerant architecture with reconfigurable path planning applied to an unmanned rotorcraft," in Proceedings of the American Helicopter Society 61st Annual Forum, June 1-3, 2005, Grapeview, TX, USA.

[28] D.B. Jourdan, M.D. Piedmonte, V. Gavrilets, and D.W. Vos, "Enhancing UAV survivability through damage tolerant control," AIAA Guidance, Navigation, and Control Conference, August 2-5, 2010, Toronto, Ontario, Canada.

[29] G. Chowdhary, W.M. DeBusk, and E.N. Johnson, "Real-time system identification of a small multi-engine aircraft with structural damage," Georgia Institute of Technology, Atlanta, GA, USA.

[30] http://www.armytimes.com/article/20080705/NEWS/807050307/Fighter-flies-most-wing-gone (Accessed on 2014-09-03).

[31] E.N. Johnson, G.V. Chowdhary, and M.S. Kimbrell, “Guidance and control of an airplane under severe structural damage,” AIAA Infotech Aerospace, April 20-22, 2010, Atlanta, Georgia, USA.

[32] R. Isermann, “Fault-diagnosis systems, an introduction from fault detection to fault tolerance,” Springer-Verlag, Berlin Heidelberg, 2006.

[33] J.J. Guillaume Ducard, “Fault-tolerant flight control and guidance systems: practical methods for unmanned aerial vehicles,” Springer, 2009.

[34] G. Chowdhary, W.M. DeBusk, and E.N. Johnson, “Real-time system identification of a small multi-engine aircraft with structural damage,” Georgia Institute of Technology, Atlanta, GA, USA.

[35] Y.M. Zhang, and J. Jiang, “Bibliographical review on reconfigurable fault-tolerant control systems,” Annual Reviews in Control,” Vol. 32, No. 2, 2008, pp. 229-252.

[36] R.J. Adams, “Robust Multivariable Flight Control,” Springer-Verlag, 1994.

[37] J.F. Magni, S. Bennani, and J. Terlouw, “Robust flight control: a design challenge,” Springer-Verlag, London, 1997. In Lecture Notes in Control and Information Sciences, 224.

[38] S.I. Al Swailem, “Application of robust control in unmanned vehicle flight control system design,” PhD thesis, Cranfield University, 2004.

[39] L.D. Desborough, R.M. Miller,“Increasing customer value of industrial control performance monitoring Honeywells experience,” Chemical Process Control-VI, Tuscon, Arizona, USA, Jan. 2001, Vol 98.

[40] B.T. Whitehead, and S.R. Bieniawskiy, “Model reference adaptive control of a quadrotor UAV,” AIAA Guidance, Navigation, and Control Conference, 2-5 August 2010, Toronto, Ontario, Canada.

[41] Z.T. Dydek, and A.M. Annaswamy, "Combined/composite adaptive control of a quadrotor UAV in the presence of actuator uncertainty," AIAA Guidance, Navigation, and Control Conference, 2-5 August 2010, Toronto, Ontario, Canada.

[42] K.J Astrom, and B. Wittenmark, "Adaptive control," Text book, Second Edition, Dover, 2008.

[43] W.C. Durham, "Constrained control allocation," Journal of Guidance, Control, and Dynamics, Vol 17, No 4, pp 717-725, 1993.

[44] Y.M. Zhang, V.S. Suresh, B. Jiang, and D. Theilliol, "Reconfigurable control allocation against aircraft control effector failures," In Proc. of IEEE Conf. on Control Applications, pp. 1197-1202, Singapore.

[45] H. Alwi, and C. Edwards, "Fault tolerant control using sliding modes with on-line control allocation," Automatica, Vol 7, No 44, pp. 1859-1866.

[46] W.J. Rugh, J.S. Shamma,"Research on gain scheduling," Automatica, Vol 36, No 10, pp. 1401-1425, 2000.

[47] B.M. Albaker and N.A. Rahim, "Flight path PID controller for propeller-driven fixed-wing unmanned aerial vehicles," in International Journal of the Physical Sciences Vol 6, No 8, pp. 1947-1964, 2011.

[48] D. Kingston, R. Beard1, T. McLain, M. Larsen, W. Ren, "Autonomous vehicle technologies for small fixed-wing UAVs," in AIAA-2003-6559.

[49] M. Ahsan, K. Rafique, and F. Mazhar, "Optimization based tuning of autopilot gains for a fixed-wing UAV," World Academy of Science, Engineering and Technology Vol 7, 2013.

[50] B. Kada, Y. Ghazzawi, "Robust PID controller design for an UAV flight control system," in Proceedings of the World Congress on Engineering and Computer Science 2011 Vol II, WCECS 2011, October 19-21, 2011, San Francisco, USA.

[51] http://copter.ardupilot.com/wiki/apm25board (Accessed on 2014-04-11).

[52] http://www.quanser.com/products/qball (Accessed on 2013-6-17).

[53] I. Sadeghzadeh, A. Chamseddine, Y.M. Zhang, and D. Theilliol, "Control allocation and re-allocation for a modified quadrotor helicopter against actuator faults," presented in the 8th IFAC Safeprocess12, August 29-31, 2012, Mexico City, Mexico.

[54] http://www.naturalpoint.com/optitrack (Accessed on 2014-04-11).

[55] http://www.draganfly.com (Accessed on 2014-04-11).

[56] http://www.robotshop.com(Accessed on 2014-04-11).

[57] T. Johansen, T. Fossen, and P. Tondel, "Efficient optimal constrainted control allocation via multiparametric programming," Journal of Guidance, Control, and Dynamics, Vol 3, No 28, pp. 506-514.

[58] M. Bodson, "Evaluation of optimization methods for control allocation," Journal of Guidance, Control, and Dynamics, Vol 4, No 25, pp. 703-711.

[59] O. Harkegard, and S.T. Glad, "Resolving actuator redundancy-optimal control vs. control allocation," Automatica, Vol 41, No 1, pp. 137-144.

[60] R. Gadient, J. Levin, and E. Lavretsky, "Comparison of model reference adaptive controller designs applied to the NASA generic transport model," AIAA Guidance, Navigation, and Control Conference, 2-5 August 2010, Toronto, Ontario, Canada.

[61] J.X. Guo, G. Tao, and Y. Liu, "Multivariable adaptive control of NASA generic transport aircraft model with damage," Journal of Guidance, Control, and Dynamics, Vol 34, No 5, pp. 1495-1506, 2011.

[62] J.X. Guo, G. Tao, and Y. Liu, " Multivariable MRAC scheme with application to a nonlinear aircraft mode," Automatica, Vol 47, No 4, pp. 804-812, 2011.

[63] K.A. Lemon, J.E. Steck, B.T. Hinson, N. Nguyen and D. Kimball, "Model reference adaptive fight control adapted for general aviation: controller gain simulation and preliminary flight testing on a bonanza fly-by-wire testbed," AIAA Guidance, Navigation, and Control Conference 2 - 5 August 2010, Toronto, Ontario Canada.

[64] J. Priyank and M.J. Nigam, "Design of a model reference adaptive controller using modified MIT rule for a second order system," Journal of Advance in Electronic and Electric Engineering, Vol 3, No 3, pp. 477-484, 2013.

[65] P. Swarnkar, S. Jain and R.K. Nema, "Effect of adaptation gain in model reference adaptive controlled second order system," Vol 1, No 3, pp. 70-75, 2011.

[66] L. Zheng, "Model reference adaptive controller design based on fuzzy inference system," Journal of Information and Computational Science, Vol 9, No 8, pp. 1683-1693, 2011.

[67] T.N Nguyen, K. Krishnakumar, and J. Bokovic, "An optimal control modification to model-reference adaptive control for fast adaptation," AIAA Guidance, Navigation and Control Conference, 2008.

[68] N. Golea, A. Golea, and M. Kadjoudj, "Nonlinear model reference adaptive control using Takagi-Sugeno fuzzy systems," Journal of Intelligent & Fuzzy Systems, No 17, pp. 47-57, 2006.

[69] J.K. Lee, B.H. Suh, K. Abi, "Model reference adaptive control of nonlinear system using feedback linearization," SICE '95. Proceedings of the 34th SICE Annual Conference, pp. 1571-1576, 1995.

[70] I. Sadeghzadeh, A. Mehta, A. Chamseddine and Y.M. Zhang, "Active fault-tolerant control of a quadrotor UAV based on gain-scheduled PID control," The 25th IEEE Canadian Conference on Electrical and Computer Engineering, Montreal, Canada, April 29 - May 2, 2012.

[71] Soldiers Magazine (2007), http://findarticles.com/p/articles/mi_m0OXU/is_5_62/ai_n27236124/ (Accessed on 2012-12-23).

[72] C. Hoffmann, and H. Werner, "A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations," IEEE Transaction on Control Systems Technology, pp.1063-6536, 2014.

[73] P. Apkarian, P. Gahinet, "A convex characterization of gain-scheduled Hcontrollers," IEEE Transaction Automatic Control, Vol 40, No 5, pp. 853-854, 1995.

[74] J.S. Shamma and J.R. Cloutier, "Gain-scheduled missile autopilot design using linear parameter varying transformation," AIAA Guidance, Control and Dynamics, Vol 16, No 2, pp. 256-263, 1998.

[75] B. Bamieh and L. Giarr, "LPV models: identification for gain scheduling control," European Control Conference, Porto, Italy.

[76] T. Hu and Z. Lin, "Composite quadratic lyapunov functions for strained control systems," IEEE Transaction, Automatic Control, Vol 48, No 3, pp. 440-450, 2003.

[77] A. Serirojanakul and M. Wongsaisuwan, "Optimal control of quad-rotor helicopter using feedback LPV method," 9th International Conference onElectrical Engineering/Electronics, Computer, Telecommunications and Information Technology, pp. 1-4, 2012.

[78] P. Apkarian, P. Gahinet, and G. Becker, "Self-scheduled H_∞ control of linear parameter-varying systems: a design example," Automatica Vol 31, pp. 1251-1261.

[79] S.L.M.D. Rangajeeva and J.F. Whidborne, "Linear parameter varying control of a quadrotor," 6th International Conference on Industrial and Information Systems, ICIIS, Seri Lanka, 2011.

[80] S. Chumalee, and J.F. Whidbrone, "Gain-scheduled H1 autopilot design via parameter dependent lyapunov functions," In AIAA Guidance, Navigation and Control Conference, Chicago, Illinois.

[81] P. Apkarian, and R. Adams, "Advanced gain-scheduling techniques for uncertain system," IEEE Transactions on Control System Technology, Vol 6, No 1, pp. 21-32, 1998.

[82] G.J. Balas, "Linear parameter varying control and its application to aerospace systems," 23rd Congress of International Council of the Aeronautical Sciences, 8-13 September, 2002, Toronto, Canada, pp. 541.1-541.9.

[83] http://www.helipal.com/trex-800e-dfc-trekker-dfc-super-combo.html (Accessed on 2014-06-01).