AFRL-RB-WP-TR-2010-3109

STRUCTURAL TECHNOLOGY EVALUATION AND ANALYSIS PROGRAM (STEAP)

Delivery Order 0037: Prognosis-Based Control Reconfiguration for an Aircraft with Faulty Actuator to Enable Performance in a Degraded State

Vishal V. Mahulkar

Purdue University

DECEMBER 2010
Final Report

AIR FORCE RESEARCH LABORATORY
AIR VEHICLES DIRECTORATE
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the USAF 88th Air Base Wing (88 ABW) Public Affairs Office (PAO) and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RB-WP-TR-2010-3109 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH THE ASSIGNED DISTRIBUTION STATEMENT.

*//Signature//

MARK M. DERRISO
Project Engineer
Integrated Systems Branch

//Signature//

AARON M. KLOSTERMAN, Chief
Integrated Systems Branch
Structures Division

//Signature//

DAVID M. PRATT, PhD
Technical Advisor
Structures Division
Air Vehicles Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

*Disseminated copies will show "//Signature//" stamped or typed above the signature blocks.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
December 2010	Final	19 December 2008 – 31 December 2010

4. TITLE AND SUBTITLE
STRUCTURAL TECHNOLOGY EVALUATION AND ANALYSIS PROGRAM (STEAP)
Delivery Order 0037: Prognosis-Based Control Reconfiguration for an Aircraft with Faulty Actuator to Enable Performance in a Degraded State

5a. CONTRACT NUMBER
FA8650-04-D-3446-0037

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
0602201

6. AUTHOR(S)
Vishal V. Mahulkar

5d. PROJECT NUMBER
A0CA

5e. TASK NUMBER

5f. WORK UNIT NUMBER
A0CA0B

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

By:	For:
Purdue University	General Dynamics Information Technology, Inc. 5100 Springfield Pike, Suite 509 Dayton, OH 45431

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory
Air Vehicles Directorate
Wright-Patterson Air Force Base, OH 45433-7542
Air Force Materiel Command
United States Air Force

10. SPONSORING/MONITORING AGENCY ACRONYM(S)
AFRL/RBSI

11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)
AFRL-RB-WP-TR-2010-3109

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES
Report contains color. PAO Case Number: 88ABW-2010-6014; Clearance Date: 10 Nov 2010.

14. ABSTRACT
The objective of this work is to develop a framework for prognosis-based control reconfiguration of dynamic systems with faults. A conventional feedback control design for a process plant or vehicle system may result in unsatisfactory performance (even instability), in the event of malfunctions in actuators, sensors, or other components of the system. In order to overcome the limitations of conventional feedback, new controllers need to be developed which are capable of tolerating component malfunctions while still maintaining desirable and robust performance and stability properties. In cases where the malfunctions are faults, additional fault growth dynamics is introduced which leads to a continuously changing system response. The controller also needs to adapt to these changes while minimizing the rate of growth of the fault.

15. SUBJECT TERMS
control augmentation system, flight control system, fault detection and isolation, health monitoring, linear matrix inequalities, model predictive control, pseudo inverse method, Kalman filter

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT:	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON (Monitor)
a. REPORT	b. ABSTRACT	c. THIS PAGE			Mark M. Derriso
Unclassified	Unclassified	Unclassified	SAR	200	19b. TELEPHONE NUMBER *(Include Area Code)* N/A

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

SYMBOLS

Aircraft States

V_t	velocity of the aircraft with respect to the air
α	angle of attack
β	side slip angle
p	roll rate
q	pitch rate
r	yaw rate
ϕ	roll attitude
θ	pitch attitude
ψ	yaw attitude
q_1, q_2, q_3, q_4	quarternion orientation variables

Aircraft inputs

δ_t	thrust
δ_{lef}	leading edge flap angle
δ_e	elevator input
δ_a	aileron input
δ_r	rudder input

Hydraulic Actuator

m	mass
x_L	cylinder position
P_1	chamber one pressure
P_2	chamber two pressure
P_L	load pressure

b	viscous damping
F_{fc}	modelled Coulomb friction force
$\tilde{f}$	external disturbances
V_t	total hydraulic volume
β_e	oil bulk modulus
A	cylinder ram area
Q_L	load flow rate
C_d	valve orifice coefficient of discharge
w	spool valve area gradient
P_s	supply pressure
P_r	return line pressure
x_v	displacement of the directional proportional valve spool
ω_v	valve natural frequency
ζ_v	valve damping ratio
ρ	oil density
k_v	valve gain
S_{c_3}	scaling constant
S_{c_4}	scaling constant
C_{tm}	internal leakage coefficient
C_{em1}	chamber one external leakage coefficient
C_{em2}	chamber two external leakage coefficient
a_{leak}	leakage area

Filtering

$\mathcal{N}(\mu, \sigma)$	Gaussian normal distribution with μ mean and σ variance
$\mathbb{R}^n$	n-dimensional Real space
$\hat{\bullet}$	posteriori estimate

$\bar{\bullet}$	a priori estimate
$\tilde{\bullet}$	estimation error
$HT(\bullet)$	Householder Transformation
diag($\bullet$)	diagonal matrix
$E[\bullet]$	expected value
$\|\| \bullet \|\|$	norm

Units

Mach	Mach number
Pa	Pascal
m	meter
ft	feet
rad	radians

ABBREVIATIONS

CAS	Control Augmentation System
CD	Control Distributer
FCS	Flight Control System
FDI	Fault Detection and Isolation
FDIR	Fault Detection Isolation and Reconfiguration
FTC	Fault-Tolerant Control
GI	Generic Inputs
HM-FDIR	Health Monitoring – Fault Detection Isolation and Reconfiguration
LMI	Linear Matrix Inequalities
LPV	Linear Parameter Varying
MFRFC	Model Following Reconfigurable Flight Control
MMST	Multiple Model Switching and Tuning
MPC	Model Predictive Control
NN	Neural Networks
PAH	Pitch Attitude Hold
PIM	Pseudo Inverse Method
RAH	Roll Attitude Hold
SAS	Stability Augmentation System
UAV	Unmanned Aerial Vehicle
VHF	Very High Frequency
VOR	VHF Omni-directional Radio Range
HIL	Hardware-in-the-loop
PE	Persistent Excitation
KF	Kalman Filter

EKF	Extended Kalman Filter
DDF	Divided Difference Filter
UKF	Unscented Kalman Filter
SPKF	Sigma Point Kalman Filters

ABSTRACT

Mahulkar, Vishal V. Ph.D., Purdue University, December 2010. Prognosis Based Control Reconfiguration for an Aircraft with Faulty Actuator to Enable Performance in a Degraded State. Major Professor: Dr. Douglas E. Adams, School of Mechanical Engineering.

The objective of this work is to develop a framework for prognosis based control reconfiguration of dynamic systems with faults. A conventional feedback control design for a process plant or vehicle system may result in unsatisfactory performance (even instability), in the event of malfunctions in actuators, sensors, or other components of the system. In order to overcome the limitations of conventional feedback, new controllers need to be developed which are capable of tolerating component malfunctions whilst still maintaining desirable and robust performance and stability properties. In case where the malfunctions are faults, additional fault growth dynamics is introduced which leads to a continuously changing system response. The controller also needs to adapt to these changes while minimizing the rate of growth of the fault.

Due to cost and safety requirements, the ability to accommodate faults constitutes a desirable characteristic which can be incorporated in the control design process of a high performance aircraft. The specific case of a high performance aircraft with a faulty hydraulic actuator for a control surface was the main topic of investigation. A full six degree of freedom nonlinear model for an F-16 aircraft was developed using Simulink. A control system was then developed to allow guidance and navigation of the aircraft in the three dimensional space. A full nonlinear model of the hydraulic actuator with an internal leakage fault due to a faulty seal was also developed and combined with the aircraft model. The aircraft model was then simplified through linearization to allow development of a prognosis based control reconfiguration strategy and implementation in real-time.

The strategy was based on Model Predictive Control, where the optimal control problem was broken into an offline component and an online component. Offline component was used to generate a static map, which was employed for prognosis, based on the performance requirements and the mission profile. The online component used this static map to perform a mixed integer programming to optimize response speed and trajectory to satisfy constraints on degradation. The effectiveness of the reconfiguration strategy was demonstrated for simple missions in a longitudinal plane. At the lower level, a divided difference filtering algorithm was implemented to identify the fault in the control actuator. The fault information was then used in an adaptive framework to develop a fault tolerant controller for the actuator.

A hydraulic actuator test bench was designed to validate the developed models and control strategies through hardware-in-the-loop simulations. Experimental results demonstrated the effectiveness of the fault identification algorithm and the performance improvement obtained through implementation of the adaptive control strategy. The effectiveness of the reconfiguration strategy was also demonstrated experimentally by implementing an unknown wear function and comparing the results with and without the reconfiguration algorithm.

1. INTRODUCTION

The success of military operations involving manned and unmanned aircraft depend on their ability to perform precision maneuvers with accuracy, repeatability, and safety for the crew. In the civilian sector, crashes can occur due to inoperable aircraft leading to reduced public confidence in air travel. Several regulatory agencies are pushing for the development of technologies to reduce the fatal accident rate in the commercial sector that currently stands at a little under two per million flight hours [1].

Modern technological systems rely heavily on sophisticated control systems to meet increased safety and performance demands. This reliance on control is particularly high in safety critical applications, such as spacecraft, aircraft, nuclear power plants, chemical plants processing hazardous materials etc., where unattended and minor faults could potentially develop into catastrophic failures if maintenance is not performed in a timely and proper manner. By compensating for faults to some degree, conventional closed loop feedback control design for a process plant or air vehicle system may prevent that fault from being observed and will eventually develop into a control loop malfunction resulting in unsatisfactory performance (even instability). In the event of faults such as malfunctions in aircraft flight control actuators, sensor faults, or faults in other components of the system, new control techniques and design approaches need to be developed to minimize losses and avoid potential risks due to the faults. The new controllers should be capable of enduring system malfunctions while still achieving the desired level of overall system stability and performance. In short, the aim is to make the system *fault-tolerant* [2]. If the measures implemented are successful, they will restore the system behaviour to acceptable levels. The new controller modules must also be able to *predict* the rate at which the fault is growing due to the changes in the system dynamics and minimize the degradation rate to

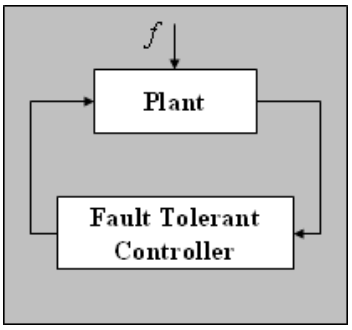


Figure 1.1. Fault-tolerant system [2].

allow the current task to be completed before expensive repair work is needed. Thus, another important aim is to *extend the life* of the component while minimizing the loss of performance in the system.

The term "controller" above is used in a very general sense. It does not only include the usual feedback or feedforward control law, but also the decision making layer that determines the control reconfiguration. The behaviour of the plant is analyzed by this layer to identify fault and modify the control law to keep the system in the region of acceptable performance. The process of accomplishing these goals is known in the literature as "Online Health Monitoring and Fault Detection, Identification and Reconfiguration (HM-FDIR)" [3].

1.1 Motivation

One of the important issues to consider when designing a fault-tolerant control system is whether to recover the original system performance/functionality completely or to accept some degree of performance degradation after the occurrence of a fault. The consequences of not taking into account the degradation in performance must be studied. It is also necessary to take the rate of degradation and degradation level into consideration when designing the control reconfiguration system. These issues have been addressed to some extent in the literature. Another issue that has received

little attention in the literature is the effect of control reconfiguration on the faulty system.

In the past decade, there has been substantial progress in the development of on-line FDIR techniques in aerospace applications [4–8] etc. A large number of techniques have been proposed and some have been tested on operating aircraft [8]. The FDIR techniques are generally divided into two categories:

1. Active: In the active fault-tolerant approach, it is assumed that a priori information about the fault or mechanism for detecting and isolating unanticipated faults is available. This information is then used to redesign the control system to achieve the desired performance and robustness.

2. Passive: In the passive approach, a closed-loop system can be designed to provide some inherent fault tolerance by carefully choosing the feedback gains, which take into account the effects of both the faults and parametric uncertainties in the model or disturbance.

Most of the existing research in the literature is centered around the objective of recovering as much of the pre-fault system performance as possible [9–14]. Some of these approaches assume total failure of the subsystem (actuator, sensor etc.) and then take one of two actions: (a) replace the failed components by their analytical or physical redundant counterparts, or (b) completely remove the failed components from the plant model. If the design objective is to restore the original performance of the system given faults in the subsystem components as stated above, then an unsustainable level of performance may be required from other (healthy) subsystems. If insufficient control authority is available to recover nominal performance, then the system is regarded as "unreconfigurable". This type of strategy is not optimal for two main reasons:

1. Extreme performance requirements may further damage the system, and

2. A faulty actuator may still be able to provide useful function.

Some authors (e.g. Zhang et al. [15]) have considered the problem of reconfiguration of a partially operational actuator; however, they fail to analyze and consider the fault dynamics or the effect of reconfiguration on the system under consideration. Thus, the current approaches presented in the literature have the following main drawbacks:

1. The magnitude of the fault is not considered while establishing the recovered level of performance. If the fault is small, it might be possible to recover the nominal performance level. But, if the nominal performance level is not recoverable and the controller still demands it, the system may degrade faster.

2. The fault dynamics are not considered during the reconfiguration process. For example, if there is a crack in an aircraft wing and aggressive maneuvers are requested, the crack is likely to grow faster.

3. The effect of reconfiguration on the system and, specifically, the fault is often neglected. For example, if reconfiguration of a worn system demands it to operate at high velocities and loads, the wear process is going to be accelerated resulting in more rapid failure.

Thus, the main objective of this dissertation is to find a trade-off between various constraints and objectives of the system in order to maximize safety and performance, minimize degradation and satisfy imposed mission constraints. The system under consideration is a high performance aircraft or a UAV with a faulty subsystem. The following section details the main objectives.

1.2 Objectives

The main goals of the research in this dissertation are as follows:

1. Develop models and modeling software for prediction of degraded aircraft performance that will help in simulating realistic fault scenarios under operational flight loads.

(a) The model will incorporate all aircraft subsystems of interest. This simulation model will include mathematical models of the propulsion systems, flight mechanics, flight control actuators, sensors, and the environment (i.e., wind, gravity, and atmosphere)

(b) Faults and failure models for aircraft subsystems such as actuators, sensors, etc. will be incorporated.

2. Flight control strategies will be developed to control and design mission profiles in the presence of constraints such as,

   (a) Performance constraints on the mission in the presence of faults;

   (b) Minimization constraints on excessive rates of degradation during the mission.

3. To validate the model and control strategies, a hardware in loop methodology will be developed specifically for actuators using real-time data acquisition and control.

1.3 Organization

The above stated objectives will be addressed in the following chapters. Chapter 2 contains a detailed review of the literature followed by the problem formulation, assumptions, approach and the scope of the work that has been undertaken. The development of the aircraft, hydraulic actuator, and the degradation models is detailed in Chapter 3. Chapter 4 will discuss in detail the Hardware-in-the-loop simulation environment. The detailed design of the experimental setup, the implementation steps and system identification process will also be discussed in this chapter. Chapter 5 will discuss the development of a baseline control structure for the systems under consideration. It will also deal with details of the implementation along with electrical and mechanical issues that were faced, followed by verification and validation of the developed models. An adaptive fault tolerant control strategy will be proposed

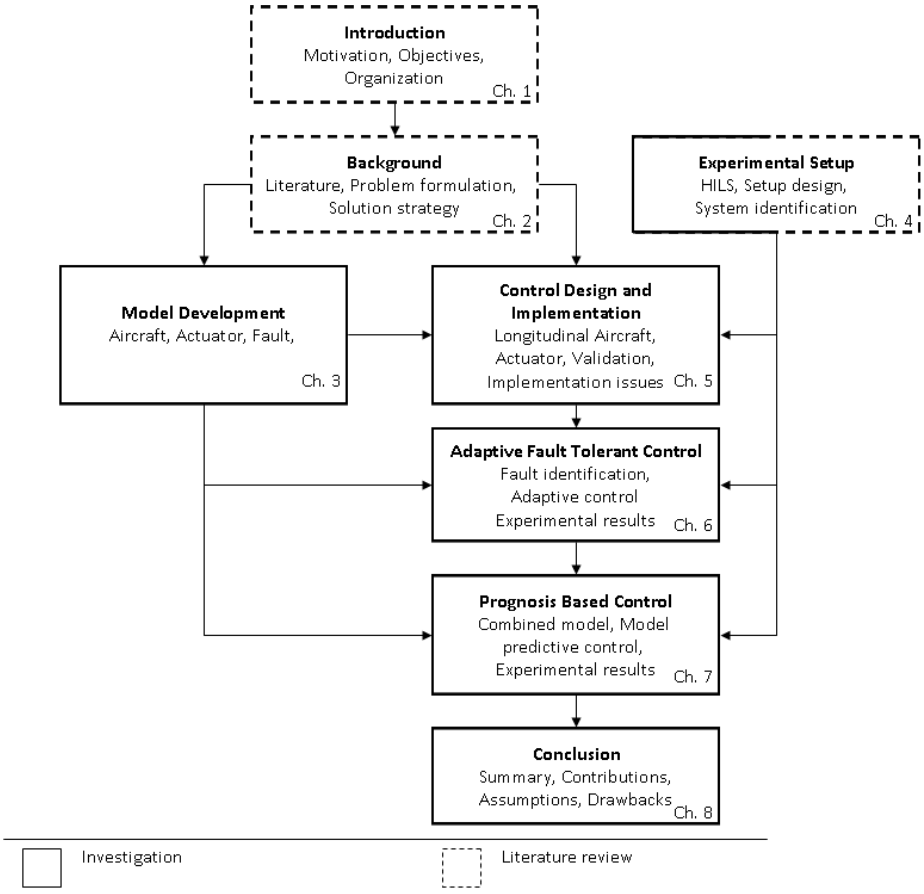


Figure 1.2. Organization of report.

and implemented. This will be followed by a prognosis-based control strategy and corresponding results in Chapter 6. The dissertation will conclude in Chapter 7 giving a summary of main results, contributions, a reiteration of important assumptions, drawbacks, and future directions. Figure 1.2 gives a snapshot of the organization of this document.

2. BACKGROUND

The application considered in this dissertation is a high performance aircraft/UAV with a faulty subsystem. The subsystems under consideration are the hydraulic control surface actuators, which maintain stability and allow navigation. The fault considered is a leakage fault. This chapter provides an overview of flight control systems followed by a detailed review of the literature concerning different aspects of FDIR: fault identification, fault-tolerant control systems, reconfigurable control systems, and path planning. An overview of fault statistics in the hydraulic actuator aircraft subsystem is also provided. The proposed approach is presented and formalized.

2.1 Flight Control System

The evolution of modern aircraft created the need for power-driven aerodynamic control surfaces and automatic pilot control systems. In addition, better performance requirements created a need to augment the stability of the aircraft dynamics over some segments of the flight envelope. Figure 2.1 shows the altitude-Mach envelope of one particular modern high-performance aircraft. The boundaries of the envelope are determined by a number of factors. The low speed limit is set by the maximum lift that can be generated (α limit), and the high speed limit follows a constant dynamic pressure contour that is limited by structural considerations. At higher altitudes, the speed becomes limited by the maximum engine thrust, and the service ceiling is the altitude limit at which the combination of engine thrust and airframe characteristics cannot produce a certain minimum rate of climb [16]. Consequently, the resulting envelope covers a wide range of dynamic pressures: from as low as $2390Pa$ during landing to $102770Pa$ at Mach 1.2 flight at sea level. Large variations in dynamic pressure cause corresponding large variations in the aerodynamic coefficients

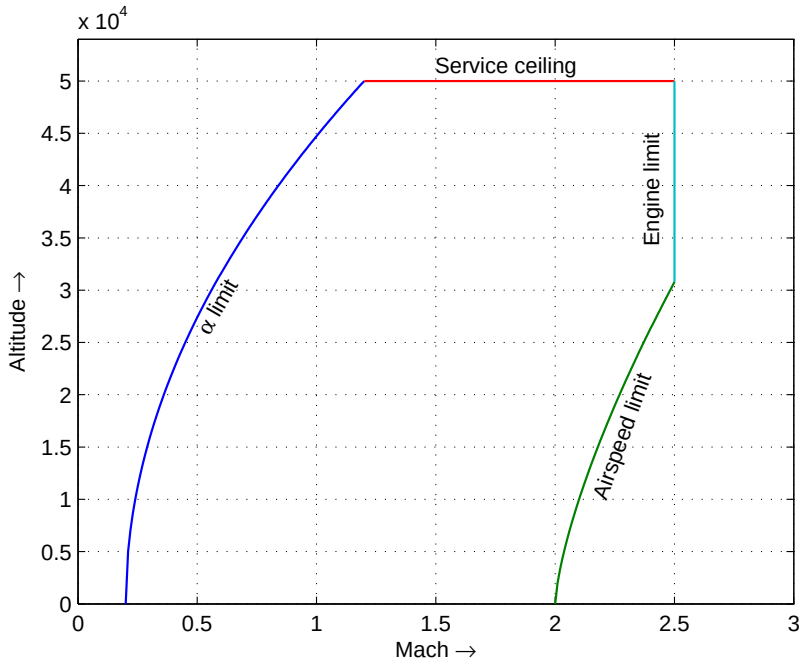


Figure 2.1. Altitude Mach envelope of a high performance aircraft [16].

in the equations of motion. Other factors such Mach number and aerodynamic angles also cause variations in the coefficients. These changes result in varying dynamics at different operating points in the envelope. These problems can be overcome by using a feedback control strategy to modify the aircraft dynamic characteristics. The responsiveness of an aircraft to maneuvering commands is determined in part by the speed of the rotational modes such as the short-period, roll, and dutch-roll modes. The frequencies of these modes are sufficiently high that a pilot would find it difficult or impossible to control the aircraft if the modes were lightly damped or unstable. It is, therefore, necessary to provide an automatic control system to prescribe these modes with suitable damping ratios and natural frequencies. Such control systems are called stability augmentation systems (SAS). If the augmentation system controls the modes in addition to providing a particular type of response to control inputs, the system is known as a control augmentation system (CAS). The slow modes (spiral and phugoid) are usually controllable by a pilot, but to avoid continuous attention

to controlling these modes, an autopilot is usually provided. The common types of SAS, CAS, and the autopilot functions are listed in Table 2.1:

Table 2.1 Control and augmentation systems for aircrafts [16].

SAS	CAS	Autopilots
Roll damper	Roll rate	Pitch attitude hold
Pitch damper	Pitch rate	Altitude hold
Yaw damper	Normal acceleration	Speed hold
	Lateral/directional	Automatic landing
		Roll angle hold
		Turn coordination
		Heading hold/VOR hold

2.1.1 Hierarchical Decomposition Of FCS

An aircraft requires guidance, navigation, and control to perform its missions. A human pilot interacts with the aircraft at different operational levels. A pilot performs three main functions: sensing, regulation, and decision making. Figure 2.2 shows a schematic of different layers and the interaction of the pilot with the aircraft. These interactions and controls can be separated into hierarchical decision layers as follows [17, 18]:

Strategic layer: This layer corresponds to the definition of mission objectives by a central command decision making entity, which is in most cases a human operator.

Tactical layer: In this layer, the motion planning algorithm determines how to best fulfill the goals set by the upper strategic layer.

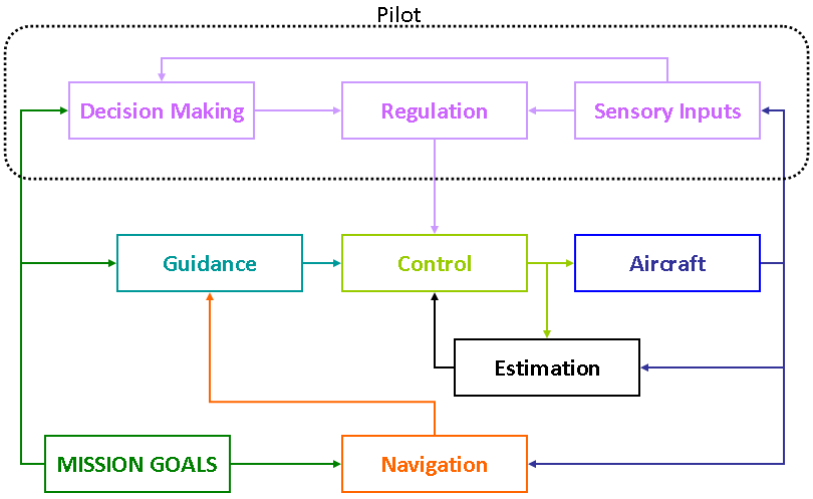


Figure 2.2. Aircraft subsystems and their interactions.

Reflexive layer: This layer attempts to track the trajectory planned at the tactical layer in addition to regulating and stabilizing the vehicle.

The hierarchical structure thus encompasses stability augmentation, control augmentation, autopilot, as well as a flight management system. There has been a trend towards automating as many pilot tasks as possible to reduce the load on a pilot, improve safety, and reduce flight costs. In the case of unmanned aerial vehicles (UAVs), all the tasks must be automated. Gain scheduling and switching are used to improve performance in different flight regimes. Control theory, heuristics, and reduced order optimization are used to achieve optimal trajectory management. These approaches are examples of intelligence applied to the operation and control of aerospace systems [17].

Investment in this type of flight control is justified only if it improves the function and performance of the aircraft, saves time and/or money required to complete a mission, or improves the safety and reliability of the system. Like all systems, those used in aircrafts are also subject to faults and failures. The hierarchical structure should be able to recognize the changes in system dynamics due to these faults/failures and take corresponding corrective actions. The latter is known in the literature

by several names: "Fault-tolerant Control" [2, 3, 9, 10, 19–22], "Reconfigurable Control Systems", [5–7, 11, 23–25], "Intelligent Control Systems" [17, 26], "Restructurable Control Systems" [27–31], "Self-repairing Control Systems" [32], "Fault Accommodation" [4, 33, 34], "Failure Compensation" [35, 36], etc. Closely related work has also been done in "Supervisory Control", "Fault Detection and Isolation", "Fault Diagnosis" and "Hierarchical Control Systems". The motivation for all of this prior research has been to overcome the unsatisfactory response of a conventional feedback controller designs in the event of a malfunctioning component. The principles of fault tolerant design are reviewed next to provide a survey of recent approaches developed by other researchers.

2.2 Fault Tolerant Control

A *fault* in a dynamical system is a deviation of the system structure or the system parameters from the nominal situation. Examples of system faults are blocking of an actuator, loss of a sensor, or loss of a mechanical or electrical connection of a system component. Parametrical changes are brought about, for example, by wear or damage. All these faults change the dynamic input/output characteristics of the plant leading to changes in the closed loop response.

Faults must be distinguished from disturbances and model uncertainties. Disturbances and model uncertainties are nuisances, which are known to exist, but the effects of these nuisances on the system performance can be attenuated through carefully designed robust controllers or measures such as filtering. However, disturbances and uncertainties can only be handled up to a certain size. Faults typically cause more severe changes in plant dynamics resulting in effects that cannot be suppressed by a fixed controller. Furthermore, faults have a tendency to grow over time and may lead to complete failure if no countermeasures are taken. Fault-tolerant control aims at changing the control law to attenuate the effects of the fault and prevent failure.

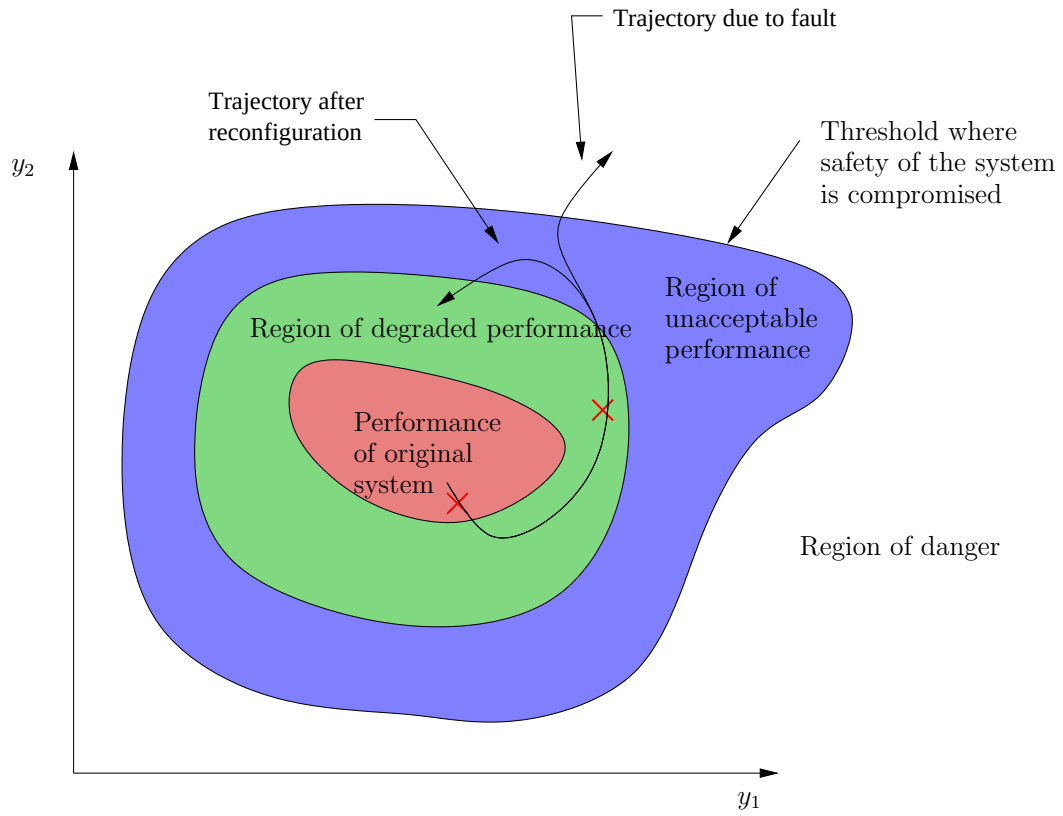


Figure 2.3. Changes in the region of performance envelope due to faults [2].

Figure 2.3 shows a representative evolution of system performance as the level of a fault in one of the subsystems increases. In the absence of any fault the system operates in its normal region of performance. Initially, when the level of the fault is small, the feedback control obscures the presence of the fault. As the leevel of the fault grows, the system performance degrades. If no other control measures are implemented, the system may eventually become unstable. The objective of fault-tolerant control is to bring the performance back within the region of acceptable performance. Generally, the approach that is used to enable fault-tolerance in systems consists of two steps:

1. Fault Diagnosis: Detect the existence, location, and magnitude of existing faults. The diagnostics block uses the measured inputs and outputs and tests

their consistency with the plant model. The output of the block is a characterization of the fault suitable for use by the controller redesign block.

2. Controller Redesign: Adapt the controller so that the system continues to satisfy established goals. The re-design block uses the fault information and adjusts the controller to the given scenario.

The general architecture of a fault-tolerant controller is depicted in Figure 2.4. f represents the fault in the plant, d is the disturbance on the plant, the reference and actual outputs are represented by y_{ref} and y respectively, and u is the control input. As was mentioned earlier, the term controller is used in a very broad sense and, hence, the connection between the controller redesign block and the controller block is shown with a double arrow to indicate an information link. The redesign of the controller may not only result in new controller parameters but also in a new control configuration.

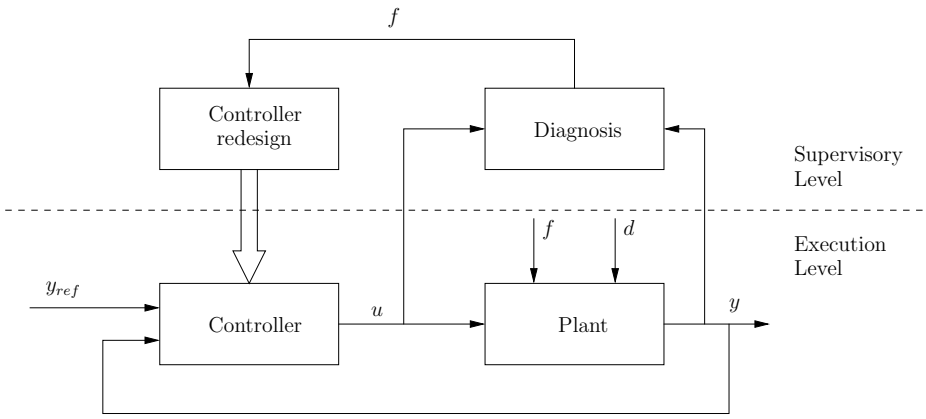


Figure 2.4. Architecture of a fault-tolerant control [2].

2.2.1 Fault Detection And Isolation (FDI)

Fault detection is the first step towards fault-tolerant control. Figure 2.5 shows the general structure of a fault diagnosis module. A variety of methods have been

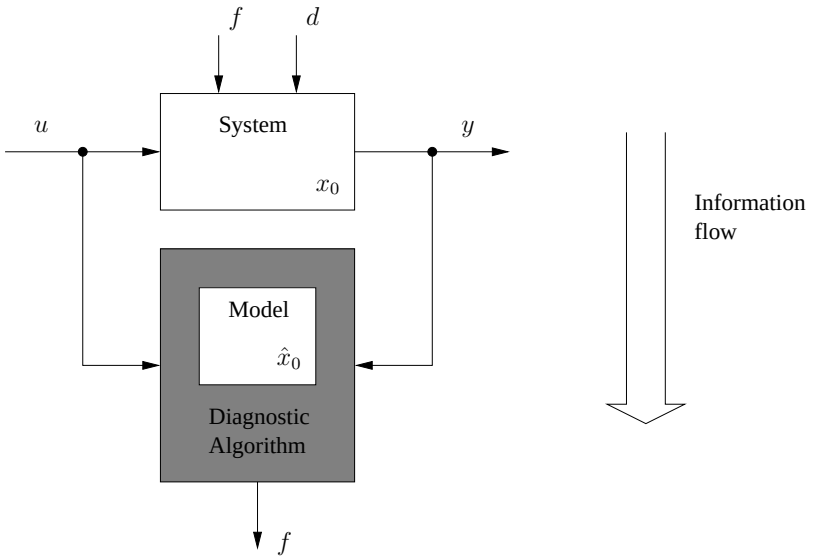


Figure 2.5. Structure of a fault diagnosis module [2].

proposed in the literature for early detection of faults in dynamical systems. These methods are broadly divided into two main types: hardware redundancy based and analytical redundancy based methods [37]. Analytical redundancy based methods are further divided into signal model based methods and process model based methods [38, 39]. An excellent review of fault detection and diagnosis methods is available in several survey papers [10, 40–42]. The most important of the analytical redundancy based approaches fall under parity space techniques, parameter estimation techniques, statistical techniques, and observer based techniques [43].

Model based techniques have received much attention in the last few decades [44]. A number of observer-based techniques have been proposed for linear systems, for example, eigenstructure assignment [45, 46], unknown input observer [38, 47, 48], etc., and for non-linear systems, for example, robust observer based methods [43, 49]. A drawback of observer-based methods in isolating faults is that they utilize the different directions of the different faults in the state space of the model, and, as a result, these methods are not capable of isolating faults that have the same direction in the system state space [50]. Parameter estimation techniques [51], on the other hand,

can detect and isolate faults, and may diagnose fault size, even for faults having the same direction in the state space of the model. Application of parameter estimation techniques to linearized models have been discussed in the literature [50]. Extended Kalman Filters (EKF) [52] and adaptive observers [53–55] have also found application in fault detection.

For an excellent bibliographical review on fault detection and identification, the reader is referred to Zhang and Jiang in [21].

2.2.2 Control Reconfiguration

Fault diagnosis and identification is a supervisory process and it alone cannot maintain performance and functionality of the faulty system in absence of fault tolerant control. Fault tolerant control is a control method that can accommodate system faults and failures automatically and maintain overall system stability and performance. A fault tolerant controller can be classified into two main types: passive and active.

1. Passive approaches make use of robust control techniques to ensure that the system remains insensitive to certain faults with bounded magnitudes and without using any online fault information [31]. The faulty system continues to operate with the same controller and the same control structure implying that the emphasis is placed on recovering the original system performance. Thus, the basic idea of passive fault-tolerance is to make the closed loop system robust against uncertainties and some very restrictive range of *likely* faults. A passive fault-tolerant control treats faults in the same manner as modeling uncertainties. No guarantee of stability and performance can be made [10].

   The methods generally used in the design of such control systems are: quantitative feedback theory [56, 57], frequency domain H_∞ [58], robust design based on "4 parameter" controller [10], sliding mode control [59, 60], and linear matrix inequalities [61]. The application of sliding mode control to reconfigurable

flight control problems as with other applications has a tendency to produce severe chattering. The use of a boundary layer on the sliding surface has been employed by some researchers to damp this chattering [59]. Robust control techniques generally result in smoother control signals. In a recent publication, H_2 with linear matrix inequalities was utilized to construct a fault-tolerant control with guaranteed tracking performance [61, 62]. The advantages that make passive approaches appealing are that they do not require fault detection and isolation; and passive approaches are also generally easy to implement [63]. A passive fault-tolerant controller acts as a good baseline controller that can then be used further in active reconfiguration.

2. Active approaches are usually variable in their structure. A wide variety of active approaches have been proposed by researchers. Active fault-tolerant methods can be more usefully classified based on whether or not [10]:

   (a) they use off-line (pre-computed) control laws,

   (b) they perform online fault accommodation,

   (c) they are tolerant to unanticipated faults using fault detection and isolation, or

   (d) they are dependent on use of a baseline controller.

   The simplest way to achieve fault-tolerance is to store pre-computed gain parameters. This method originated from the development of flight control systems where gain scheduling was considered as a solution for dealing with changes in flight aerodynamic coefficients with changes in flight parameters such as Mach number, altitude, and angle-of-attack. This approach has been used in flight control, space control, chemical process control, etc. For example, Rauch [58], provided a pre-computed control law rescheduling approach to F/A-18 aircraft. This experimental system had induced control surface faults and was flight-tested under turbulent conditions. Huzmezan and Maciejowski [64] describe re-

configuration and scheduling using quasi-linear parameter varying (LPV) models with an FDI scheme and constrained model based predictive control. Application to a missile example was also provided by the authors. An LPV reconfigurable control design was also developed by Ganguli *et al.* [65] and applied to a Boeing 747. Theoretical investigations of control law re-scheduling have been carried out by Shamma *et al.* [66] and others. Moerder *et al.* [67] have developed a scheme to monitor the systems control impairment status, and provide state estimates that are used in an optimal gain scheduling scheme. Control law re-scheduling can be thought of as feedback control with gains that are adjusted through feedforward compensation. The compensations as applied by the researchers have typically been open-loop adjustments because there is no feedback from the closed loop system performance to compensate for the action of incorrect scheduling [10]. To overcome this open-loop adjustment, Zheng *et al.* [68] have developed an LMI based approach for synthesizing control feedback as a function of "fault effect vectors". This approach was demonstrated on a longitudinal motion flight control for an unmanned aircraft.

The concept of feedback linearization can be used to overcome the disadvantages of linear controllers. Feedback linearization can implicitly take into account the effect of coupled motions. The methodology has been applied by Meyer *et al.* [69], Ochi & Kanai [28,29]. An important feature of feedback linearization is the concept of *control distributor* (CD) and *generic inputs* (GI) [28]. The CD is introduced to reduce the actual input vector to a GI vector so that the number of inputs are equal to the number of outputs. Ochi et al. applied this method to a large transport aircraft by using the concept of an imaginary actuator to generate a particular GI signal [29].

Model-following is an alternative to feedback linearization. The goal of a model following approach is to imitate the performance characteristics of a desirable model. The model-following reconfigurable flight control (MFRFC) system was first proposed by Huang *et al.* [30]. Morse *et al.* [70] used an adaptive model-

following methodology for a multi-variable system. The MFRFC has the capability to distribute the control effort without explicit knowledge of the fault. The ideal form of model-following is perfect model-following in which the behaviour of the system can be completely specified. The conditions for perfect model-following are very restrictive because most systems have more states than inputs [10]. As an approximation, researchers have proposed explicit model-following, which requires system outputs to follow those of the model in the least-squares sense. When model-following is approximated by minimizing a quadratic function of the actual and modeled states, it is known as implicit model-following.

The pseudo-inverse method [24] (PIM) also known as control mixing [71] attempts to recover the performance of a nominal system by computing an approximate matrix inverse. The main objectives of PIM in reconfigurable control are [10]:

(a) to maintain as much simplicity as possible,

(b) to approximate the nominal system as closely as possible with the reconfigured system, and

(c) to provide graceful degradation in performance subsequent to a fault.

Given a linear system:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{2.1}$$

the control $u = Kx$ results in a closed loop system

$$\dot{x} = (A + BK)x \tag{2.2}$$

$$y = Cx \tag{2.3}$$

If the post fault system is given by:

$$\dot{x}_f = A_f x_f + B_f u_f$$
$$y_f = C_f x_f \tag{2.4}$$

with A_f and B_f stabilizable, the gain matrix K_f can be computed to stabilize the system

$$\dot{x}_f = (A_f + B_f K_f) x_f \tag{2.5}$$
$$y_f = C_f x_f \tag{2.6}$$

The PIM selects K_f to minimize:

$$J(K_f) = ||(A + BK) - (A_f + B_f K_f)||_F \tag{2.7}$$

where $|| \cdot ||_F$ is the Frobenius norm [24]. The equation (2.7) can be solved by using the pseudo-inverse of B_f:

$$K_f = B_f^+ (A + BK - A_f) \tag{2.8}$$

The main difficulty arises when B_f is not of full row rank. Gao and Antsaklis [24] presented a method to generate a K_f that maintains stability even in this case. By integrating fault detection and identification into the design approach, Bošković [5, 6] advanced the pseudo-inverse methodology. His approach applies multiple models, switching, and tuning (MMST) [5]. PIM controllers are designed for a set of models of plant dynamics, the model that best approximates the plant dynamics is chosen, and the corresponding controller is activated. The methodology assumes that each model has at least one stabilizing controller and proper switching between controllers can stabilize the plant.

Eigenstructure reassignment is another method to generate the gain matrix, K_f, that stabilizes the post fault system as given in equation (2.4). The basis behind this method is that the response of a closed loop system is determined by the location of its eigenvalues and the direction of the corresponding eigenvectors.

K_f is chosen to place the eigenvalues of the closed loop system at the desired locations or in the desired regions. Any remaining degree of freedom is used to minimize the distance between the achieved eigenvectors and a desired set of eigenvectors [45, 72–74]. This minimization problem is more difficult but achieves better response when an explicit pseudo-inverse for B_f does not exist.

Successful implementation of both the pseudo-inverse method and eigenstructure assignment requires the control designer to address control saturation. It is often not possible to match the performance of the nominal system in the presence of faults and attempting to recover nominal performance will result in control saturation. Bodson suggests four command limiting techniques that recover a degree of stability in the event of actuator saturation [75].

Optimal control techniques offer a more refined method to alleviate control saturation in the post fault system. Model predictive control (MPC) has been applied by researchers to obtain a level of fault-tolerance [64, 76–79]. MPC generates control inputs that are usually based on the objective of minimizing a cost function such as the integrated tracking error. The designer can implement various constraints during the optimization such as position and rate saturation on control inputs. Receding horizon model predictive control produces an optimal control sequence for a finite receding time interval [80, 81]. The primary drawback of MPC is the computation time required to solve the optimization problem.

All of the approaches discussed above require a fault detection and identification capability. A separate variety of adaptive reconfigurable flight controllers achieve fault-tolerance without dependence on explicit FDI schemes. Such controllers typically include a single adaptive controller that can accommodate a range of faults without switching. Because these controllers do not involve switching, the interaction between the FDI mechanism and the reconfiguring controller does not cause a concern for stability [10]. The adaptation is driven

by an estimation process that does not require prior knowledge of the faults. Adaptive controllers are divided into two categories:

(a) Direct adaptive control systems which attempt to learn control parameters.

(b) Indirect adaptive control systems that learn parameters of the plant.

Both direct and indirect adaptive controllers draw information from the response of the system. The identification process requires signals that contain a sufficient level of information [82].

Indirect adaptive control generates useful information about the plant and can be used for flight planning, FDI as well as flight control. All the conventional fault-tolerant methods described earlier such as the pseudo-inverse method, eigenstructure assignment, optimal control etc. also become available to the designer. Traditional indirect adaptive control systems are based on the certainty equivalence principle, so that a recursive identifier is combined with a control algorithm that uses the estimated parameters as if they were perfect estimates [83]. Such schemes can only handle a very limited class of failures. Indirect adaptive control techniques with receding horizon MPC have been used by Miller *et al.* [84].

Direct adaptive control estimates the control parameters and not plant parameters. The estimated parameters are immediately useful in the construction of a control vector. Multivariable adaptive control, adaptive neural networks, variable structure control, and adaptive loop shaping are some of the adaptive control techniques that have been applied by researchers to reconfigurable flight control. Bodson considered the application of multivariable adaptive control techniques to flight control reconfiguration [7]. He also presents the comparison of three implementations: indirect adaptive, direct adaptive input error, and direct adaptive output error. Due to the robustness of sliding mode control techniques, Shtessel applied a reconfiguration strategy based on a continuous sliding-mode controller with direct boundary-layer adaptation for flight control

reconfiguration [85]. Another reconfiguration strategy applied to sliding mode control is variable structure control [86].

Adaptive neural networks (NNs) are an appealing choice for providing fault-tolerance in flight control systems due to following properties [22]:

(a) Learning and adaptation: NNs can be trained using past recorded or simulated data or current data.

(b) NNs are applicable to nonlinear systems due to their mapping capabilities.

(c) NNs are multi-input, multi-output and are thus applicable to multivariable systems.

(d) NNs have an architecture suitable for parallelization and thus lead to high speed hardware implementation.

Direct adaptive NNs were used on the Air Force Reconfigurable Control for a Tailless Fighter Aircraft (RESTORE) program due to their ability to stabilize a vehicle following the ocurrence of damage [11]. Adaptive NNs employ adaptation to cancel dynamic inversion errors, which result from faults. Numerous derivatives of adaptive neural networks such as pseudo-control hedging [87], dynamic cell structure [88] etc. have been developed.

2.2.3 Path/Trajectory Planning

Most UAV flight controllers have a flight path planner built in. Path planners, also referred to as trajectory generators or outer loop controllers, generate a flight path to accomplish a set of objectives. The flight path is a sequence of position and/or velocity commands for the low level controllers. Flight path planners have to consider various restrictions before generating a trajectory: the plans should be conservative enough so that the lower level controller can easily track them, and for linear controllers, the trajectories should not induce significant nonlinear dynamics. After the onset of a fault, the capabilities of the vehicle are unknown and the trajectory generation

has to take additional constraints into consideration. Pseudo-control hedging allows the aircraft to remain stable in the presence of actuator saturations [87]. Adaptive limit detection uses an adaptive neural network to calculate the control margins of an aircraft and predict quasi-steady response [89]. Limit avoidance protects the aircraft from exceeding its limits through the use of envelope protection systems [90].

Algorithms such as online optimization, mixed integer programming, and MPC have been used to generate flight paths [84, 91, 92]. More recent developments involve the use of finite automaton based reconfiguration for flight paths [18, 93]. Some of these techniques are computationally expensive and cannot be performed in real-time. A variation is to store optimization results generated by off-line optimization in a polynomial neural network, which can be accessed in real-time [94].

2.2.4 Hydraulic Actuators

Fast response times and high size to power ratios have made hydraulic systems very popular in the industrial sector for delivering large forces and torques [95]. Their industrial applications include positioning [96–98], active suspensions [99–101], material testing, aircraft, and industrial hydraulic systems. With increasing economic demand for high plant availability, safety of hydraulic systems, and risks associated with faults, reliability becomes an important factor in hydraulic applications. In aircraft, because of their high power density and fast response, hydraulic drives are used to position the control surfaces. Figure 2.6 shows a schematic diagram of a typical hydraulic drive [102]. Because of the complexity of a hydraulic circuit and difficult environmental conditions, the reliability of hydraulic systems is always an important consideration.

Hydraulic components wear during operation which causes the system parameters to drift gradually. This may eventually lead to faults and failures. Furthermore, since the fluid is operated under high pressure, any leakage or seal damage as well as other malfunctions of the system can cause degradation in performance. Failure rates

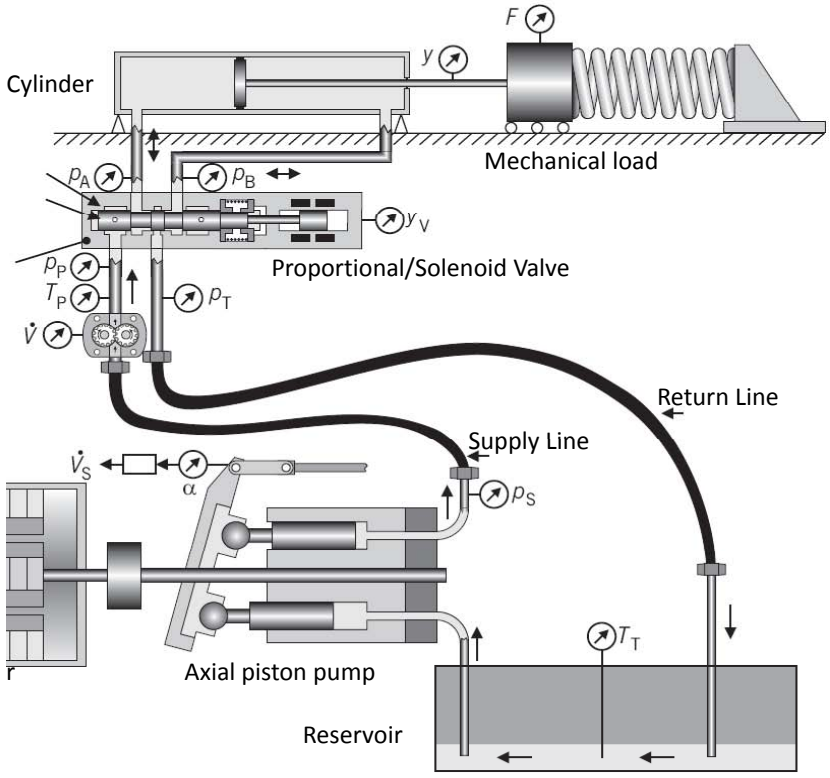
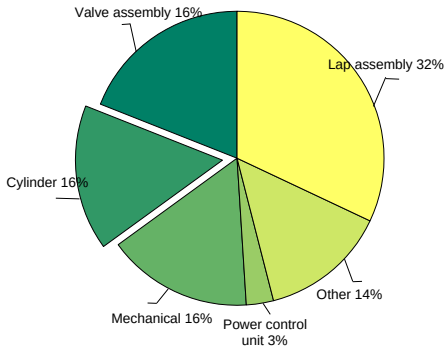


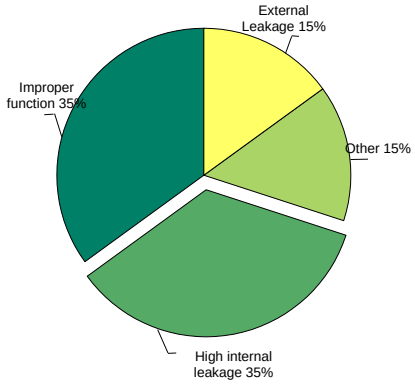
Figure 2.6. Schematic of a typical hydraulic actuator experimental setup.

and other fault statistics are important factors that determine the fault detection scenarios that a diagnosis system will typically be faced with in practical operations. Unfortunately, it is typically difficult to obtain failure rates. This difficulty has to do with the fact that the respective numerical values are protected as intellectual property within companies and are not published. One exception is in the area of aircrafts, where the high safety standards of both airlines and manufacturers necessitate intense engagement and documentation in safety related issues.

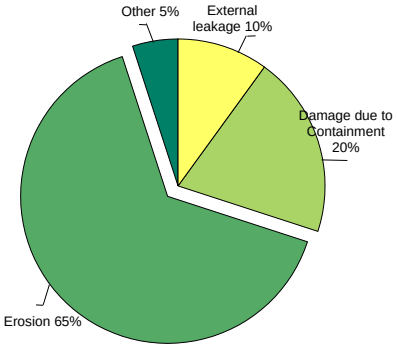
Münchhof obtained data related to the failures from two industrial sources [102]. The data is presented in graphical format in Figures 2.7(a), 2.7(b), 2.7(c), and 2.7(d). Table 2.2 shows the failure rates for hydraulic components obtained from theoretical



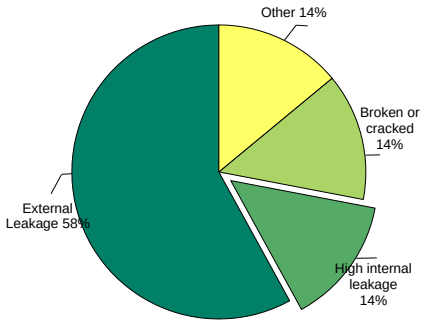
(a) Failure rates for different hydraulic components



(b) Failure rates for control valve assemblies



(c) Failure rates in lap assemblies



(d) Failure rates for hydraulic cylinders

Figure 2.7. Statistics of failure rates for hydraulic components in aerospace applications [102].

considerations. The defects (or faults) listed in Table 2.2 are classified according to the criticality of the detects as follows:

- A: Flight Safety Critical Defects Severeness–Class 1
- B: Flight Safety Involved Defects Severeness–Class 2
- C: Defects Severeness–Class 3
- D: Defects Severeness–Class 4

Table 2.2: Failure rates of hydraulic components in aircrafts [102].

Unit	Type of Fault	Fault	$\lambda[\cdot 10^{-9}]$	Crit.	Effect
Linear Motor	Electrical	Coil Cable Break	0.4	D	Reduction of valve spool actuation force due to loss of one coil
		Short-circuit	0.001	C	Loss of one coil system and amplifier
		Hardover	0.006	C	Amplifier fault, parasitic force, high electrical and thermal stress
		Sensor fault	0.8	C	Loss/reduction of drive control in affected control circuit
	Mechanical	Breakdown of mechanical component	0.001	B	Increased friction, drive jam, complete loss possible
		Contamination	0.001	D	Increased friction
		Field weakening	0.003	C	Reduction of controllability and linearity
		Break or tension release of return spring	0.5	B	Offset of one-sided fault, less control quality, loss of contrlability
Direct drive valve	Mechanical	Drive shaft break	0.001	B	Loss of controllability, total loss of actuator
		Jam due to splinters	0.001	B	Loss of controllability, total loss of actuator
		Mechanical friction	0.006	D	Reduced spool dynamics, reduced dynamics
		Breakdown of mechanical components	0.001	B	Loss/reduction of actuator function, external leakage
	Hydraulic	Oil filter clogging	0.01	C	Reduced force, reduced piston dynamics
		Air bubbles	0.04	D	Reduced controllabiliy
		Control edge wear	0.003	D	Reduced controllability

Table 2.2 – continued from previous page

Unit	Type of Fault	Fault	$\lambda[\cdot 10^{-9}]$	Crit.	Effect
Tandem Cylinder		Internal leakage	**0.5**	D	Reduced controllability, higher flow in neutral position
	Electrical	Defect of Sensor cable	0.4	B	Loss of drive control of affected circuit
		LVDT Sensor fault	0.8	C	Reduction/loss of controllability
	Mechanical	Mechanical friction	0.06	D	Reduced piston dynamics
		Structural breakdown	0.001	C	Reduction/loss of actuator function, external leakage
	Hydraulic	**Seal breakdown**	2.0	C	Reduced actuation forces, reduced piston dynamics, high internal and external leakage
		Leakage of connection lines	0.4	B	High external leakage, affects hydraulic supply
		Internal Leakage	0.01	C	Reduction of actuation forces and controllability
		Inter-system leakage	0.5	C	Reduction of actuation forces and controllability in both hydraulic circuits

In this application, the leakage faults will be considered. Leakage faults are a general symptom of worn hydraulic components. In most cases, leakage increases progressively over a long period of time causing the system dynamics to also change. Based on the location, the leakage can be classified into: (i) internal (crossport) leakage, where the fluid leaks to another part of the circuit within the hydraulic system, for example, from one chamber of the hydraulic actuator to the other and (ii) external leakage where the fluid leaks out of the hydraulic circuit. Whereas external leakage can be found through visual inspection, internal leakage caused by seal damage

cannot be detected easily until the actuator seal is severely damaged resulting in degraded performance. Thus, it is important to identify this fault as part of any health monitoring strategy. This kind of fault can lead to severe degradation of performance leading to safety issues and eventually a non-responsive actuator.

Before the failure occurs, most components go through the process of fault initiation followed by fault growth due to lack of any control measures, eventually leading to complete failure. It is possible to model some of these faults and fault growths either analytically or empirically [103–108]. The availability of a continuum model makes it possible to introduce the fault dynamics in the control design and reconfiguration process.

2.3 Problem Formulation

Despite the vast amount of research that has been conducted on fault-tolerance and reconfigurable control for aircraft, several shortcomings still exist. The majority of reconfigurable flight control research aims at recovering the performance of the nominal system. The severity of the fault should be one of the factors determining the recovered performance level. Furthermore, there is no consideration for the effect of the recovery process on the fault dynamics. Most researchers model a fault as some loss of input performance by multiplying the input matrix by a scaling factor. The effects of a fault should be allowed to enter the system dynamics implicitly through the dynamics of the actuator. The current research also lacks adaptive reconfigurable path planning algorithms. Thus, based on the shortcomings identified in the available research and discussion in Chapter 1, the goals to be accomplished by this dissertation research are as follows:

1. Develop a full 6 degree of freedom nonlinear model for a high performance aircraft.

2. Develop a detailed model for a hydraulic actuator as a part of the aircraft flight control subsystem.

3. Develop a degradation model for a fault in one of the actuators in the flight control subsystem.

4. Develop a reconfiguration strategy to accommodate faults by recovering acceptable level of aircraft performance while minimizing the degradation rate of the faulty subsystem.

5. Validate the developed control strategies through hardware-in-the-loop simulations.

The problem is formulated as follows to be solved mathematically:

$$\text{Plant Dynamics:} \qquad \dot{x}_p = f_p(x_p, \bar{p}_p, u_p) \tag{2.9}$$

$$\text{Plant Output:} \qquad y_p = h(x_p, \bar{p}_p, u_p) \tag{2.10}$$

$$\text{Initial Conditions:} \qquad x_p(t_0) = x_{p_0} \tag{2.11}$$

$$\text{Actuator dynamics:} \qquad \dot{x}_a = f_a(x_a, \bar{p}_a, u_a) \tag{2.12}$$

$$\text{Degradation Model:} \qquad \dot{d}_a = f_d(d_a, \bar{p}_a, x_p, x_a, u_p, u_a) \tag{2.13}$$

where,

x_p	are the plant states
y_p	are the plant outputs
$\bar{p}_p$	are the nominal plant parameters
u_p	are the inputs to the plant
f_p	represents the plant dynamics
x_a	are the actuator states
$\bar{p}_a$	are the nominal actuator parameters
u_a	are the inputs to the actuator
f_a	represents the actuator dynamics
d_a	represents the damage index
f_d	represents the damage dynamics

Given a performance index J, the control problem in the presence of faults can be defined as:

Problem 2.1 (Optimal Control problem)**.**

$$\min_{u} \quad J = J(x,u) \qquad \textit{performance index} \tag{2.14}$$

$$s.t.: \quad \dot{x} = f(x,u,p) \qquad \textit{system equations} \tag{2.15}$$

$$x(0) = x_0 \qquad \textit{initial conditions} \tag{2.16}$$

$$\lambda(x,u) \in [\lambda_l, \lambda_u] \qquad \textit{constraints} \tag{2.17}$$

$$x \in [x_l, x_u] \qquad \textit{state limits} \tag{2.18}$$

$$u \in [u_l, u_u] \qquad \textit{control limits} \tag{2.19}$$

Here, all the states have been merged into a single vector x and the inputs into u. The performance index J contains indices for both performance of the aircraft as well as the degradation mechanism. For example, J can be defined as follows:

$$J = \int_0^\infty (e_y^T Q e_y + d_a^T M d_a + u^T R u) dt \tag{2.20}$$

where e_y is the output error. Q and M are positive semidefinite matrices, and R is a positive definite matrix resulting in a non-negative performance index J and, hence, the minimization control problem as constructed in Problem 2.1 has a solution. There are a number of ways the performance index can be modified to suit the requirement. For example, instead of an output error e_y, the derivative of the error can be used. In the same manner, instead of the absolute value of damage, the damage rate $\dot{d_a}$ could be used.

2.4 Approach

A block diagram representing the solution strategy is presented in Figure 2.8. The plant consists of models for the aircraft, the hydraulic actuator, and the fault.

A fault estimator give information about the current level of degradation within certain confidence intervals. This information along with the reference mission profile is utilized by the prognosis module to establish the amount of degradation the system will observe at the end of the mission. Given an upper limit on acceptable level of degradation, the flight path optimizer and the control reconfiguration module modify the flight profile and the control structure/parameters respectively to meet the degradation constraints. All of this is done in an optimal fashion to ensure that best possible performance can be extracted from the aircraft within the available constrain region.

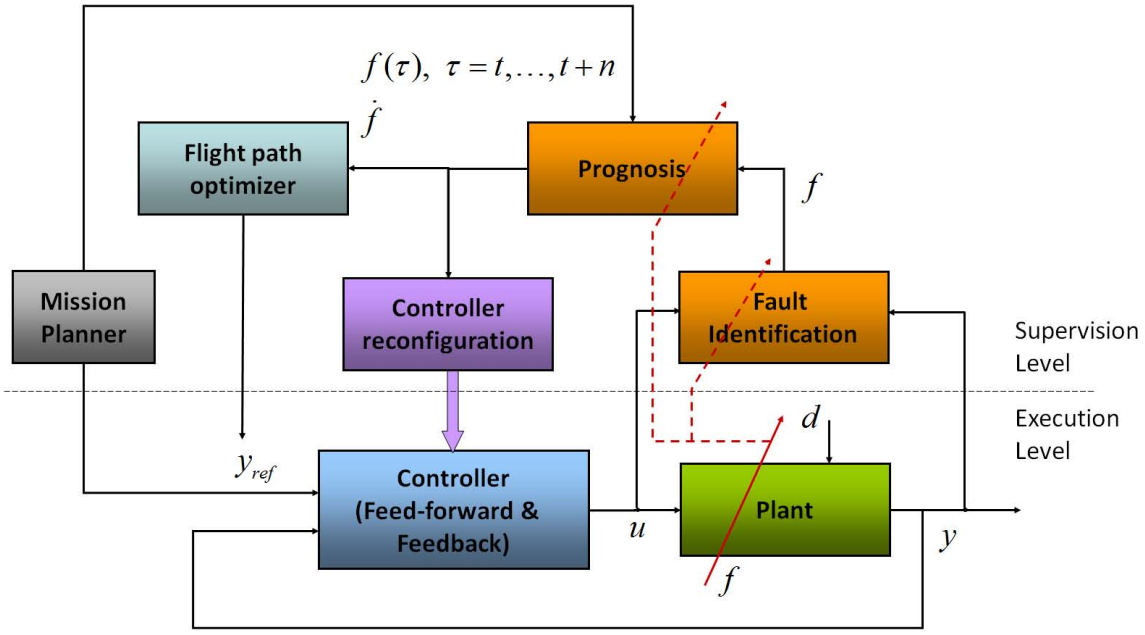


Figure 2.8. Block diagram of the approach.

3. MODEL DEVELOPMENT

In this chapter, models for the aircraft and the hydraulic actuator are developed. A model for a degradation mode of the hydraulic actuator viz. an internal leakage fault in the cylinder due to a damaged seal will also be developed. The models will then be simplified to permit real-time implementation.

3.1 Aircraft Model

A six degree of freedom aircraft model was developed for implementation using Simulink. The equations of motion for the aircraft are recorded in Appendix A. For brevity, only the form of the final equations that were obtained from the derivation exercise are given here:

$$\dot{x}_p = f_p(x_p) + g_p(x_p)u_p \tag{3.1}$$

$$y_p = h_p(x_p) \tag{3.2}$$

x_p are the states of the aircraft system, u_p are the inputs, and y_p are the outputs.

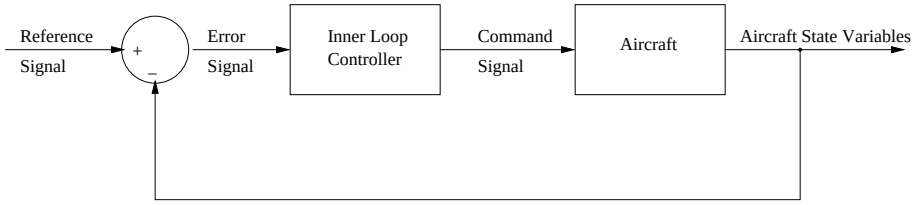
3.2 Autopilot Structure

In this section, an autopilot system for the aircraft is developed based on "classical" feedback control theory. The basic functions of an autopilot can be divided into two areas: guidance and control.

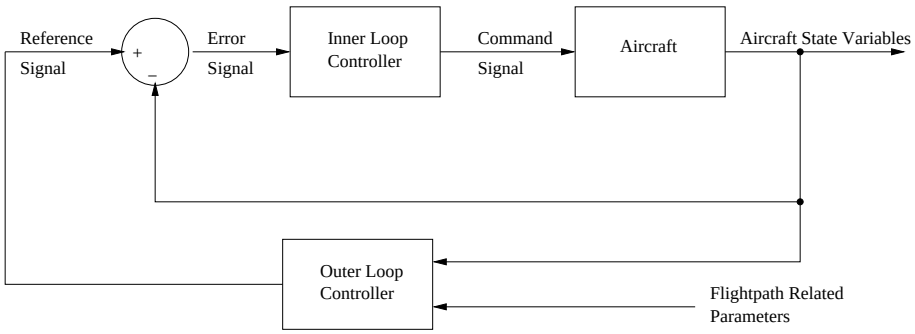
- Guidance is the action of determining the course and speed relative to some reference co-ordinate system to be followed by the vehicle.
- Control is the development and application of appropriate forces and moments through the use of control surfaces to establish an equilibrium state of the

vehicle, to restore a disturbed vehicle to its equilibrium or to regulate the vehicle within desired limits.

The control loops ensure a fast and stable response of the aircraft to commands created by the guidance loops. The autopilot can be divided into *inner* and *outer* loops. The inner loops (Figure 3.1(a)) control the roll and pitch attitudes of the aircraft. The roll and pitch commands are created by an outer loop (Figure 3.1(b)), which guides the aircraft along a desired trajectory. An aircraft autopilot uses five



(a) Inner loops of the autopilot



(b) Outer loops of the autopilot

Figure 3.1. Control loops for an autonomous aircraft.

longitudinal and five lateral control modes. The longitudinal control laws control the elevators and the thrust; the lateral control laws control the ailerons and the rudders [109, 110]:

Longitudinal autopilot modes

1. Pitch Attitude Hold (PAH): The Pitch Attitude Hold mode is the basic longitudinal autopilot mode; it controls the pitch angle θ by applying appropriate deflections of the elevator, δ_e, if the value of θ differs from the desired reference value. Normally, the PAH mode serves as the inner loop for the Altitude Hold, Altitude Select, and longitudinal Glideslope mode.
2. Altitude Hold (ALH): The Altitude Hold mode is used to maintain a reference altitude, which is entered by the pilot. This mode uses the PAH mode.
3. Altitude Select (ALS): The Altitude Select mode actually controls the rate of climb of the aircraft.
4. Approach Glideslope (GS): In approach mode, the aircraft must be guided along the reference planes of the glideslope and localizer. The Glideslope mode is the longitudinal part of the approach mode, which brings the aircraft from level flight into a descent, following the glideslope reference plane. The localizer provides runway centerline guidance to aircraft. These reference planes are provided by the Instrument Landing System (ILS) radio signals, which can be detected in the aircraft.
5. Go Around (GA): The Go Around mode is used to cancel an approach.

Lateral autopilot modes

1. Roll Attitude Hold (RAH): The Roll Attitude Hold mode is the basic lateral autopilot mode. The main purpose of this mode is to serve as the inner-loop for the other lateral autopilot modes, but it is also possible to use the RAH mode separately, for instance, for fly-by-wire control via a side-stick.
2. Heading Hold/Heading Select (HH): The Heading Hold / Heading Select mode is used to maintain or select a certain heading of the vehicle.

3. Approach Localizer: (LOC) In approach mode, the autopilot uses ILS signals to line up along the runway centerline, and to follow the glideslope reference line.
4. Navigation (NAV): This mode allows an aircraft to fly along a VOR bearing.
5. Go Around (GA): The lateral part of the Go Around mode, which is used if an approach has to be canceled, is a special case of the Roll Attitude Hold mode.

It is possible to make a distinction between lateral and longitudinal modes based on corresponding motions; however, these two modes are not independent and to prevent lateral movements affecting the performance of longitudinal controls loops, it is necessary to include some lateral/longitudinal interconnection. This approach is known as turn compensation. The autopilot as described here functions properly only if the velocity is maintained more or less constant during the maneuvers.

3.3 Longitudinal Autopilot Modes Implementation

3.3.1 Pitch Attitude Hold (PAH)

The pitch attitude hold mode is the basic longitudinal autopilot mode. It controls the pitch angle θ through an appropriate input to the elevator, δ_e, if the value of θ differs from the reference. PAH serves as an inner loop for the ALH mode. A proportional and integral controller is used to ensure that no steady-state errors remain. The θ-feedback decreases the damping of the short period mode, which is compensated by feeding back the pitch rate q.

3.3.2 Altitude Hold Mode (ALH) with Turn Compensation

The altitude hold mode is used to maintain a reference altitude as obtained from the waypoint information. It uses the PAH mode as the inner loop with the reference

pitch, θ_{ref}, proportional to the difference between the reference and actual altitude, $\Delta H = H_{ref} - H$. The block diagram implementation of the ALH mode with an inner PAH mode is shown in Figure 3.2. The turn compensation block is used to compensate for loss of lift due to a turning maneuver. The utility of this block will become clear when we examine the results from a turning flight. A sample simulation result is shown in Figure 3.3. It can be seen that the control strategy is able to achieve the desired altitude with a very small error. The maximum and minimum pitch angles have been limited to prevent the aircraft from exceeding limits on the angle of attack, which in turn limits the rate of descent, recorded at about 150m/s and the rate of ascent, which was recorded at about 80m/s.

3.4 Lateral Autopilot Modes Implementation

3.4.1 Roll Attitude Hold (RAH) with Turn Compensation

The roll attitude hold mode is a basic lateral autopilot mode. It serves as an inner loop for other lateral modes. The deviation of the actual roll attitude from the commanded attitude is fed back to the ailerons through proportional and integrating controllers to ensure that the desired roll rate is reached. A roll causes an aircraft to execute a turning maneuver with a non-zero side slip angle. A turn-coordination loop is used to accomplish the following two objectives:

i. to suppress the side slip angle during turning maneuvers through appropriate aileron and rudder deflections.

ii. to suppress the adverse yaw generated when a turn maneuver is initiated through aileron deflections.

3.4.2 Heading Hold (HH)

The heading hold mode is used to maintain a certain heading of the aircraft. This mode uses the RAH mode as the inner loop. A reference value of roll angle,

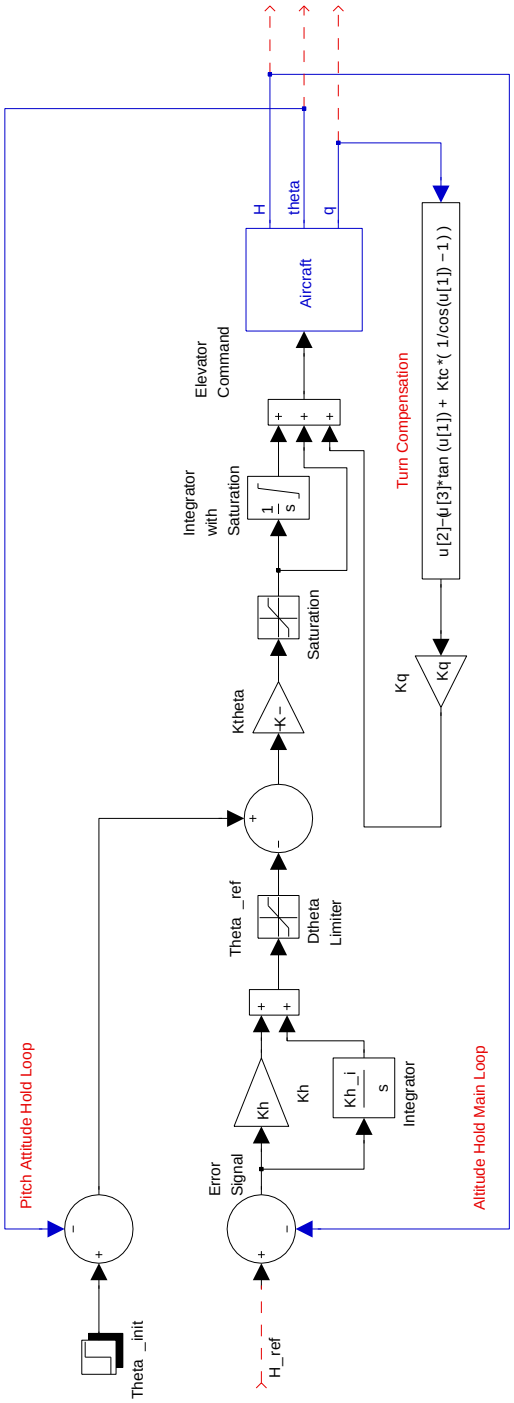
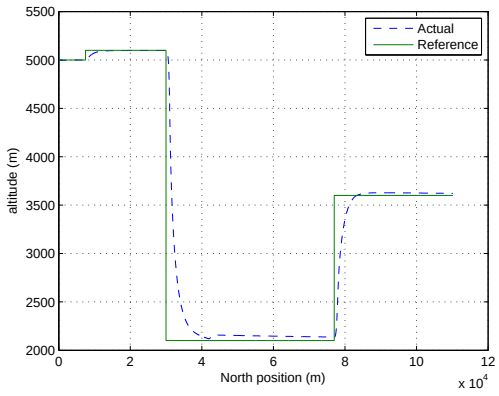
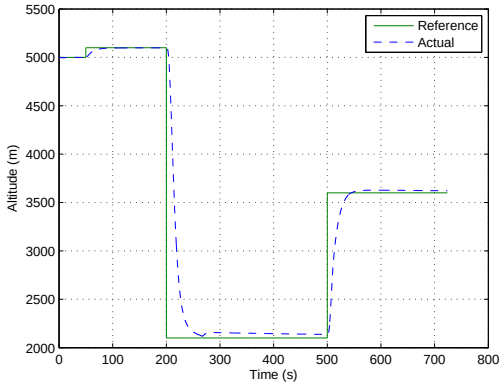


Figure 3.2. Altitude hold mode block diagram.



(a) Trajectory in vertical plane, with North position on x-axis and altitude on y-axis



(b) Altitude response with time

Figure 3.3. Altitude hold mode simulation results.

ϕ, is computed by using the yaw angle feedback, ψ. This reference roll angle, ϕ, is the difference between actual heading and the reference heading, $\Delta\psi = \psi_{ref} - \psi$, multiplied by an amplification factor. The turn coordination loops help in suppressing the sideslip angle and the adverse yaw. Figure 3.4 shows the Simulink implementation for the HH mode with RAH inner loop and turn coordination. The results from a sample simulation over a period of about $500s$ are presented in the Figure 3.5. The series of heading angle changes was $(\pi/2, \pi/4, -\pi, \pi/4)$. Figure 3.5(a) plots the reference and the actual heading angles. It is clear from this figure that the rate of change of the heading angle has been limited to prevent the roll angle from becoming too large. A turn causes loss of lift and the altitude tends to drop. Figure 3.5(b) shows how the turn compensation programmed in the ALH loop helps keep the altitude within bounds. The trajectory of the aircraft as seen from above (horizontal plane) is shown in Figure 3.5(c).

3.5 Navigation and Guidance

A simple autonomous aircraft mission can be defined as a sequence of "waypoints" that the aircraft needs to visit in a given order. This sequence of points can be supplied by the mission planner or generated online during execution. A waypoint is usually characterized by its latitude, longitude, altitude, and velocity, but more complex structure such as maneuvers to be performed on arrival at the desired location, time at which the location needs to be reached etc. can also be established. The navigation strategy implemented here is called "Waypoint Navigation". In this navigation scheme, the aircraft must pass through a sequence of predefined points in 3-D space. The autopilot achieves this using the altitude hold mode from the longitudinal control laws and heading hold mode from lateral control laws. As soon as a particular waypoint is reached, the aircraft attempts to achieve the altitude and velocity for the next waypoint. It is assumed that perfect information about the aircraft states is available. If the aircraft states are not available, they can be estimated from sensors

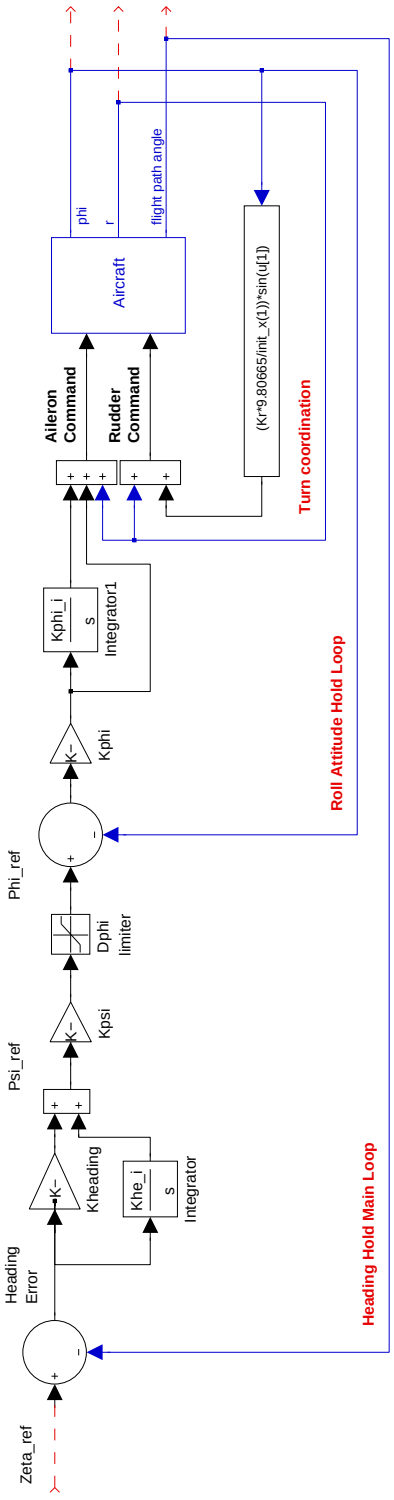
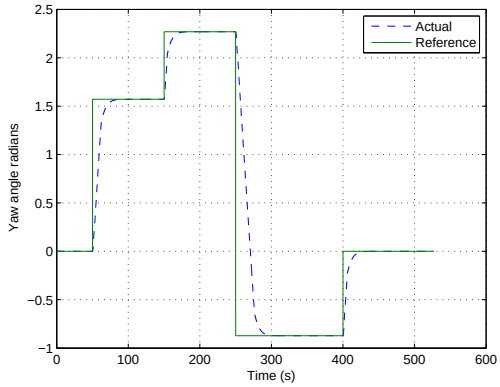
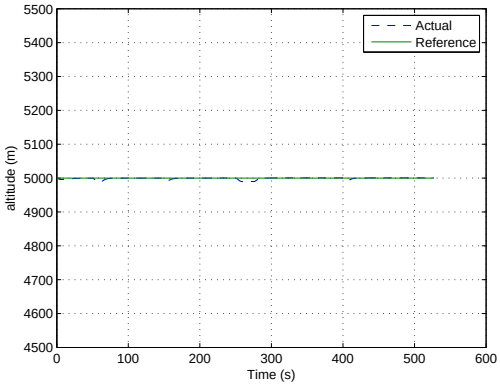


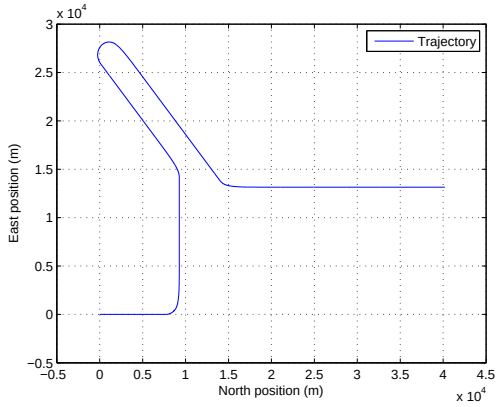
Figure 3.4. Heading hold mode block diagram.



(a) Simulation result for heading hold



(b) Altitude change during turning flight



(c) Trajectory in horizontal plane

Figure 3.5. Turning flight.

such as GPS for latitude, longitude, and altitude information, gyroscopes for sensing angular rates, pressure sensors for measuring altitude and velocity, accelerometer for measuring accelerations, magnetic heading sensors etc. This navigation strategy provides all the necessary inputs for the autopilot to operate.

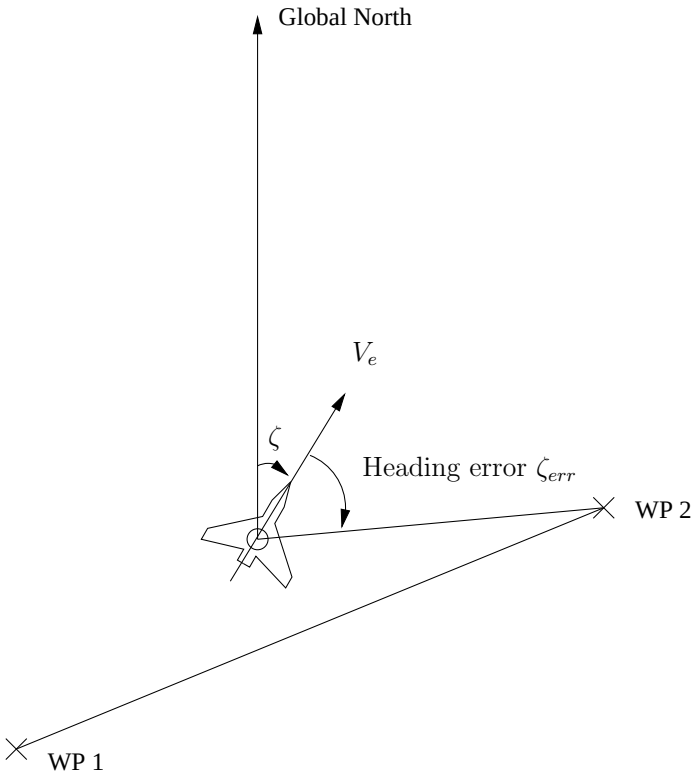


Figure 3.6. Line Of Sight Guidance strategy.

A simple Line Of Sight (LoS) guidance strategy is implemented. According to this strategy, at any instant, the aircraft attempts to follow the line joining its current position and the waypoint to which it is headed. The strategy is explained in Figure 3.6. The heading error, ζ_{err}, is calculated by observing the difference between the actual flight path angle, ζ, of the aircraft and the angle between the line joining the aircraft with the waypoint it is trying to reach and the global north.

When an aircraft reaches within 200m radius of the target point, the waypoint is assumed to have been reached and the aircraft starts heading towards the next waypoint.

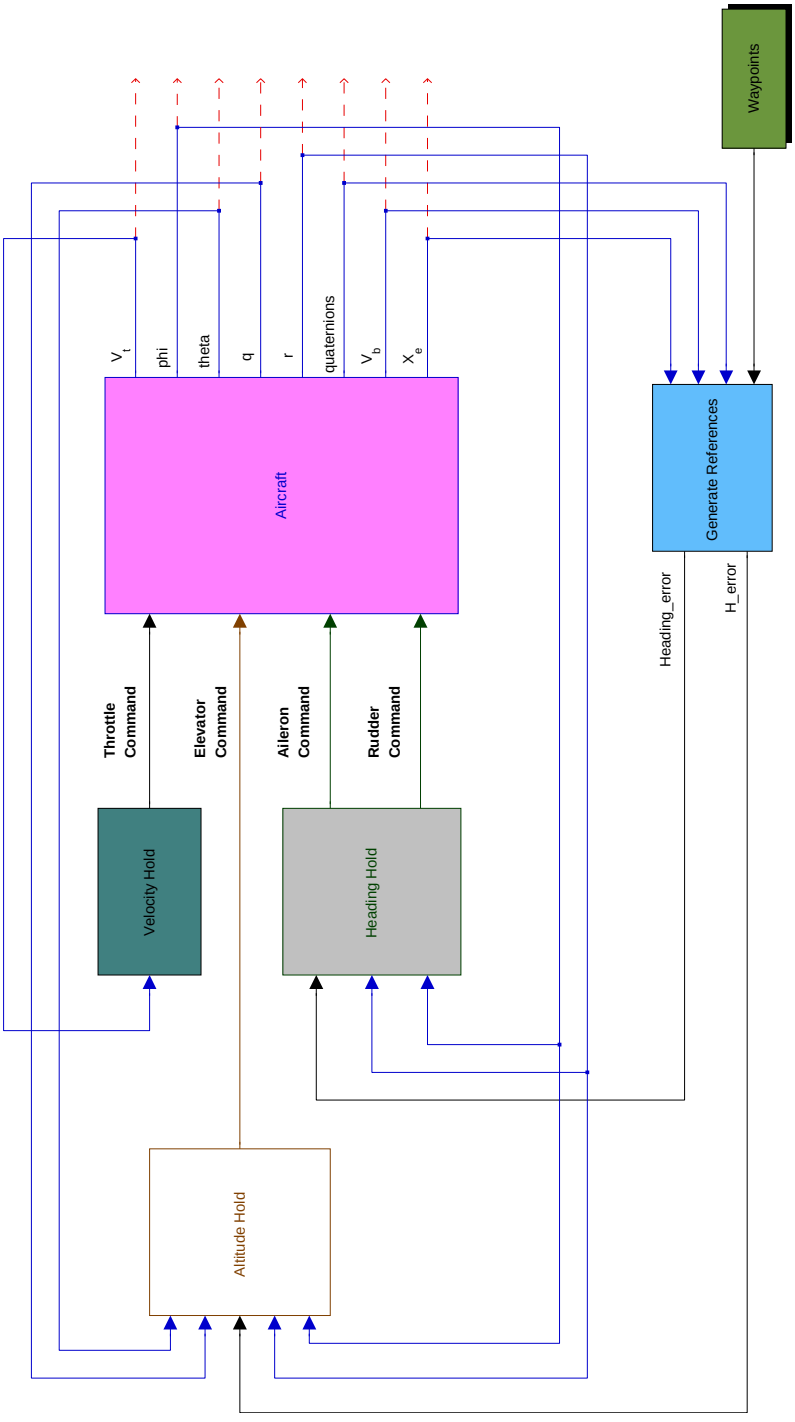
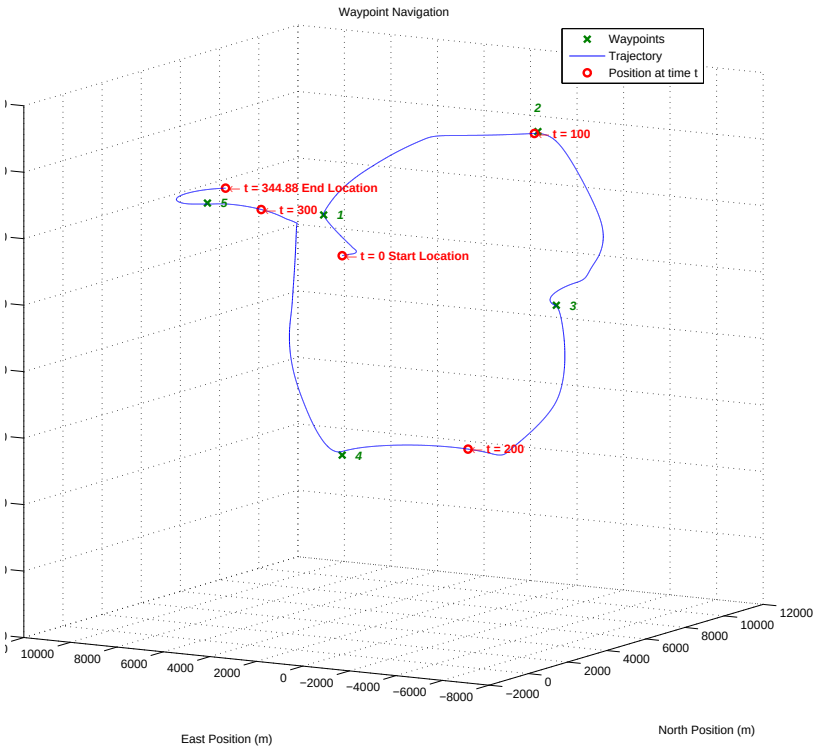
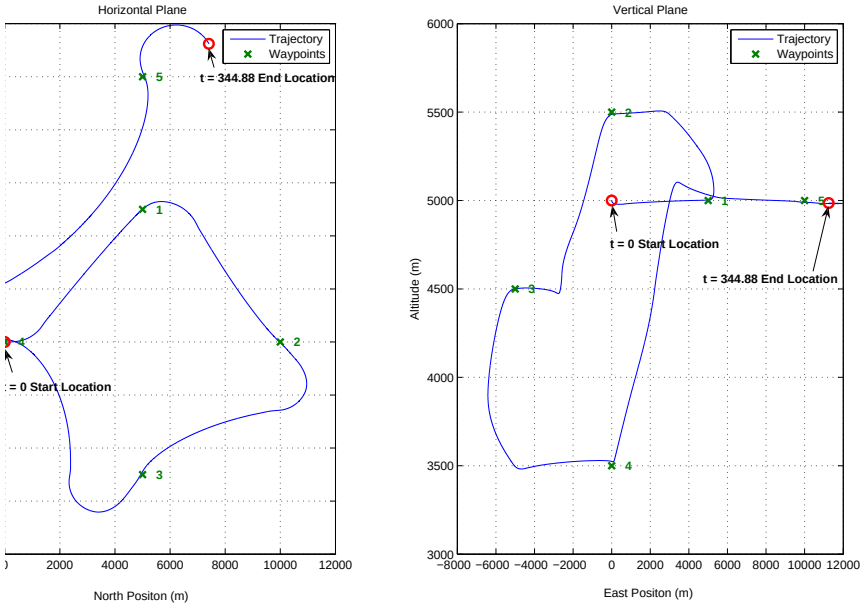


Figure 3.7. Waypoint navigation control structure.



(a) Waypoint navigation simulation results in three-dimensional space



(b) Planar views of waypoint navigation

Figure 3.8. Waypoint simulation for the full nonlinear aircraft model.

A combination of ALH and HH can be used to make the aircraft accomplish a mission defined by a series of waypoints. The block diagram for the entire control setup is shown in Figure 3.7. The functions of each of the blocks in Figure 3.7 are as follows:

1. Aircraft: This block contains the nonlinear equation of motion for simulating an F-16 model as derived in Appendix A.

2. Heading Hold: This block contains the HH mode control and generates commands for the ailerons and rudder.

3. Altitude Hold: This block contains the ALH mode control and generates commands for the elevators.

4. Velocity Hold: This block contains a simple proportional controller to maintain the desired velocity and generates a command for the engines.

5. Waypoints: This block supplies the user defined waypoints to the reference generator.

6. Reference Generator: This block generates altitude and heading errors depending on the current altitude and heading and reference altitude and heading, as supplied by the guidance strategy discussed above.

A simulation was performed with waypoints to be navigated given in Table 3.1. Figure 3.8(a) shows the performance of the lateral and longitudinal control loops in the three-dimensional space. The aircraft starts off from coordinates $(0, 0, 5000)m$ towards north with a velocity of $150m/s$. The controller tries to maintain the velocity at a constant value using the thrust action and no constraints are placed on the time required to reach a certain waypoint. Figure 3.8(b) shows the performance in the horizontal and vertical planes. It is clear that the longitudinal and lateral modes are strongly coupled for this aircraft.

Table 3.1 Waypoints.

North position (m)	East position (m)	Altitude (m)
5000	5000	5000
10000	0	5500
5000	-5000	4500
0	0	3500
5000	10000	5000

3.6 Hydraulic Actuator

We begin by developing a full nonlinear model for a hydraulic actuator as a first step. Figure 3.9 shows the schematic used for developing the mathematical model. The nonlinear equations of motion of a double rod hydraulic actuator are given as [111],

$$m\ddot{x}_L = P_L A - b\dot{x}_L - F_{fc} + \tilde{f} \tag{3.3}$$

$$\frac{V_t}{4\beta_e}\dot{P}_L = -A\dot{x}_L + Q_L \tag{3.4}$$

where x_L is the displacement of the piston, $P_L = P_1 - P_2$ is the load pressure of the cylinder, A is the ram area of the cylinder, F_{fc} represents the modeled Coulomb friction force, and $\tilde{f}$ represents the external disturbances such as unmodeled friction forces. Q_L is the load flow and is related to the spool displacement as follows [111]:

$$Q_L = C_d w x_v \sqrt{\frac{P_s - \text{sgn}(x_v)P_L}{\rho}} \tag{3.5}$$

where C_d is the valve orifice coefficient of discharge, w is the spool valve area gradient, P_s is the supply pressure, and ρ is the density of the hydraulic fluid. x_v is the displacement of the spool of the directional proportional valve with the dynamics expressed as a second order system with natural frequency ω_v and damping ratio ζ_v.

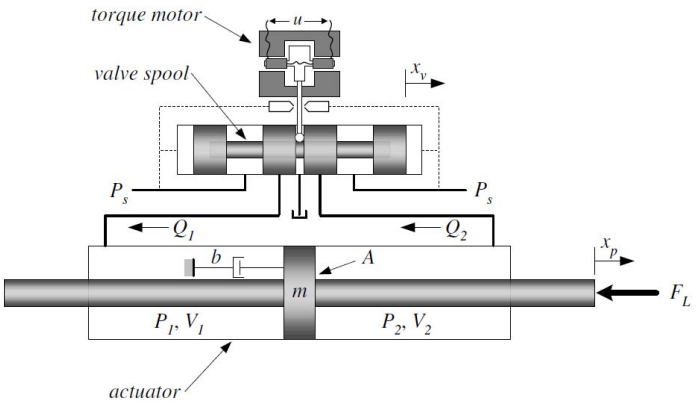


Figure 3.9. Schematic of servovalve controlled hydraulic actuator used for mathematical modeling.

$$\ddot{x}_v + 2\zeta_v\omega_v\dot{x}_v + \omega_v^2 x_v = k_v\omega_v^2 u \tag{3.6}$$

where u is the voltage input to the valve and k_v is the valve gain.

Scaling of state variables can sometimes be very important in minimizing numerical errors and scaling also facilitates gain tuning [112]. For this purpose, new scaling factors for the load pressure and valve opening are introduced as $\bar{P}_L = P_L/S_{c_3}$ and $\bar{x}_v = x_v/S_{c_4}$, where S_{c_3} and S_{c_4} are the constant scaling factors. The system equations (3.3)–(3.4) and (3.6) can now be written as

$$\ddot{x}_L = \frac{AS_{c3}}{m}\left(\bar{P}_L - \frac{b}{AS_{c_3}}\dot{x}_L - \frac{F_{fc}}{AS_{c_3}}\right) + d \tag{3.7}$$

$$d = \frac{1}{m}\tilde{f}$$

$$\dot{\bar{P}}_L = \frac{4\beta_e}{V_t}\frac{S_{c_4}}{\sqrt{S_{c_3}}}\frac{C_d w}{\sqrt{\rho}}\left[- \frac{1}{S_{c_4}\sqrt{S_{c_3}}}\frac{\sqrt{\rho}}{C_d w}A\dot{x}_L \right. \tag{3.8}$$

$$\left. + g_3\left(\bar{P}_L, \bar{x}_v\right)\bar{x}_v\right] \tag{3.9}$$

$$\ddot{\bar{x}}_v = -2\zeta_v\omega_v\dot{\bar{x}}_v - \omega_v^2\bar{x}_v + \frac{k_v}{S_{c_4}}\omega_v^2 u \tag{3.10}$$

where $g_3\left(\bar{P}_L, \bar{x}_v\right) = \sqrt{\bar{P}_s - \mathrm{sgn}(x_v)\bar{P}_L}$ and $\bar{P}_s = P_s/S_{c_3}$. State variables $x = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \end{pmatrix}^T \triangleq \begin{pmatrix} x_L & \dot{x}_L & \bar{P}_L & \bar{x}_v & \dot{\bar{x}}_v \end{pmatrix}^T$ are then defined. The state space form of the system can be expressed as

$$
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \theta_1 \left(x_3 - \bar{b} x_2 - \bar{F}_{fc} \right) + \theta_2 + \tilde{d} \\
\dot{x}_3 &= \theta_3 \left(-\bar{A} x_2 + g_3 x_4 \right) \qquad (3.11) \\
\dot{x}_4 &= x_5 \qquad (3.12) \\
\dot{x}_5 &= 2\zeta_v \omega_v x_5 - \omega_v^2 x_4 + \bar{K}_v \omega_v^2 u
\end{aligned}
$$

where

$$
\begin{array}{lll}
\bar{P}_L = \dfrac{P_L}{S_{c_3}} & \bar{x}_v = \dfrac{x_v}{S_{c_4}} & \dot{\bar{x}}_v = \dfrac{\dot{x}_v}{S_{c_4}} \\
\bar{K}_v = \dfrac{k_v}{S_{c_4}} & \theta_1 = \dfrac{A S_{c_3}}{m} & \bar{b} = \dfrac{b}{A S_{c_3}} \\
\theta_2 = d_n & \theta_3 = \dfrac{4\beta_e}{V_t} \dfrac{S_{c_4}}{\sqrt{\rho}} \dfrac{C_d w}{\sqrt{S_{c_3}}} & \bar{A} = \dfrac{1}{S_{c_4} \sqrt{S_{c_3}}} \dfrac{\sqrt{\rho}}{C_d w} A
\end{array} \qquad (3.13)
$$

The techniques developed in this research are suited to address faults that can be modeled in a parametric form. In most dynamical systems, if a fault affects the physics of the system, then it is possible to find an approximate parametric equation to describe the effects of the fault. The uncertainty in the fault model can be dealt with through a stochastic estimation process. More details on the estimation process are presented in Chapter 5. For example, the internal/external leakage faults in a hydraulic actuator develop due to wearing of the seal material and can be modeled as parametric faults based on equations of the flow through an orifice [111].

1. Internal Leakage

$$Q_{leak} = C_{tm} \sqrt{|P_L|} \operatorname{sgn} P_L \qquad (3.14)$$

2. External Leakages

$$Q_{leak} = C_{em1} \sqrt{P_1 - P_r} \qquad (3.15)$$

$$Q_{leak} = C_{em2} \sqrt{P_2 - P_r} \qquad (3.16)$$

where C_{tm} represents the coefficient of the total internal leakage of the cylinder. Similarly, C_{em1} and C_{em2} represent the coefficients of external leakages for the two chambers of he cylinder. For internal leakage,

$$C_{tm} \propto x_{v_{leak}} \tag{3.17}$$

$$\text{or } C_{tm} = k_q x_{v_{leak}} \tag{3.18}$$

$x_{v_{leak}}$ is a linear function of the area of leakage, a_{leak}. Introducing the internal leakage in the system equations (3.11), the following is obtained:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \theta_1 \left(x_3 - \bar{b} x_2 - \bar{F}_{fc} \right) + \theta_2 + \tilde{d} \\
\dot{x}_3 &= \theta_3 \left(-\bar{A} x_2 - \bar{C}_{tm} \sqrt{|x_3|} \operatorname{sgn}(x_3) + g_3 x_4 \right) \\
\dot{x}_4 &= x_5 \\
\dot{x}_5 &= 2\zeta_v \omega_v x_5 - \omega_v^2 x_4 + \bar{K}_v \omega_v^2 u
\end{aligned} \tag{3.19}$$

where

$$\bar{C}_{tm} = \frac{\sqrt{S_{c_3}}}{S_{c_4}} \frac{\sqrt{\rho}}{C_d w} C_{tm} \tag{3.20}$$

Because of the symmetry of the double rod actuator, the dynamic components of the pressures, P_1 and P_2, in the two chambers of the cylinder are combined in one equation given in (3.4). For introducing the external leakage, the two pressure equations need to written separately since the leakage in one chamber directly affects the dynamics of that chamber only.

$$\frac{V_1}{\beta_e} \dot{P}_1 = -A \dot{x}_L - C_{tm} \sqrt{|P_L|} \operatorname{sgn}(P_L) - C_{em1} \sqrt{P_1 - P_r} + Q_L \tag{3.21}$$

$$\frac{V_2}{\beta_e} \dot{P}_2 = -A \dot{x}_L - C_{tm} \sqrt{|P_L|} \operatorname{sgn}(P_L) - C_{em2} \sqrt{P_2 - P_r} - Q_L \tag{3.22}$$

P_r is the return line pressure.

As the operation of the piston continues, the faulty seal will erode further. This erosion is reflected by an increase in a_{leak}, which leads to reduced performance and eventually complete failure. In the next sections, models for the seal wear as a function of the states of the system are developed.

3.6.1 Wear Model

Few studies are available on the wear rate of a faulty seal in a hydraulic actuator [103, 107, 113]. It is assumed that the seal is made of unfilled PTFE material. An empirical model is implemented in a lookup table as a function of the force acting on the seal, the velocity of the piston, and the distance traveled by the piston. The typical wear behaviour of an unfilled PTFE seal is given in [103] and is shown in Figure 3.10. As is clear from Figure 3.10, the wear rate for a piston with lower velocity for the same displacement and force will be lower. Hence, it might be possible to sacrifice some performance for increased seal life.

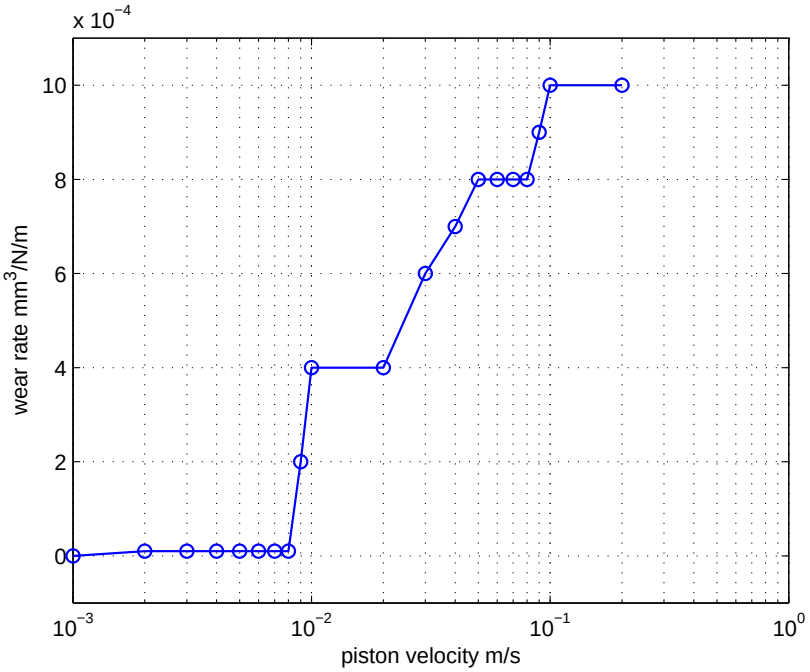


Figure 3.10. Wear rate of an unfilled polytetrafluoroethylene (PTFE) seal [103].

3.6.2 Aerodynamic Force

The aerodynamic distributing load F_L acting on the control surfaces is modeled as in [114]:

$$F_{ext} = 2K_L\bar{q}C_h(\alpha, \delta)M^*(M, \delta) \tag{3.23}$$

where $\bar{q}$ is the dynamic pressure. The nonlinear hinge moment coefficient is captured in C_h and depends on the angle of attack, α, and the control surface deflection δ. A typical lookup table associated with C_h is shown in Figure 3.11. The multiplier M^* captures the effect of the mach number, M.

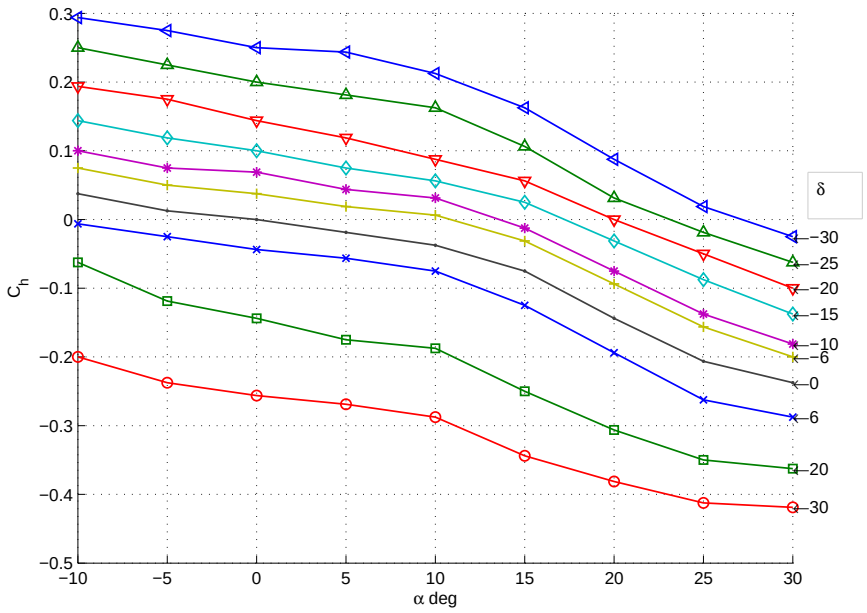


Figure 3.11. Variation of C_h is a function of angle of attack α and control surface deflection δ [114].

3.7 Model Simplification

It is important to validate and verify the control strategies that are developed by implementing them on actual test hardware. Since some of the equipment is not

available in a laboratory setting (e.g. an aircraft), for validation purposes, the aircraft model also needs to be simulated. Thus, the control and optimization strategies developed need to run in real-time together with the aircraft simulation model. Simulating a full nonlinear aircraft model is computationally expensive because of the time step requirements of the hydraulic controller. To ensure real-time implementation, the aircraft model developed in section 3.1 was linearized around an altitude of 10000 ft (3048 m) and a velocity of 500 ft/s (152.4 m/s); and only the longitudinal modes were selected. The state equations thus obtained are given below:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{3.24}$$

where,

$$A = \begin{bmatrix} 0 & 500.0 & -0.0000 & -500.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0.0001 & -32.1700 & -0.0130 & -2.9483 & -1.0283 & 0.0016 & 0.1018 \\ 0.0000 & 0.0000 & -0.0003 & -0.7506 & 0.9281 & -0.0000 & -0.0016 \\ -0.0000 & 0 & 0.0000 & -1.8365 & -1.0271 & 0 & -0.1335 \\ 0 & 0 & 0 & 0 & 0 & -1.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -20.2 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1.0000 & 0 \\ 0 & 20.2000 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\pi}{180} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\pi}{180} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\pi}{180} & 0 & 0 \end{bmatrix}$$

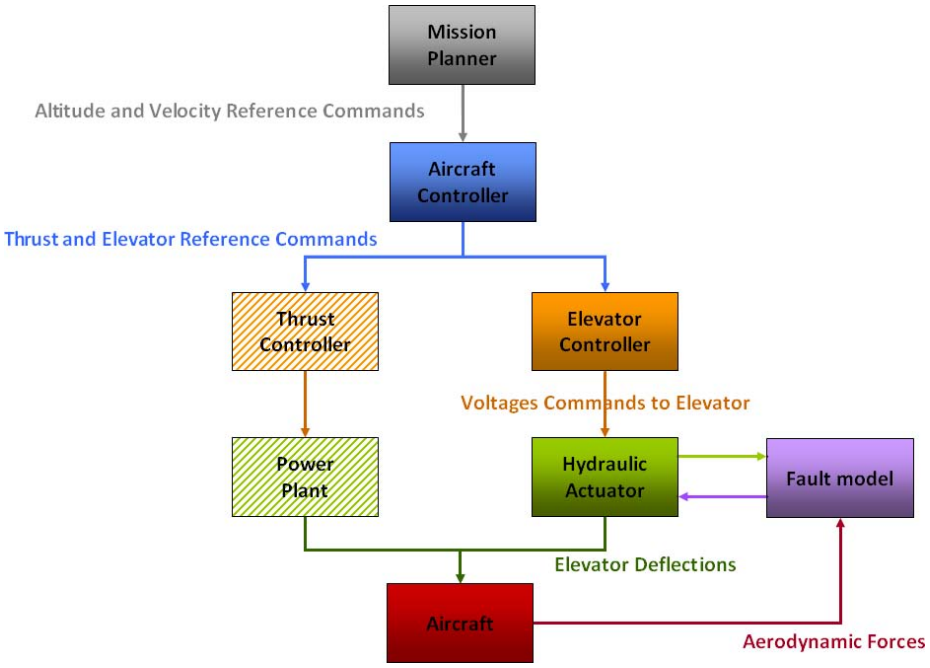


Figure 3.12. Block diagram of the combined models and their interactions.

$x = \begin{pmatrix} h & \theta & v_t & \alpha & q & \delta_t & \delta_e \end{pmatrix}^T$, $u = \begin{pmatrix} u_t & u_e \end{pmatrix}^T$. The description of states and inputs is given in Tables 3.2 and 3.3. The combined model of the aircraft, the actuator, and

Table 3.2 States.

Symbol	Description
h	altitude
θ	pitch angle
v_t	speed
α	angle of attack
q	pitch rate

Table 3.3 Inputs.

Symbol	Description
δ_t	thrust
δ_e	elevator deflection
u_t	input to engine
u_e	input to elevator spoolvalve

the fault is shown in Figure 3.12. The arrows show the direction of information and signal flow in the model. This diagram will be revisited in Chapter 5 dealing with control development and implementation. It should be noted that only the aircraft

model is linearized; all other nonlinearities such as rate limits, saturation limits, and delays due to aircraft actuators are still present. Furthermore, the hydraulic actuator model is a fully nonlinear model.

4. EXPERIMENTAL SETUP

This chapter will present an overview of the hardware-in-the-loop simulation, its components, and its requirements. A detailed design of the experimental setup will be presented followed by the implementation process and system identification procedure.

4.1 Hardware-in-the-loop

Full software simulation of a system typically requires a high fidelity model of all the system components. In some cases, subsystem models cannot be adequately characterized by mathematical models. For example, actuator dynamics are often hard to model with nonlinearities such as friction, dead zone, hysteresis, etc.; embedded microcontrollers with a resident code are also difficult to take into account. In such cases, it is appropriate to embed the actual system as it is into the simulation environment creating a hardware-in-the-loop simulation (HILS). Use of HILS, where actual hardware is embedded into the simulation, was first investigated in aircraft and space applications, and is slowly percolating to other industries. Onboard-controller-in-the-loop simulation (OILS) is also a popular strategy. Often HILS is also used to include OILS. When hardware devices or humans are embedded into a simulation, the simulation of all its components must proceed in "real-time". The words "real-time" indicate the simulation of each component is performed such that input and output signals show the same time dependent values as in real world dynamic operation.

The basic principle of HILS is that some subsystems are physically embedded within a real-time simulation model. In HILS, the embedded system can be thought of as being fooled into thinking that it is operating in the real-world with real inputs and outputs, in real-time. A software with real-time simulation capabilities and

hardware with necessary communication abilities (A/D,D/A converters for communications with analog signals and digital ports for communication with digital signals) are both necessary to perform HILS [115]. While performing HILS for a real sys-

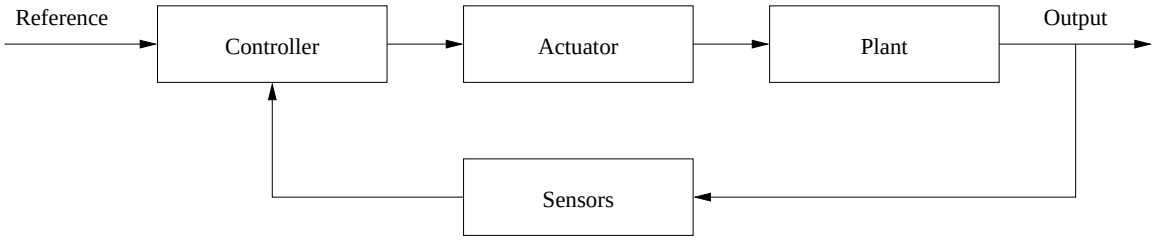


Figure 4.1. Block diagram of an actual system.

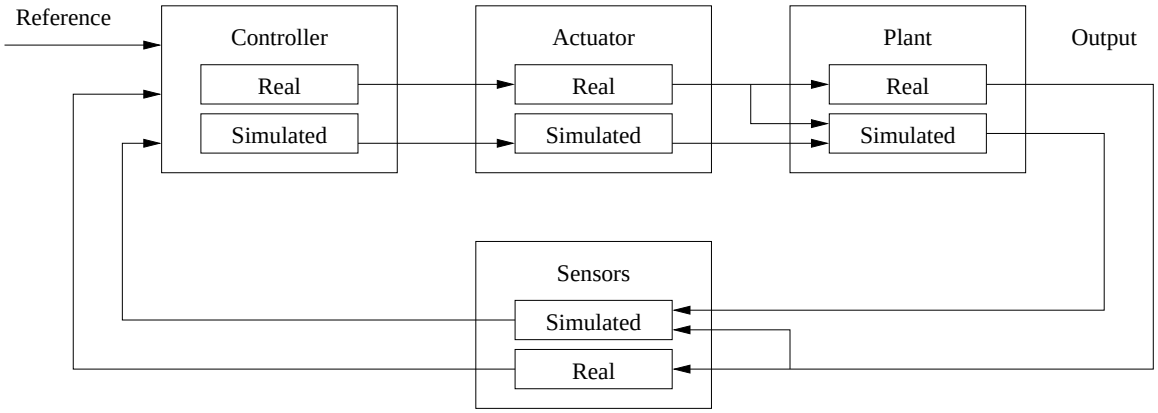


Figure 4.2. Hardware-in-the-loop simulation of system shown in Figure 4.1.

tem, control system hardware and software usually comprise the real system. The controlled process consisting of physical systems and sensors can then be either fully or partially simulated. Figure 4.1 shows the block diagram and signal flow for a real system and Figure 4.2 shows some possible combinations of HILS for this system. Frequently, some actuators are real, and the process and sensors are simulated. The reason is that actuators and control hardware often form one integrated subsystem. Also, actuators are difficult to model precisely and simulate in real-time. The use of

real sensors together with the simulated process may require considerable realization efforts because no real sensor input exists and it must be generated artificially. When the actual controller is included in the hardware in HILS, the simulation is called an onboard-controller-in-the-loop simulation (OILS) or OBC-in-the-loop simulation.

4.1.1 Advantages of HILS

Hardware-in-loop-simulation studies are frequently used in some industries (e.g. aerospace, automotive, and chemical) for testing system components when full system tests may be limited in scope because of constraints in terms of safety or availability of a complete system for testing purposes. The various advantages of doing hardware-in-the-loop simulations are as follows [116]:

- Design and testing of the control hardware and software without the need to operate the real process (moving the process field into the laboratory)
- Testing of the control hardware and software under extreme environmental conditions in the laboratory (e.g. high/low temperatures, high accelerations and mechanical shocks, aggressive media, electromagnetic compatibility)
- Testing of the effects of the faults and failures of actuators, sensors and computers on the overall system
- Operating and testing of extreme and dangerous operating conditions
- Reproducible experiments, frequently repeatable
- Easy operation with different man-machine interfaces (e.g. in cockpit-design and in the training of operators)
- Saving of cost and development time

4.1.2 Historical Development

The first approaches to HILS were first realized for real-time simulation of instruments with a fixed cockpit linktrainer in 1936, and later on to move a cockpit according to aircraft motions for the training of pilots. Here, the cockpit and the pilot were real, and the motions were generated by electrical and hydraulic actuators. The first generation HILS used analog tube controllers and analog motion simulations, which were subsequently replaced by analog computers and then digital computers in 1953.

HIL motion simulators were also built for the dynamic testing of vehicle components (e.g. suspensions, bodies) with hydraulic or electrical actuators (testing machines); for example, to simulate the excitation of the wheels by a road surface. Another interesting type of HIL motion simulation is vehicle driving simulations. Dynamic motor teststands, where the engines are real and the vehicles and gears are simulated by some other hardware (an electrical DC or AC motor) together with a digital computer, are a special kind of HIL simulator.

With the development of digital electronic control systems for vehicles, for example, ABS (Antilock Braking Systems) for brakes, and TCS (Traction Control System) for drive chains and automatic gears, the associated HIL simulators followed various stages of development. First versions used high-performance workstations and process computers. However, the amount of real-time simulation was very limited. The availability of parallel computers, in the form of transputers, Reduced Instruction Set Computers (RISC) processors with onchip RAM, high-speed communication links, and more efficient Digital Signal Processors (DSP) then opened the way for real-time simulation of complete hydraulic systems, sensors, actuators, and suspension systems. Further research showed how more complex, comprehensive mechanical systems can be simulated in real-time by parallel computers; examples include multi-body systems, brake systems, and combustion engines. Some of these HIL simulations were in-house developments within companies, especially in the fields of aircraft and auto-

mobiles. Typical configurations consist of either transputer or DSP cards, a host PC and special processes, sensors, and actuators [116].

4.1.3 HIL System Components

An HIL simulation requires three main components to function: the simulation model, the control system and the input/output interface. The basic breakdown of these components is given below:

1. Simulation model: For real-time simulation, dynamic models for the subsystems are required. Several compromises have to be made with regard to model complexity and calculation time. There are two main ways of developing mathematical models [116]:

   (a) The first method is theoretical modeling based on physical laws and expressed by equations. After simplifying assumptions, these equations are stated for single process elements. They can be divided into balance equations for mass, energy and momentum, constitutive equations for sources, transformers and converters, and phenomenological equations for irreversible processes like dissipative elements or sinks. The interconnections of the process elements are described by continuity equations (node law) and compatibility equations (closed circuit law). Based on these equations, an overall model can be developed. For lumped-parameter systems, the process model can be represented in either state-space form or input/output form, i.e. differential equations or transfer functions. For distributed-parameter systems, in general partial differential equations are obtained.

   (b) The second method is experimental modeling or model identification. The model structure and parameters are extracted from input/output signals through minimization of error between the process and the model. The advantage of experimental modeling is that less time is consumed in the

development of the process. However, the quality of the models is highly dependent on the use of appropriate model structure assumptions, and on the measurement data used.

2. Control system: The control system developed for HILS applications is almost always the real system. This is because it is easy to transfer a control system from a test setup to a real process model after extensive testing, provided the implementation hardware is comparable. It can also help in the selection of hardware based on the computational, input/output, and memory storage requirements [117]. The control strategy is usually formulated and tested in an offline simulation environment such as Simulink or LabView. This is called Software-in-the-loop (SITL) configuration. The control source code can also be compiled into a pure software simulation environment allowing testing and validation without tying the hardware.

3. I/O card and signal interface: The simulated sub-systems (digital domain) in the HIL simulation need to exchange data with hardware (analog domain) or real sub-systems. This can be achieved using I/O cards and other interfaces like serial and parallel port. Depending on the chosen HIL simulation scenario different types of analog signals must be converted to digital domain and vice versa. A/D and D/A converters are used to give signals to the controller exactly in the same form as it will get if connected to the real process. The main roles of signal interface design are [118]:

   (a) to adjust the voltage range of the selected signals to the voltage range of A/D and D/A converters

   (b) to transform signals from differential to single-ended formats and vice versa

   (c) to electronically isolate appropriate physical subsystems which are simulated or not used from the other physical subsystems which are included in the selected HIL simulation scenario

4.1.4 HIL System Requirements

The components of the HIL system must satisfy following requirements in order to generate realistic simulations and results:

1. Simulation model: The following are the requirements caused by simulation model [119]:

   (a) The simulation model should include all essential dynamics of the real system. Usually the accurate model is computationally expensive and the computational power of the HIL hardware must be high

   (b) The required computational power depends on the complexity of the model, the integration algorithm and the simulation time step

   (c) A suitable time step can be determined by off-line simulation by comparing the responses of the simulation model with different time steps. The time step depends also on the integration algorithm such that longer time steps can be used with higher order algorithms

   (d) Numerical accuracy in the off-line and real-time simulation can be different because of different floating point numbers

   (e) If single precision is used then very small time steps should be avoided, because it can cause errors

2. I/O card and signal interface: The requirements caused by analog sensors are moderate. Analog measurement signals can be generated from the model by DA-converters. The resolution of the DA-converter must be at least as high as the resolution of the AD-converter of the controller. Similarly, the analog output generated by the controller can be imported into the simulation model via AD-converter. The resolution of the AD-converter of the HIL hardware must be at least as high as resolution of the DA-converter of the controller card [119].

3. Selection of time step: The following points need to be remembered [119, 120]:

   (a) The simulation time step must be such that numerical integration is accurate enough

   (b) Simulation time step also depends on the sample time of the controller. Sampling causes an error which has a maximum value equal to the simulation time step. In order to keep the error small, the simulation time step is at maximum one fifth of the controller sample time. This rule can be relaxed if the controller is not sensitive to time delays

   (c) If sufficiently small simulation time cannot be achieved, the simulation model must be simplified

4.2 Experimental Setup

This part of the chapter will provide details of the experimental setup used in the HIL simulations designed to validate the control strategies. We begin by giving the requirements the experimental setup must fulfill. A design will then be presented based on these requirements.

4.2.1 Requirements

Since it is not possible to test an actual aircraft in the laboratory setting, we choose the hydraulic actuator which powers the control surfaces, the sensors, and the controller as the real system. The software simulation contains the aircraft model and generates commands for the actuator. Thus the processor must be capable of running both the controller and the aircraft model in real-time. An aircraft actuator experiences varying aerodynamic loads depending on various aircraft states. The experimental setup must be able to simulate these realistic distributed aerodynamic loads acting on the control surfaces (Section 3.6.2). And finally the design must also facilitate introduction of different types of faults as listed in Section 2.2.4.

4.2.2 Components

Based on the above requirements, the components required to realize the HIL simulator are:

1. Hydraulic cylinders which would act as the control surface as well as to apply aerodynamic loads,
2. Directional proportional valves to control the hydraulic cylinders,
3. Position, pressure, temperature sensors, and flow meters to measure desired states and supply them to the controller,
4. Hydraulic power unit to supply fluid at rated pressure and flow rate,
5. Real-time processor to run the aircraft model and the controller, and
6. A visualization and control interface for interaction.

4.2.3 Mechanical Design

Using the requirements and the components, a design is proposed as shown in Figures 4.3 and 4.4. Figure 4.3 shows the hydraulic circuit diagram for the control surface actuator loop. This is the main loop in which the position and velocity are controlled. All the faults and failures are simulated in this loop. The loads are simulated using another circuit as shown in figure 4.4. These cylinders of these two circuits are connected end to end using a load cell. These two circuits are powered using the same hydraulic power supply. Figure 4.5 shows a picture of the experimental setup. The left part is the actuator cylinder and the right side the load cylinder. The system has the ability to simulate following faults:

1. leakage between the two hydraulic actuator chambers,
2. partial/complete blockage between supply and return lines,

3. proportional control valve malfunction,
4. changes in the rated supply pressure,
5. changes in the rated supply flow rate,
6. changes in the cylinder friction and damping coefficients,
7. external leakages,
8. internal leakage in the pump,
9. cracked actuator resulting in limit on the maximum force that can be applied by the actuator.

User-defined functions can also be implemented on some the fault producing components to simulate fault growth. A glossary of symbols used in this figure is given in Appendix B. The description of the mechanical components labeled in Figures 4.3 and 4.4 and their necessity is explained below:

1. Hydraulic power unit (Label 1) is a combination of a hydraulic pump driven by a prime mover usually an electric motor. The purpose of this component is to provide a constant supply of hydraulic fluid either at constant pressure and/or a constant flow rate. This application will have a pressure compensated variable displacement axial piston pump. Pressure compensation is required because the pump must power two circuits at the same time: (a) the actuator circuit and (b) the load circuit. Variable displacement is needed to avoid power loss when the load requires less than full flow or full pressure. The maximum system pressure will be limited to 100 bar and flowrate to 20 litres per minute. The maximum system pressure should be around 60% of what the pump can supply to accommodate unexpected demands [121]. The pump is rated at $300 bars$ at 60 litres per minute. The unit can be used to introduce fault numbers 4 and 5 electronically.

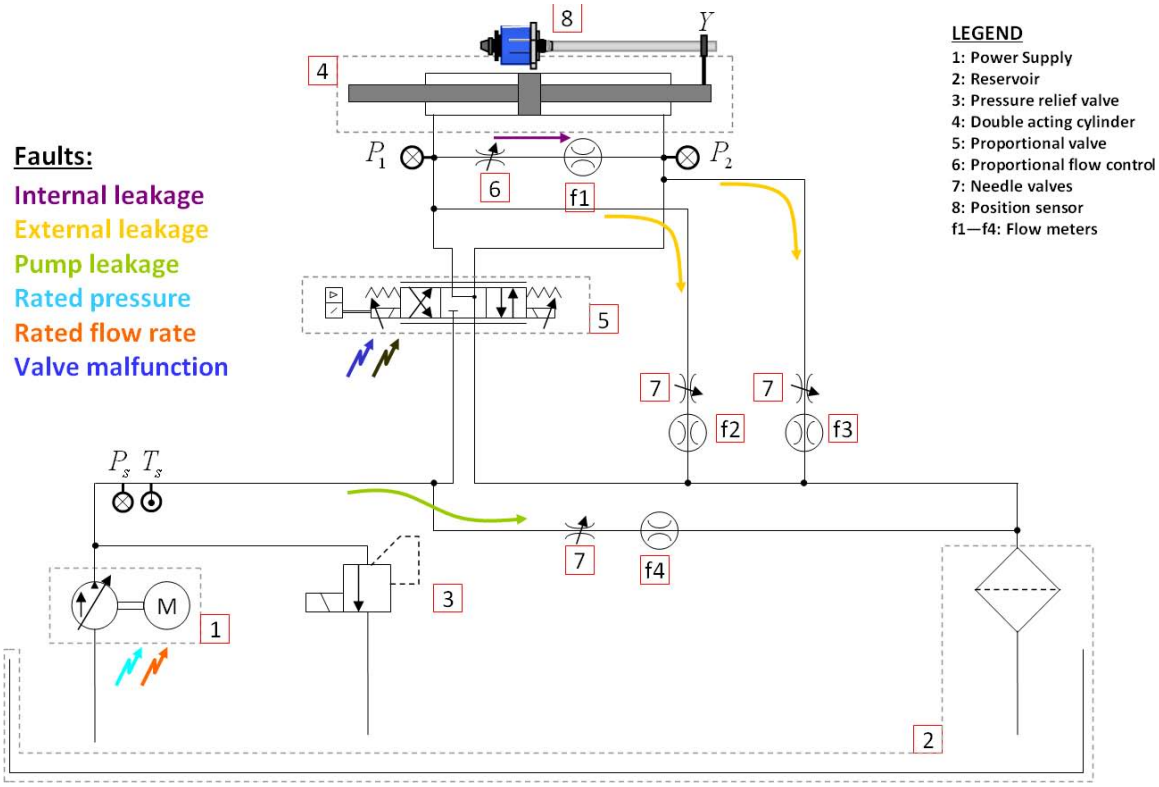


Figure 4.3. Schematic of the actuator circuit used to simulate the aircraft control surface. This circuit forms part of the hardware in the HIL simulation and is used to simulate different hydraulic fault.

2. Hydraulic fluid storage (Label 2) is the component where the fluid is stored and conditioned before being picked up by the pump. The conditioning may include cooling, filtering, reducing turbulence etc. The size of the storage tank varies according to the application. It is proposed to use a 100 liter storage tank for this study.

3. Pressure relief valve (Label 3) is a proportional solenoid operated valve used to limit the system pressure and prevent excess pressure buildup from damaging expensive equipment. The valve thus must have a maximum pressure limit greater than the maximum system pressure. This component can also be used to maintain a system pressure lower than that supplied by the pump. The

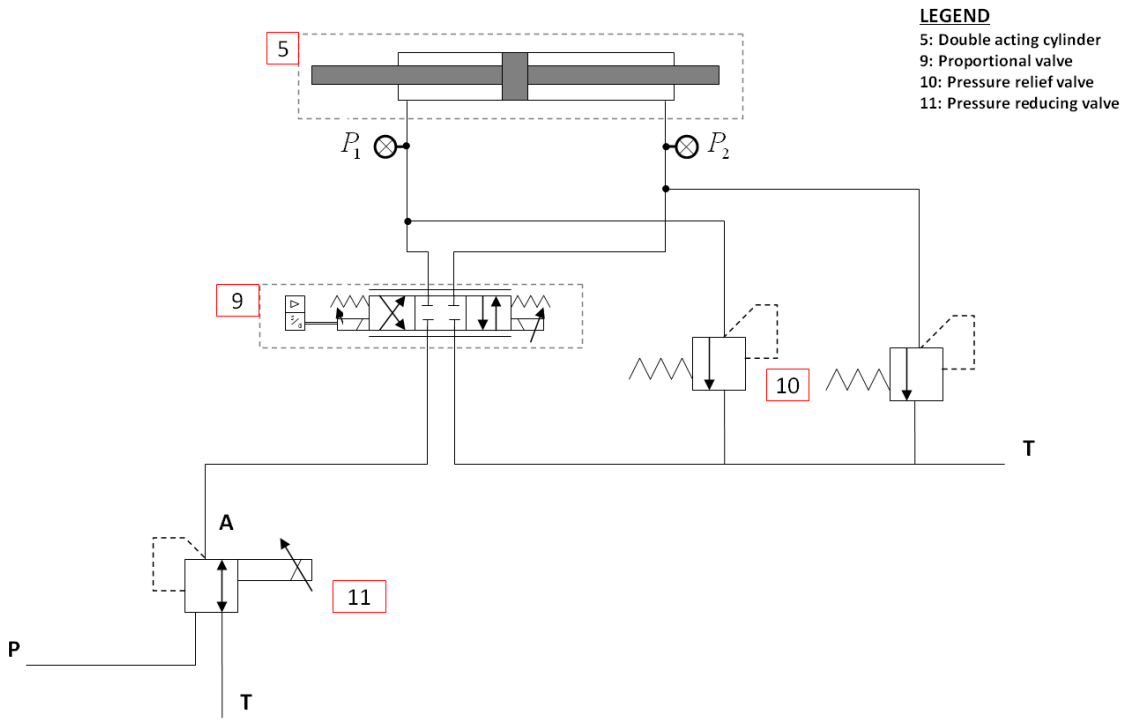


Figure 4.4. Schematic of the load circuit used to simulate aerodynamic forces on the aircraft control surfaces.

valve is a safety feature, but can also be used to introduce fault number 4. The pressure relief setting is adjusted to about 200 psi above the pump pressure setting.

4. Hydraulic cylinder (Label 4) is a double acting double rod cylinder. The specifications are based on the maximum force that the actuator needs to apply and the speed with which the actuator must move the load. In the laboratory setup, the maximum force will be limited to 10 kN and the speed will be limited to 0.5 m/s. The internal and external diameter of the cylinder are 28 mm and 40 mm respectively. The stroke is 300 mm.

5. Directional proportional valve (Label 5) is a direct solenoid operated valve used to control the flow of the hydraulic fluid and in turn control the position and

Figure 4.5. Picture of the experimental test bench showing both the load (right of the center) and actuator circuits (left of the center). The load and actuator cylinders are connected with a load cell.

velocity of the actuator cylinder. The valve must meet the following requirements: (a) it must supply the position signal of the spool so that a feedback loop can ensure the correct spool position; (b) when there is any kind of failure, the valve must allow free movement of the actuator against load, and; (c) it must allow multiple loads to be operated in parallel. For this reason, the center or the default position of the valve must be float center type, i.e., the ports connected to the actuator cylinder must drain to a tank and the pump port must be blocked [121]. Fault number 3 can be simulated electronically using this component. This valve is a very high precision valve with a response bandwidth of up to a 100 Hz. This was chosen to allow precise and fast position control.

6. Flow control (Label 6) is an electronically operated directional proportional valve. It connects the two chambers of the actuator cylinder and is used to

model internal leakage due to a faulty seal (Fault number 1). This is also a precision, high bandwidth valve to allow precise control over the internal leakage fault.

7. Flow control (Label 7) valves are simple needle valves used to control the flow rate in corresponding lines. They can be used to simulate external leakage at the two ports of the actuator cylinder (Fault number 7) or to limit the flow rate to the system (Fault number 2) or to simulate internal leakage inside the pump (Fault number 8). These valves cannot be controlled electronically.

8. Directional proportional valve (Label 9) is another direct solenoid operated valve just like the valve labeled 5. The only difference is that this valve forms a part of the load circuit and the center position is a closed center. In this type of valve, all 4 of the ports are blocked. This center position ensures that the pump port is blocked so that multiple circuits can be run in parallel [121]. In the initial design a cheaper lower bandwidth valve was utilized in the design with a bandwidth of 20 Hz. This is because precision control is not required in the load part of the circuit.

9. Pressure relief valve (Label 10) are incorporated to ensure that excess pressure does not build up in the two chambers of the load cylinder. This component is a safety feature.

10. Pressure reducing valve (Label 11) is a proportional solenoid operated valve. This valve is connected at the inlet of the load circuit to reduce the maximum pressure in the load circuit compared to that in the main actuator circuit. This arrangement is to ensure that load does not apply force that cannot be overcome by the actuator.

4.2.4 Electrical Design

The mechanical part of the system must integrate seemly with the electrical part to allow precise measurement and control of the moving parts. A real-time processor is also required to run simulation models, identification and control algorithms. Furthermore, an interface is needed in form of A/D, D/A cards for the analog part including sensors and actuators to interact with the digital part which includes the processor. Following list describes the electrical subsystem design :

1. Flow meters (Label f1–f4), as the name suggests is used to measure the flow rate through the hydraulic line. Flow rate measurements are essential to determine the valve parameters and leakage fault sizes. The flow meters ranges have been selected to ensure maximum accuracy in the entire operating range (0–8 ltrs/min).

2. Position sensor (Label 8) is fixed to the hydraulic cylinder so that precise position measurement can be performed and utilized in the feedback control algorithms. The range of the position sensor is chosen so as to cover the complete stroke of the hydraulic cylinder. The theoretical position accuracy is about 0.1 mm.

3. Pressure sensors are needed to measure supply line pressure, return line pressure, and pressures in the chambers of the cylinders. The pressure sensors utilized have an operating range of 0-200 bars.

4. Temperature sensors are needed to track the oil temperature. The physical properties of the hydraulic oil such as bulk modulus, viscosity, etc., have high sensitivity to temperature. High temperature also degrades the oil faster reducing useful system life and increasing the probability of failure.

5. Load cell is used to measure the forces exerted by the load circuit on the actuator circuit and use the measured value in the force feedback control loop.

6. Real-time processor is need to run the aircraft model, the fault identification algorithm, and the control strategies. The size of the processor and the on-board memory should be sufficient for real-time operation. A dSPACE DS1005 processor board is used for the current application. The board provides sufficient bandwidth to execute the required programs in real-time and the capability to expand if required.

7. I/O cards are required for the digital domain and the analog domain to interact. The number and type of outputs and inputs from the processor are give in Tables 4.1 and 4.2. The I/O card consists of 32 channel A/D single ended

Table 4.1 Inputs to the data acquisition system, which are outputs of the experimental setup.

	Sensor	Output Type	Output min (V)	Output max (V)	Nos.
Inputs	Flow meter	Digital	-25	25	3
	Position	Analog	0	10	4
	Pressure	Analog	1	5	5
	Temperature	Analog	0	5	2
	Load cell	Analog	0	.020	1

board with a ±10 V range and 8 channel digital I/O board with TTL (±5 V) range for the inputs and a 32 channel D/A board with a ±10 V range for outputs. The output of the load cell and the flow meters is not compatible with the I/O card. An instrumentation amplifier was constructed using INA125 to amplify the load cell output to 0–10 V and a voltage comparator was constructed using LM339 to bring the ±25 V square wave signal down to TTL level. Figure 4.6 shows the I/O cards.

The block diagram representing the HIL implementation is shown in Figure 4.7.

Table 4.2 Outputs from the controller, which act as inputs to the experimental setup.

	Component	Input Type	Input min (V)	Input max (V)
Outputs	Directional Proportional Valve (actuator)	Analog	−10	10
	Directional Proportional Valve (load)	Analog	−10	10
	Directional Proportional Valve (leakage)	Analog	−10	10
	Pressure Relief	Analog	0	10
	Pressure Reducing	Analog	0	10

Figure 4.6. Picture of I/O box for the experimental setup. Top box has a D/A convertor and a digital I/O, while the bottom box hosts A/D convertors.

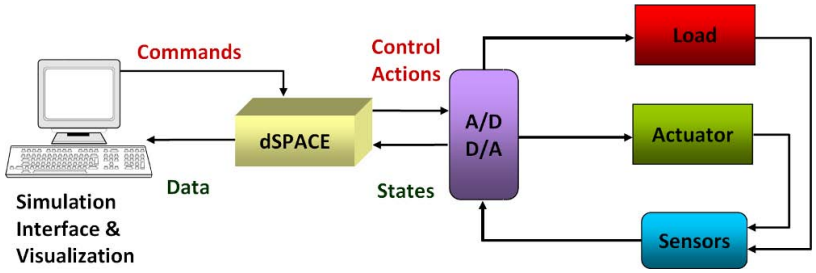


Figure 4.7. Diagram depicting layout of different blocks to accomplish the HIL implementation.

4.3 Implementation Steps

To realize the application undertaken in this research following general steps were followed:

1. Theoretical modeling
2. Experimental setup design
3. System identification
4. Control development
5. Parameter tuning

Figure 4.8 shows the block diagram of the steps followed. The first two steps have already been described. The rest of the chapter describes the system identification process.

4.4 System Identification

System identification is a general term to describe mathematical tools and algorithms that build dynamical models from measured data. A dynamical mathematical model in this context is a mathematical description of the dynamic behavior of a system or process in either the time or frequency domain. This process is important

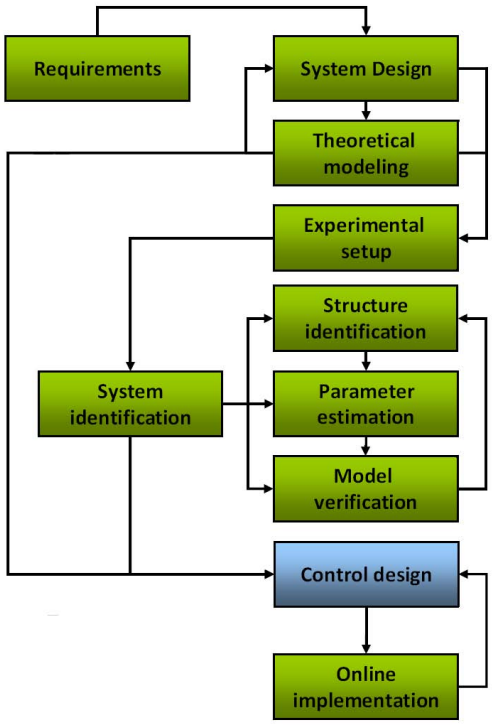


Figure 4.8. Schematic of the workflow followed to complete the implementation.

during the controller development stage to obtain an reasonable accurate mathematical model, so that thorough testing of the controller can be performed without the risk of damaging expensive equipment. This would also ensure that when the controller is implemented on the actual system, no unanticipated behaviour is observed. This section describes a general system identification procedure and its application to the hydraulic actuator setup described above [95]: A flowchart of the system identification process is shown in Fig. 4.9. A series of experiments can be carefully designed so that the most important properties are clearly observable. Some of the requirements and issues with performing the experiments are discussed here.

4.4.1 Pre-identification

A pseudo-random binary signal (PRBS) is widely used in system identification for exciting linear systems. PRBS signals are periodic signals containing two amplitude

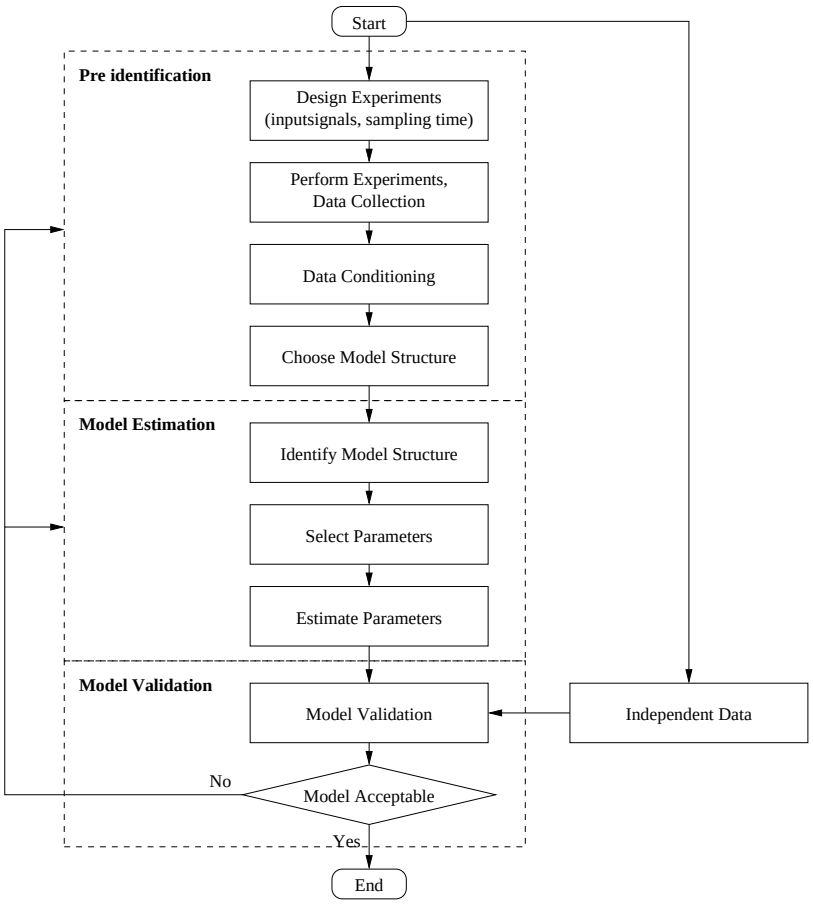


Figure 4.9. Schematic of the system identification process.

levels that closely imitate white noise. Thus, these signals excite all frequencies and are not correlated with noise signals resulting in bias free estimates. However, in the case of nonlinear systems, these signals fail to excite all the amplitudes and important information might be lost. For such systems, other types of random excitation signals are recommended:

- independent sequences with a Gaussian uniform distribution,
- Pseudo-random multi level signal (PRMS), where the level is changed at each N_s^{th} sampling instant, or
- chirp signals.

For most of the identification process, the pseudo-random multi level signal was used.

4.4.2 Sampling Interval

The choice of sampling interval is a trade-off. A very small sampling time can give large amount of data in small time intervals, but can introduce problems due to numerical ill-conditioning of estimation parameters, or can cause the system to become non-minimumphase. On the other hand, a large sampling interval will contain very little information about high frequency dynamics, but will allow smoother, rapid tracking will smaller control effort. As a rule of thumb, sampling interval is chose to be 10-20% of the settling time of the step response. For hydraulic systems, the usual range selected is in the range of 1-2 ms.

4.4.3 Data Conditioning

Raw data often needs to be conditioned before processing. Conditioning may include filtering, removing outliers, removing mean, scaling for numerical robustness etc.

4.4.4 Model Structure

System models based on complete knowledge of the process are termed “white box” models, whereas those based principally on experimental data are called “black box” models. In the case of some systems, physical insight is available; however, several parameters remain to be determined. Such systems fall under the category of “grey box” model and system identification is needed to estimate these parameters. A detailed nonlinear model of the hydraulic actuator was given in Chapter 3. The dynamics of the three directional proportional valves, used for controlling the actuator, the load, and the leakage, are identified first followed by identification of the hydraulic parameters.

4.4.5 Directional Proportional Valve

All the valves were modeled as linear second order systems from voltage input to the valve displacement output [114, 122].

$$\frac{x_v}{u} = P_v = \frac{k_v \omega_v^2}{s^2 + 2\zeta_v \omega_v s + \omega_v^2} \tag{4.1}$$

where, u is the voltage input, x_v is the valve displacement, k_v is the servo valve gain, ζ_v is the damping, and ω_v is the natural frequency. Since the model is known, we use model based identification. In model based identification the error between the outputs of the experimental result and the simulated result is minimized using an optimization over the unknown parameters. The parameter values obtained are then corroborated with the values given by the manufacturer. The initial values of the parameters for the optimization were assumed to be

$$k_v = 1 \qquad \omega_v = 15 \qquad \zeta_v = 1$$

An example trajectory of the parameters during optimization is shown in Figure 4.10. Figures 4.11, 4.12, and 4.13 shows the experimental and estimated responses of the actuator, the leakage, and the load directional proportional valves based on PRMS. The estimated parameters are listed in Table 4.3.

Table 4.3 Comparison of estimated parameters and manufacturer specifications for the directional proportional valve models.

Valve	k_v	ζ_v	ω_v for 100% amplitude (Hz)	Manufacture reported bandwidth for 5% input (Hz) amplitude
Actuator	1.0039	1.0027	40.78	100
Leakage	1.0043	0.75	38.7	100
Load	0.9679	1.2599	6	20

As can be clearly noted from above figures, the second order model is able to approximate the response of the valves very well.

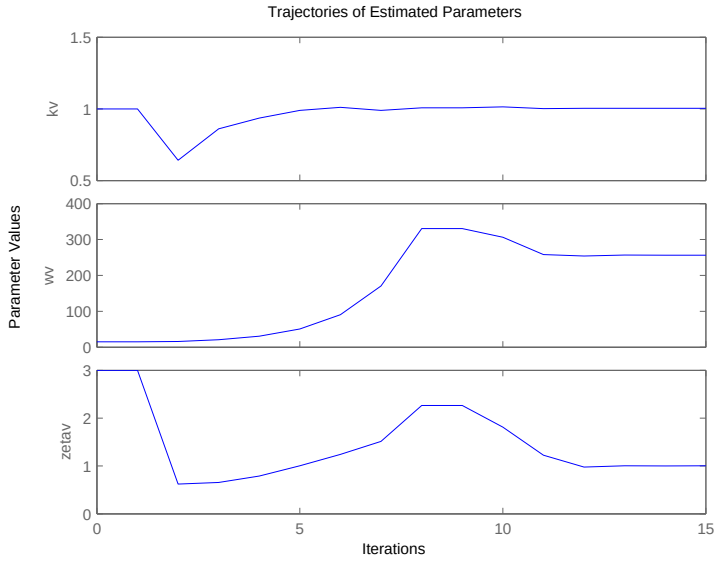
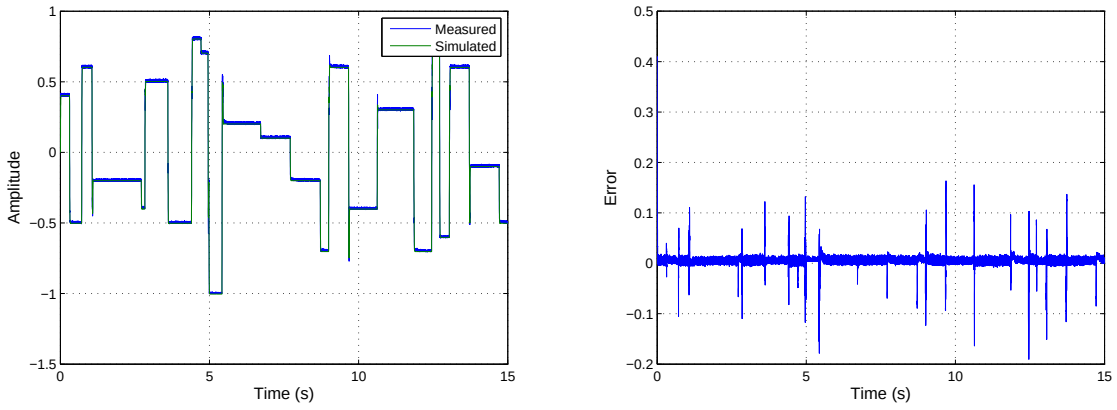


Figure 4.10. Parameter estimation trajectory for actuator directional proportional valve dynamics.

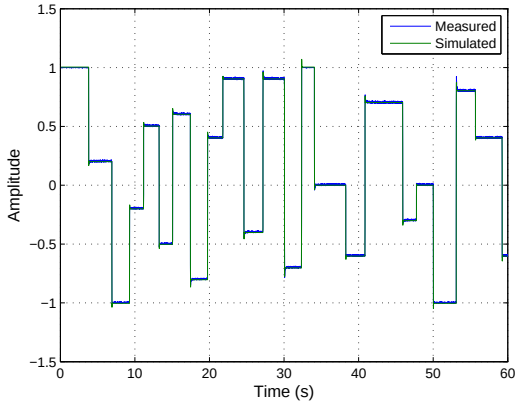


(a) Comparison of experimental and estimated responses

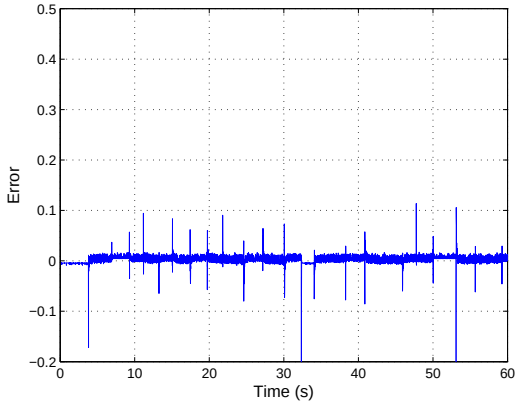
(b) Residuals

Figure 4.11. Validation of the estimated actuator valve model parameters.

Remark 4.1. There residual plots have spikes whenever there is a change in the position of the spool. This is because for small amplitudes, the valves have a very high bandwidth. For example as listed in Table 4.3, the manufacturer lists 100 Hz

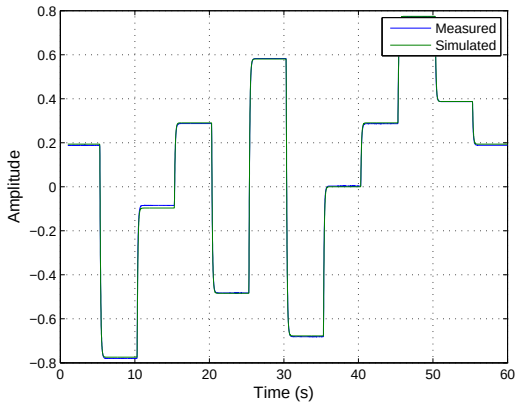


(a) Comparison of experimental and estimated responses

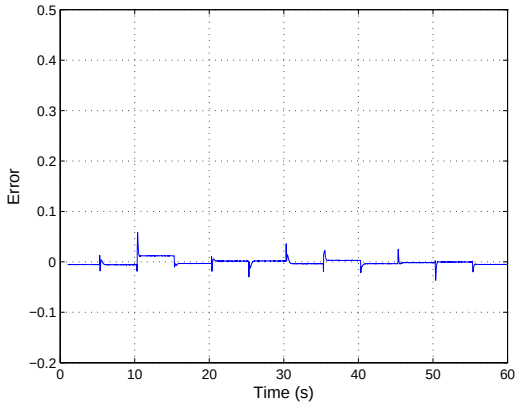


(b) Residuals

Figure 4.12. Validation of the estimated leakage valve model parameters.



(a) Comparison of experimental and estimated responses



(b) Residuals

Figure 4.13. Validation of the estimated leakage valve model parameters.

bandwidth for a 5% amplitude for the actuator directional proportional valve, where as the identified bandwidth for the full range was about 40 Hz. Bandwidth for the full range is expected to be lower than for a 5% amplitude and this lower bandwidth should be utilized in the control development. If the higher bandwidth is utilized in

control development, the controller will expect the valve to respond faster than would be physically possible leading to oscillation in the response.

4.4.6 Hydraulic Parameters

Model of the double rod double acting hydraulic actuator with a servo valve was in equations (3.4), (3.21), and (3.22). Neglecting the internal and external leakages, the equation for the double rod actuator can be rewritten as:

$$\frac{V_t}{4\beta_e}\dot{P}_L = -A\dot{x}_L + Q_L \tag{4.2}$$

$$\Rightarrow \frac{V_t}{4\beta_e}\dot{P}_L = -A\dot{x}_L + \frac{C_d w}{\sqrt{\rho}} x_v \sqrt{P_s - \mathrm{sgn}(x_v)P_L} \tag{4.3}$$

$$\Rightarrow \begin{pmatrix} -\dot{P}_L & x_v\sqrt{P_s - \mathrm{sgn}(x_v)P_L} \end{pmatrix} \begin{pmatrix} \frac{V_t}{4\beta_e} \\ \frac{C_d w}{\sqrt{\rho}} \end{pmatrix} = A\dot{x}_L \tag{4.4}$$

Systems states x_L, P_L, and x_v are available for recording through sensor measurements. These measurements can be used to generate time series for $\dot{P}_L$ and $\dot{x}_L$ using a differentiating filter given by

$$H_d = \frac{s}{\frac{1}{(2\pi f_f)^2}s^2 + \frac{2\zeta_f}{2\pi f_f}s + 1} \tag{4.5}$$

where f_f, the filter frequency, which is chosen as 100 Hz and ζ_f, the damping coefficient, which is chosen as 0.8. Equations (4.2–4.4) can now be used to obtain following the least-squares problem [96]:

$$D\mathbf{x} = b \tag{4.6}$$

$$\begin{pmatrix} -\dot{P}_L(1) & x_v(1)\sqrt{P_s - \mathrm{sgn}(x_v(1))P_L} \\ -\dot{P}_L(2) & x_v(2)\sqrt{P_s - \mathrm{sgn}(x_v(2))P_L} \\ \vdots & \vdots \\ -\dot{P}_L(n) & x_v(n)\sqrt{P_s - \mathrm{sgn}(x_v(n))P_L} \end{pmatrix} \begin{pmatrix} \frac{V_t}{4\beta_e} \\ \frac{C_d w}{\sqrt{\rho}} \end{pmatrix} = \begin{pmatrix} A\dot{x}_L(1) \\ A\dot{x}_L(2) \\ \vdots \\ A\dot{x}_L(n) \end{pmatrix} \tag{4.7}$$

Solution of this least squares problem gave estimates for the hydraulic parameters $\frac{V_t}{4\beta_e}$ and $\frac{C_d w}{\sqrt{\rho}}$. The results of 8 different runs is presented in Figures 4.14 and 4.15. The

slight variation in the estimates between each run is expected due to the stochastic nature of the measurement noise as well presence of unmodeled dynamics. Mean values from these runs were utilized in the control development process.

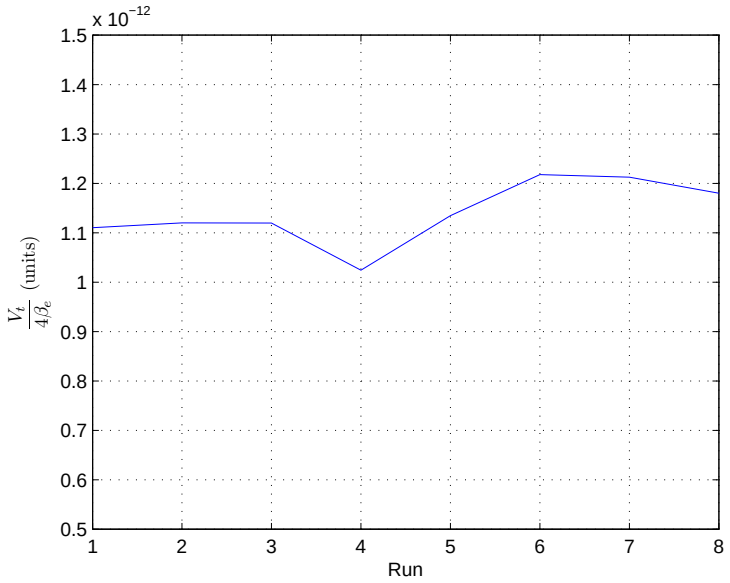


Figure 4.14. Variation of $\frac{V_t}{4\beta_e}$ for eight different runs.

4.4.7 Frictional Force

Hydraulic cylinders have very large frictional forces because of the tight fit required to seal the two chambers of the cylinder. The large magnitude of these forces make identification of friction important. There is a large uncertainty in the estimate of the friction forces which can vary based on direction of motion, velocity of motion, position of the actuator, manufacturing tolerances and defects, etc. It is therefore necessary to identify friction online during the control or fault identification process. However as an exercise to understand the average nature of the frictional force, an estimate is obtained from several constant velocity runs with velocity ranging between .001 m/s to .5 m/s.

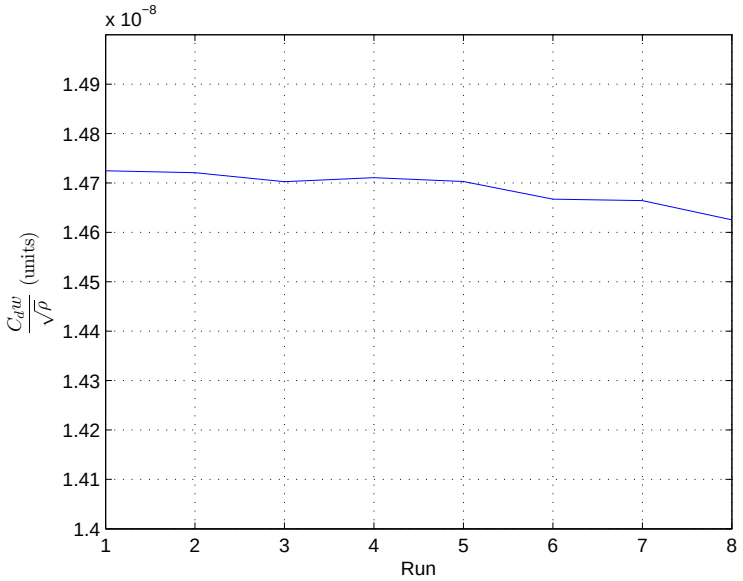


Figure 4.15. Variation of $\frac{C_d w}{\sqrt{\rho}}$ for eight different runs.

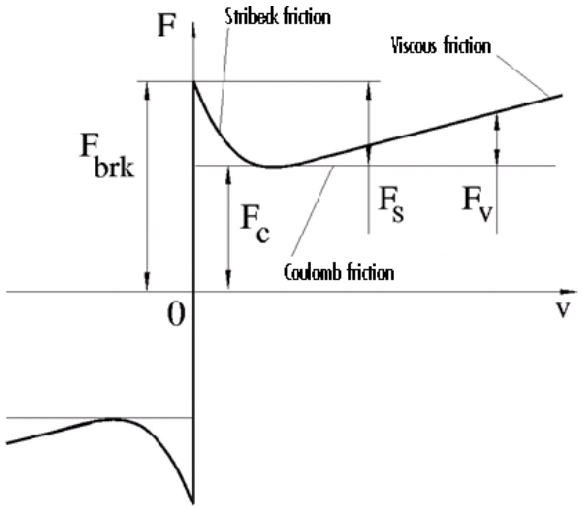


Figure 4.16. Theoretical friction model.

The friction force is typically simulated as a function of relative velocity and is assumed to be the sum of Stribeck, Coulomb, and viscous components, as shown in Figure 4.16. The Stribeck friction, F_S, is the negatively sloped characteristics taking place at low velocities. The Coulomb friction, F_C, results in a constant force at any

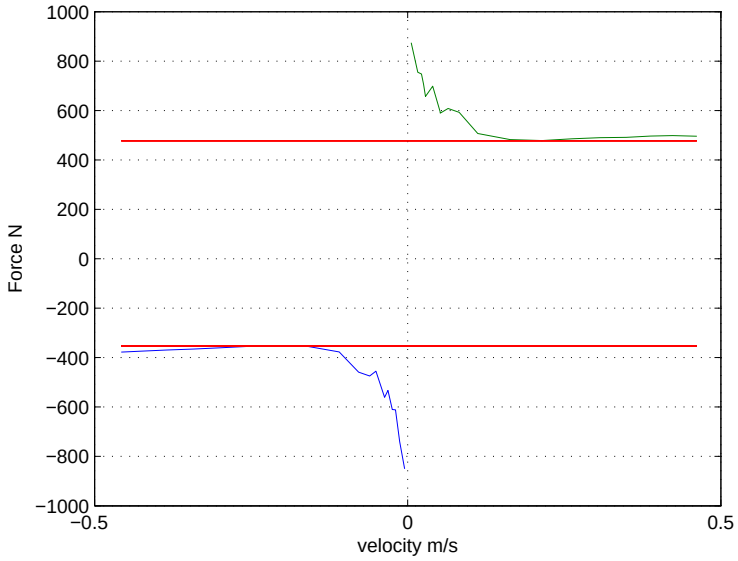


Figure 4.17. Identified actuator friction force.

velocity. The viscous friction, F_V, opposes motion with the force directly proportional to the relative velocity. The sum of the Coulomb and Stribeck frictions at the vicinity of zero velocity is often referred to as the breakaway friction, F_{brk}. The friction is approximated with the following equations:

$$F = \left(F_C + (F_{brk} - F_C)\, e^{-c_v|v|}\right) \operatorname{sgn}(v) + fv \tag{4.8}$$

where v is the relative velocity and c_v is some coefficient. To identify the frictional force between the piston and the cylinder, equation (3.3) is rewritten as follows:

$$b\dot{x}_L - F_{fc}(\dot{x}_L) = AP_L - m\ddot{x}_L \tag{4.9}$$

$$F_{Total} = AP_L - m\ddot{x}_L \tag{4.10}$$

Figure 4.17 shows the results of the total force required to move the piston at a given velocity. As can be seen the shape of the frictional force is similar to that shown in Figure 4.16.

5. CONTROL DEVELOPMENT AND IMPLEMENTATION

In this chapter control development for the longitudinal aircraft dynamics and robust control for the hydraulic actuator will be presented. Mechanical and electrical issues encountered during implementation and their solutions will also be presented. A stochastic filtering based fault identification scheme will be presented leading into development of an adaptive fault tolerant control for the hydraulic actuator.

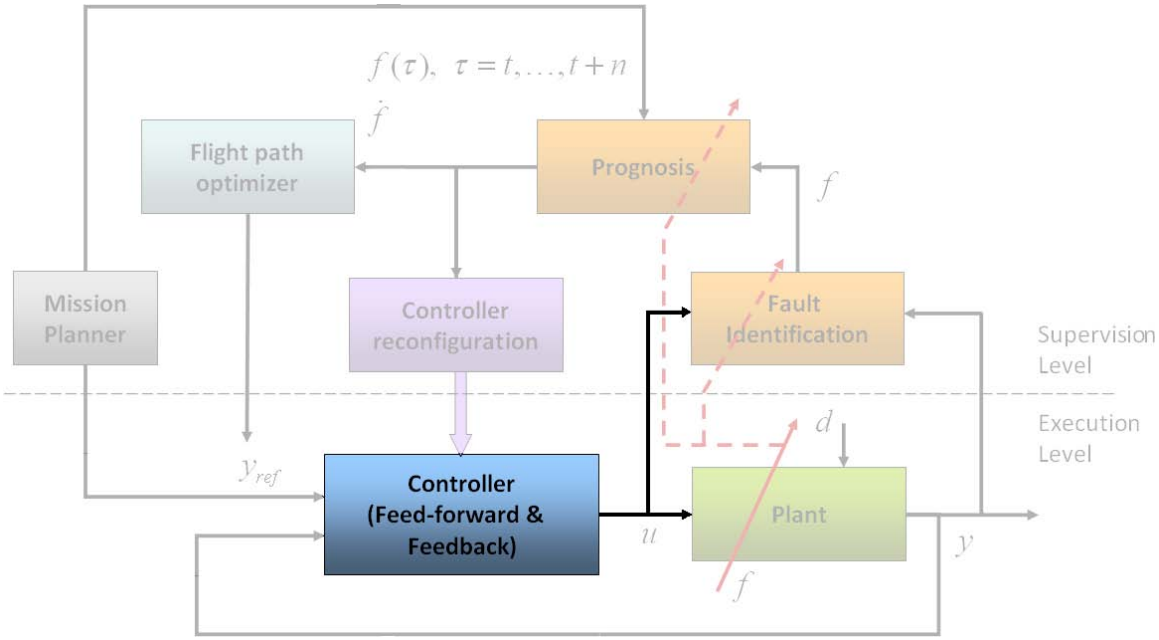


Figure 5.1. Reflexive layer.

5.1 Longitudinal aircraft

The control structure developed for the velocity and the altitude loops of the aircraft to allow way-point navigation in a longitudinal plane is similar to the structure

developed for the full nonlinear model in section 3.2. To begin with, the structure assumes simple first order dynamics for the actuators. The longitudinal dynamics has two main loops: the velocity loop and the altitude loop. The tracking error in velocity and altitude is defined as $e = \begin{pmatrix} e_v & e_h \end{pmatrix}^T$.

The objective is for aircraft states to follow the reference, i.e. to obtain zero steady state error in v_t and h. This can be done by adding integrators in both loops. However, the open loop system already has a few poles near the origin and adding more poles will make the system difficult to control. The altitude loop is more critical than the velocity loop, so an integrator is added only to the altitude loop as shown in Figure 5.2. To pull the closed loop poles into the left half plane, a compensator is added to the velocity loop. This implementation uses poles that are much faster than the system poles. The compensator that is proposed is of the form:

$$\frac{w_v}{e_v} = \frac{K_{vi}}{s + K_{vi}} + K_{vp} = K_{vp}\frac{s + 30 + K_{vi}/K_{vp}}{s + K_{vi}} \tag{5.1}$$

$$u_t = w_v + u_{t_{ff}} \tag{5.2}$$

When the altitude is changed from the nominal condition, the steady state pitch angle also changes from the nominal zero value. This steady state value is supplied by the difference in the reference and actual altitudes. To make the steady state error in altitude go to zero, a PI controller is introduced. A proportional and integrating controller is also used to make sure that no steady-state error in θ, $\Delta\theta$ remains. As long as the error $\theta - \theta_{ref}$ is not equal to zero, the signal from the integrator will increase, which leads to an increasing elevator deflection. The θ-feedback somewhat decreases the damping of the short period mode, which is compensated for by adding a feedback loop of the pitch rate (q) to the elevator. Angle of attach measurements are usually very noisy, hence a wash-out filter is used to before the angle of attach feedback:

$$\frac{w_h^e}{e_h} = \frac{K_{hi}}{s} + K_{hp} = K_{hp}\frac{s + K_{hi}/K_{hp}}{s} \tag{5.3}$$

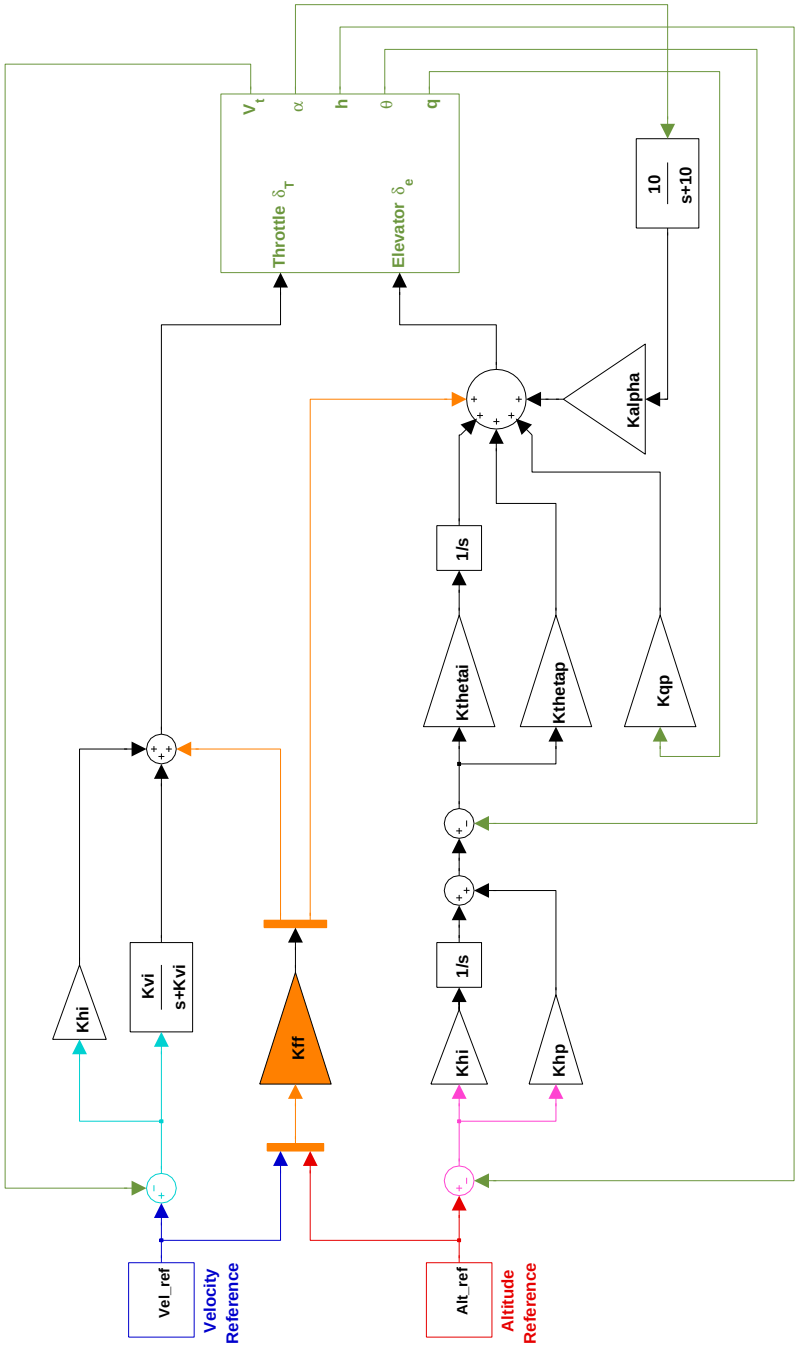


Figure 5.2. Aircraft control structure.

$$\frac{w_\theta^e}{w_h^e - \theta} = \frac{K_{\theta i}}{s} + K_{\theta p} \tag{5.4}$$

$$w_\alpha = K_\alpha \frac{10}{s+10} \tag{5.5}$$

$$u_e = K_{qp} q + w_\theta^e + w_\alpha + u_{e_{ff}} \tag{5.6}$$

where $u_{t_{ff}}$ and $u_{e_{ff}}$ are the feedforward terms and are given by:

$$\begin{pmatrix} u_{t_{ff}} \\ u_{e_{ff}} \end{pmatrix} = K_{ff} \begin{pmatrix} v_{t_{ref}} \\ h_{ref} \end{pmatrix} \tag{5.7}$$

K_{ff} is a 2×2 matrix to allow cross-coupling between the velocity and altitude inputs. Equations (5.1)–(5.7) can be written in state space form and augmented to the system model given in equation (3.24). This augmented system is then used to formulate a linear quadratic (LQ) design problem. The Simulink Response Optimization toolbox

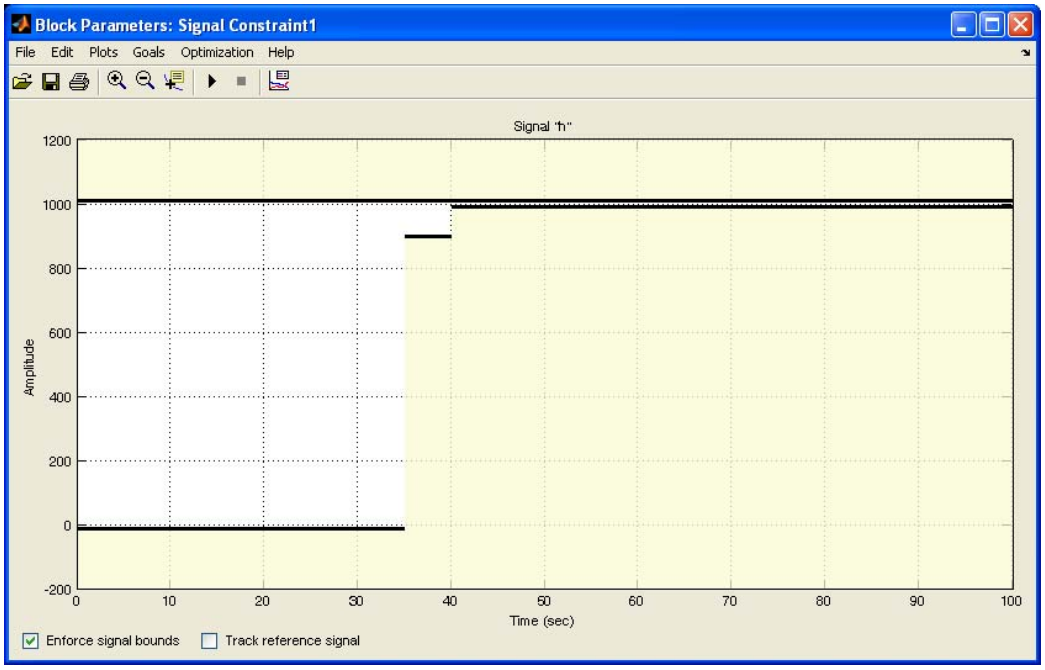


Figure 5.3. Closed loop response shaping.

is then used to optimize the closed loop step response of the full model. The toolbox allows for explicit specifications on the output performance and since the saturation and rate limiting constraints are part of the model, there is no need to specify these

constraints separately. The constraint bound data and tuned parameter information are converted into a constrained optimization problem, which is solved using standard gradient based methods. Figure 5.3 shows a sample constraint being applied on the altitude output of the model.

5.2 Hydraulic Actuator

The nonlinear state space equations of motion of a double rod hydraulic actuator derived in section 3.6 are repeated here:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \theta_1 \left(x_3 - \bar{b} x_2 - \bar{F}_{fc} \right) + \theta_2 + \tilde{d} \\
\dot{x}_3 &= \theta_3 \left(-\bar{A} x_2 - \bar{C}_{tm} \sqrt{|x_3|}\, \mathrm{sgn}(x_3) + g_3 x_4 \right) \\
\dot{x}_4 &= x_5 \\
\dot{x}_5 &= 2\zeta_v \omega_v x_5 - \omega_v^2 x_4 + \bar{K}_v \omega_v^2 u
\end{aligned} \tag{5.8}$$

As detailed in Chapter 4, the experimental setup has two parts: (i) the actuator loop, and (ii) the load loop. Details of the robust position control and force control for the actuator and the load loop respectively is presented in this section.

5.2.1 Robust Position Control

Because the hydraulic actuator operates on the principle of pressure difference in the two chambers of the cylinder, it is natural to control the position by controlling the pressure difference. A desired reference position trajectory is used to generate a reference force trajectory, and a robust control strategy based on feedback linearizing control of the pressure dynamics, equation (3.4), is then used to track the desired force trajectory asymptotically. The spool dynamics are almost 10 times faster than the hydraulic actuator dynamics and, hence, the spool dynamics can be neglected and replaced simply by $x_v = \bar{K}_v u$ for position control purposes. By neglecting the internal leakage dynamics, the force dynamics equation can be derived as follows:

$$F = P_L A \tag{5.9}$$

$$\Rightarrow \dot{F} = \dot{P}_L A \tag{5.10}$$

$$= \frac{4\beta_e}{V_t} A \left(-A\dot{x}_L + Q_L\right) \tag{5.11}$$

$$= \frac{4\beta_e}{V_t} A \left(-A\dot{x}_L + C_d w x_v \sqrt{\frac{P_s - \text{sgn}(x_v) P_L}{\rho}}\right) \tag{5.12}$$

Define

$$\varsigma = \frac{4\beta_e}{V_t} A C_d w \sqrt{\frac{P_s - \text{sgn}(x_v) P_L}{\rho}} \tag{5.13}$$

and let F_d be the desired force trajectory to be achieved. The following control input is then considered:

$$x_v = \frac{1}{\varsigma}\left(\frac{4\beta_e A^2 \dot{x}_L}{V_t} + \dot{F}_d - k_f(F - F_d)\right) \tag{5.14}$$

A Lyapunov function is defined as

$$V = \frac{1}{2} k_l (F - F_d)^2 \tag{5.15}$$

and then differentiated,

$$\dot{V} = k_l(\dot{F} - \dot{F}_d)(F - F_d) \tag{5.16}$$

$$= k_l \left(\frac{4\beta_e A^2 \dot{x}_L}{V_t} + \varsigma x_v - dot F_d\right)(F - F_d) \tag{5.17}$$

$$= -k_l k_f \left(F - F_d\right)^2 \tag{5.18}$$

producing a negative semi-definite function. Using Barbalatt's Lemma, it can be shown that the error $F - F_d$ goes to zero exponentially [123]. The desired force trajectory is chosen as follows:

$$F_d = m\ddot{x}_d - k_v\left(\dot{x}_L - \dot{x}_d\right) - k_p\left(x_L - x_d\right) + \hat{f} \tag{5.19}$$

where x_d is the desired position trajectory and $\hat{f}$ is the estimate of frictional forces on the cylinder. Using this trajectory, it can be shown that the position error is upper bounded by $e \leq \left(\frac{1}{2}\right)|\delta|$ at steady state, where $\delta = \hat{f} - \tilde{f}$.

5.2.2 Model Validation

To validate the developed model, the designed control strategy was implemented on the simulation model and the experimental setup. The controller was implemented without any external load. The control parameters that were used are given in Table 5.1. Figures 5.4–5.6 show the comparison of the experimental and simulated results for a randomly generated input signal. Figure 5.4 shows the reference trajec-

Table 5.1 Comparison of simulation and experimental control parameters.

	Simulation	Experiments
k_v	1×10^2	1×10^2
k_p	1×10^6	1×10^6
k_f	80	30

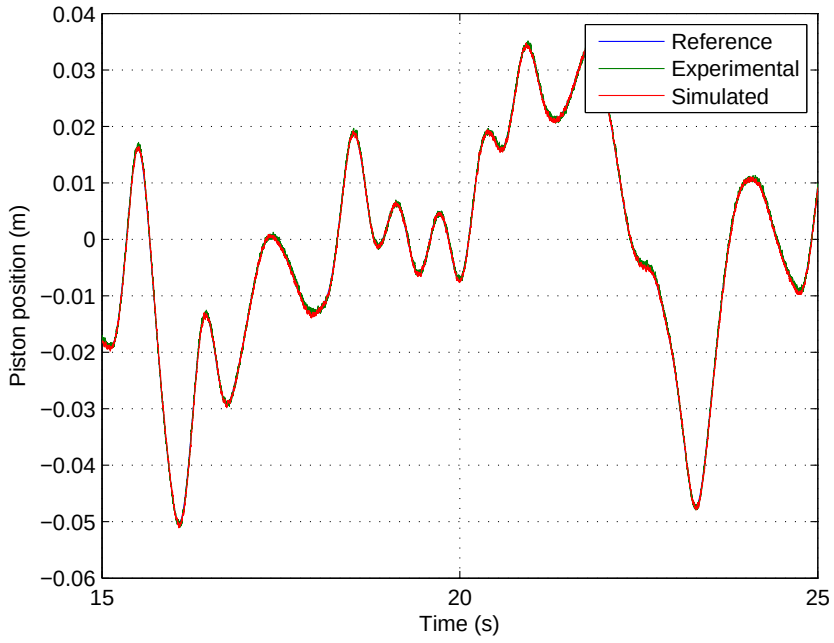


Figure 5.4. Trajectory following for simulation model and experimental setup.

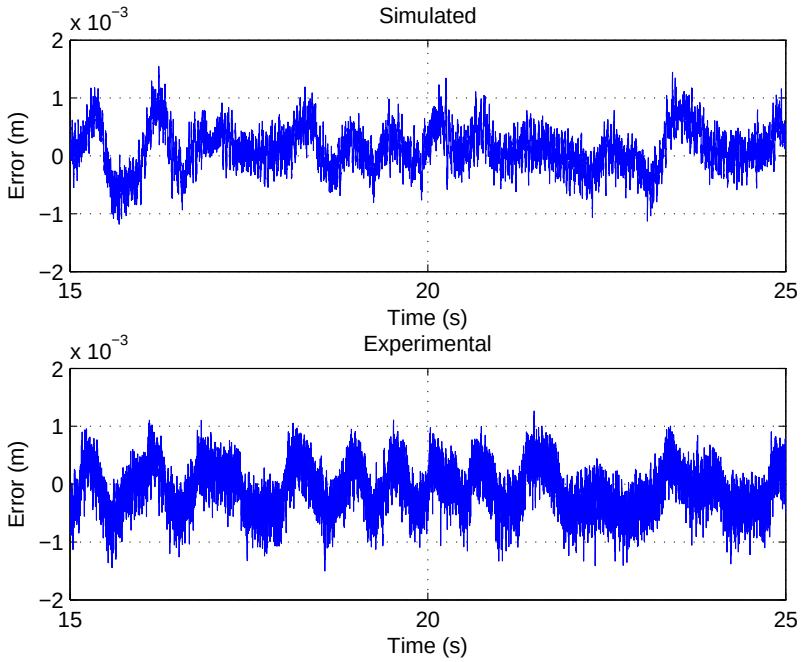


Figure 5.5. Trajectory following errors during simulation and experiments.

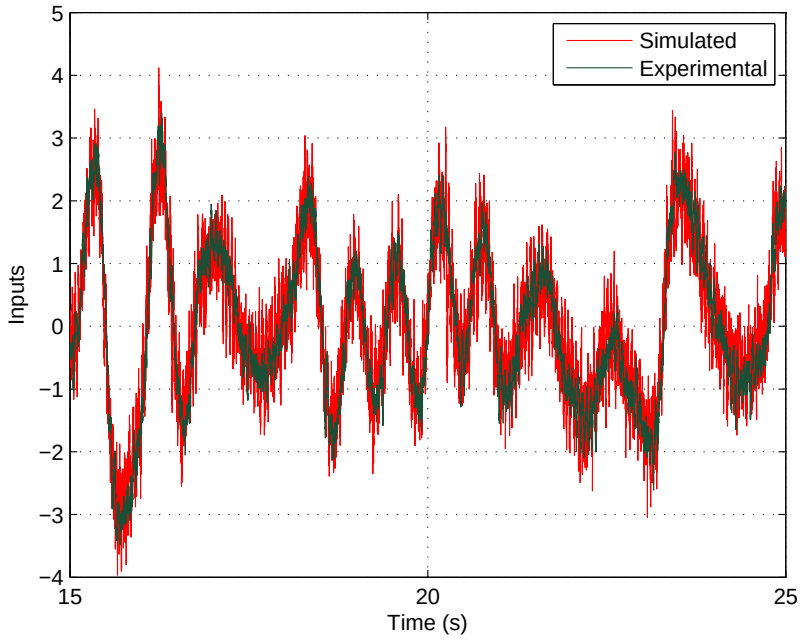


Figure 5.6. Input voltage comparison for simulation and experiments.

tory as well as simulation and experimental results for trajectory following. Figure 5.5 shows the trajectory following errors for simulated and experimental hydraulic actuators. As is evident from the figure, both simulation and experimental results are very similar. As expected, the experimental results are more noisy. The input comparisons shown in Figure 5.6 also validates the actuator model and verifies the controller developed.

Remark 5.1. The k_f parameter used in the experiments is smaller because that parameter defines the time constant of the error signal. Larger k_f results in a smaller time constant requiring larger system bandwidth for chatter free operation. This choice of k_f leads to chatter free operation in the experimental response characteristics. Simulation model allows for larger bandwidth than the physical system and hence the larger k_f. This is also evident from the input comparison shown in Figure 5.6, where the simulation input has larger "noise".

5.2.3 Robust Force Control

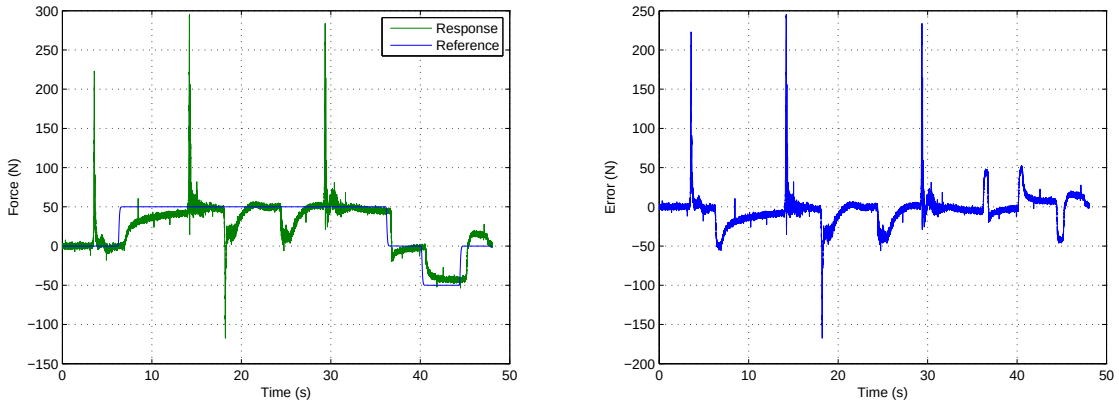
For force control, a simple PI controller with antiwindup is implemented as:

$$x_{v_l} = x_v - \frac{1}{\varsigma}\left(-K_p(F - F_c) - K_i \,\mathrm{sat}\left(\int_{\tau=0}^{\tau=t}(F - F_c)d\tau\right)\right) \tag{5.20}$$

where x_v is the input calculated for the actuator loop, equation (5.14). This term acts as a feed-forward and improves the transient tracking performance. The second term in the bracket ensure fast response and zero steady state error.

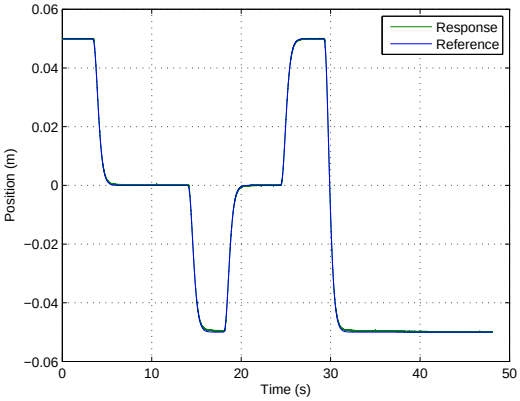
5.2.4 Mechanical Implementation Issues

The results for simultaneous force and position tracking are shown in Figure 5.7. Although the position tracking performance during simultaneous force and position tracking is very good, spikes are observed in the force trajectory following (Figure 5.7(a)). This appears to happen whenever there is a change in the position (Figure 5.7(c)). The main reasons for these spikes are:

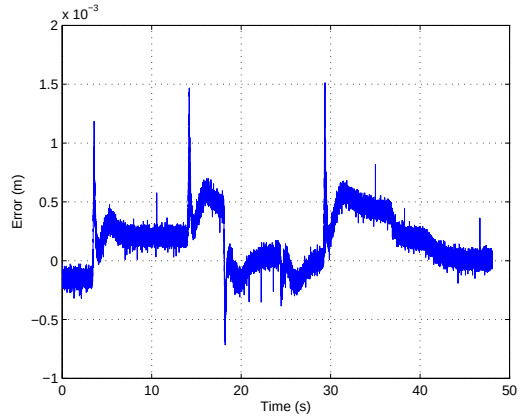


(a) Force tracking

(b) Errors in force tracking



(c) Position tracking



(d) Errors in position tracking

Figure 5.7. Simultaneous force and position tracking in load and actuator loops respectively, when they are connected together with a load cell.

1. As was described in Section 4.2.3, the proportional valve for the force control loop has a much lower bandwidth (20 Hz for 5% amplitude) as compared to the directional proportional valve in the actuator loop (100 Hz for 5% amplitude). This results in slower build up of pressure in the load cylinder chamber resulting in a slight mismatch in the accelerations of the two cylinders, which causes the spikes.

2. The force control proportional valve also has a 20% dead-zone nonlinearity. Although the system bandwidth is still much smaller than the valve bandwidths, this nonlinearity causes the pressure in the position control loop to build up a little faster than that in the force control loop, resulting in the spike. The spike however quickly decreases to zero as the valve goes out of the dead-zone.

3. During initial design, a pressure reducing valve was introduced (Section 4.2.3, Label 11) between the actuator and load parts of the experimental setup to ensure that the actuator can always overcome the load. However this leads to a lower supply pressure in the load part of the setup. Lower supply pressure also leads to lower acceleration during position control leading to the spikes.

This problem was remedied through following steps:

1. The load directional proportional valve was replaced with one matching the dynamic response characteristics of the actuator directional proportional valve

2. The pressure reducing valve was removed to reduce the pressure reduction and consequently increase the supply pressure available to the load circuit.

3. The hoses connecting the actuator part of the setup to the load part were replaced with larger diameter ones to further reduce the frictional losses.

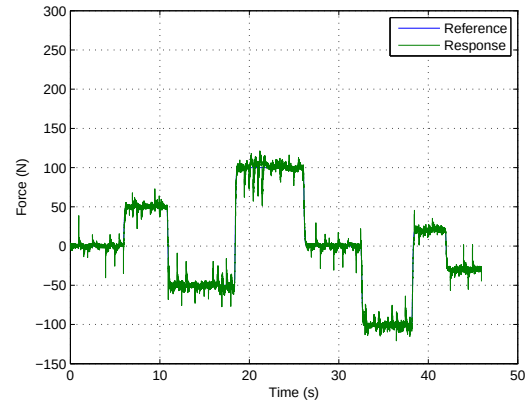
Figure 5.8 shows the comparison of pictures before and after the redesign was completed. Of the three actions performed to improve the force tracking performance, the second one involving removal of the pressure reducing valve was the most effective. The results of simultaneous force and position tracking post mechanical modifications are shown in Figure 5.9. As can be inferred from Figure 5.9(b), the force tracking performance post hardware modifications is much better. The error spikes after the modification are less than 40 N in magnitude for a much faster position tracking trajectory compared to 300 N before the modification. These spikes are attributed to slight mismatch in dynamic characteristics of the directional proportional valves

(a) Picture of initial experimental setup design

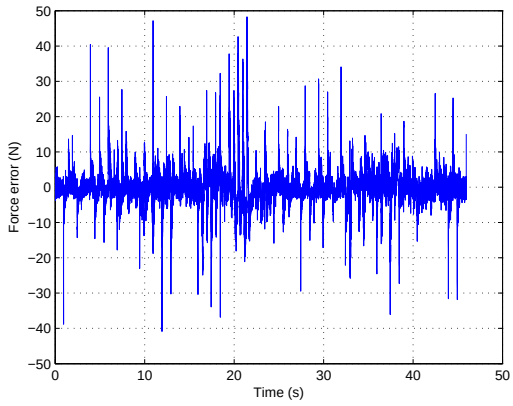
(b) Picture of the experimental setup after modifications

Figure 5.8. Experimental redesign to improve simultaneous force and position tracking performance.

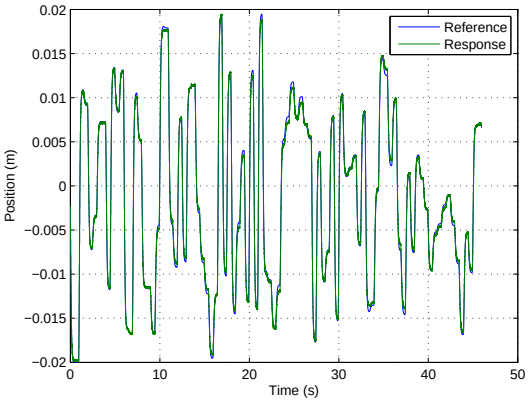
controlling the actuator circuit and the load circuit. These difference are due to manufacturing process and cannot be remedied easily. The spikes also decay in magnitude almost instantaneously compared to those before the modifications. The modifica-



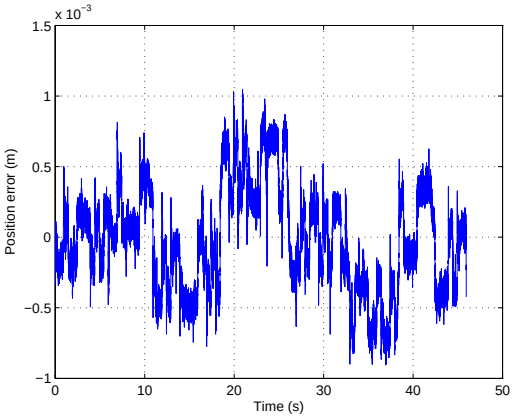
(a) Force tracking



(b) Errors in force tracking



(c) Position tracking



(d) Errors in position tracking

Figure 5.9. Simultaneous force and position tracking after hardware modifications.

tion also allow application of forces which are higher in magnitude compared to those before modifications.

5.3 Fault Identification

The leakage fault under consideration is the internal leakage fault. The fault is modeled as a parametric fault and a nonlinear filtering technique is used to estimate the states and parameters. The main advantage of this method is its ability to identify incipient damage and provide a numerical estimate of the actual leakage

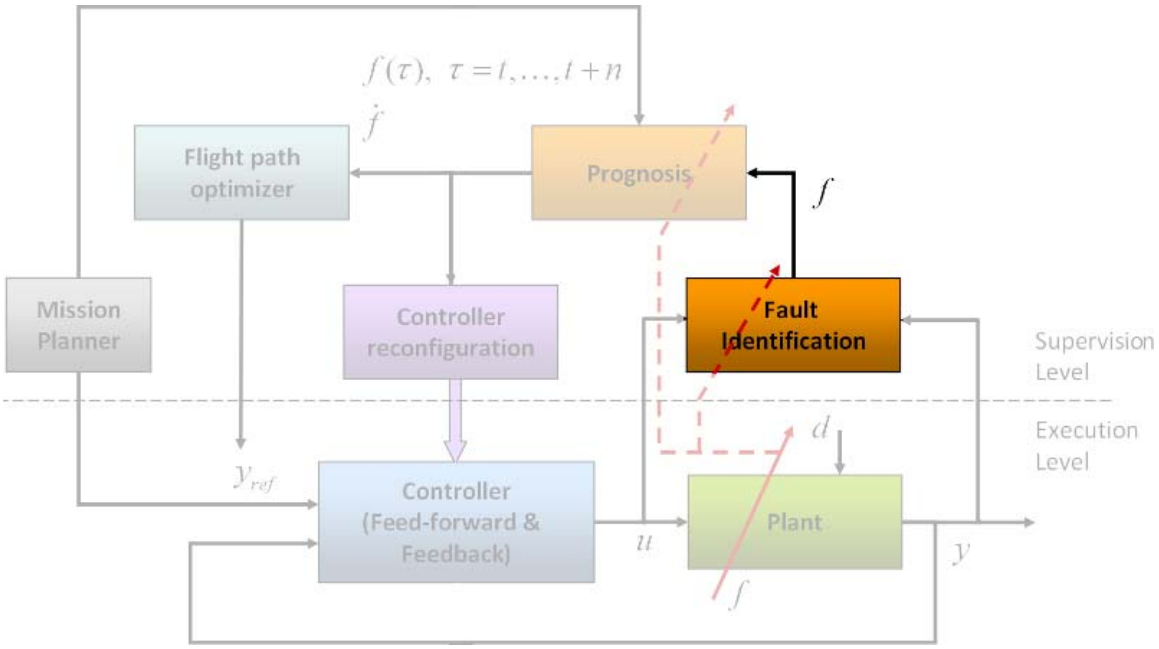


Figure 5.10. Location and interaction of fault identification module in the approach.

rate. This section gives a brief introduction to the Divided Difference Filter followed by experimental results for fault identification.

5.3.1 Divided Difference Filtering

Although the Kalman Filter (KF) has been used to solve the problem of optimal state filtering in linear models with Gaussian noise, there is no such solution available for non-linear systems. Instead, a number of estimators may be found in the literature that are basically extensions of the KF based on approximations of the non-linear system equations. The most prominent and widely used method among these is the Extended Kalman Filter (EKF) [124–126], which is obtained by first order linearization of the system equations so that the traditional KF can be applied at each step. Extensive use of the EKF over the past few decades has shown that, in several important practical problems, the first order approximation provides in-

sufficient accuracy, resulting in significant bias or even convergence problems [127]. Moreover, the derivation of the Jacobian matrix, which is necessary for the first order approximation, is non-trivial in many applications.

Recently, efficient derivative free filtering techniques, which do not require evaluation of the Jacobian matrix, have been proposed. These filters have a computational complexity similar to the EKF but are much simpler to implement. Schei [128] uses a first order approximation of a non-linear transformation that implements divided differences for the propagation of the mean and covariance of the state distribution. Quine [129] uses a minimal ensemble set of state vectors to propagate the first two moments of the state distribution to obtain a filter that is shown to be equivalent to the EKF in a limiting case. More accurate filters, which are called Sigma Point Kalman Filters (SPKFs) [130], have also been proposed that use second or higher order approximations of the non-linear transformation. This class of filters include the Unscented Kalman Filter (UKF), which is based on the unscented transform, a technique for propagating the mean and covariance of a distribution through a non-linear transformation [131], and the divided difference filter (DDF), which is based on a second order approximation using Stirling's interpolation formula [132]. Table 5.2 give a comparison of different filtering techniques that were applied for the fault identification purpose. The EKF and the Type 1 DDF were not able to identify the fault while the performance of the UKF and Type 2 DDF was similar. However, implementation of DDF was easier and hence was chosen as a nonlinear filtering technique for fault identification in this application.

This section gives a brief summary of the DDF. The reader is referred to [132] for a detailed description and analysis of the DDF. The class of systems for which this procedure is applicable is given below:

$$x(k+1) = f_m\left(x(k), u(k)\right) + v(k) \tag{5.21}$$

$$y(k) = Hx(k) + w(k) \tag{5.22}$$

Table 5.2 Comparison of different filtering algorithms.

	EKF	DDF Type 1	UKF	DDF Type 2
Approximation Order	1^{st}	1^{st}	2^{nd}	2^{nd}
Jacobian Computation Required	Yes	No	No	No
Bias/Variance Errors	Yes (large)	Yes (large)	Small	Smaller than UKF
Computational Load	Low	Lower	Slightly more than EKF	Similar to UKF
Internal Leakage Identified	No	No	Yes	**Yes**

where $x(k) \in \mathbb{R}^n$ is the state vector, $y(k) \in \mathbb{R}^p$ is the output vector, $u(k) \in \mathbb{R}^m$ is the input vector and $f_m(x(k), u(k))$ is the non-linear plant model. It is assumed that f_m is either a globally Lipschitz continuous function or that it is locally Lipschitz continuous with $x(k)$ restricted to a compact domain $\mathcal{D} \in \mathbb{R}^n$. $v(k)$ and $w(k)$ are assumed to be independent, identically distributed Gaussian random variables with $v(k) \sim \mathcal{N}(0, Q(k))$ and $w(k) \sim \mathcal{N}(0, R(k))$ called the process and measurement noise, respectively.

The DDF filter provides an estimate of the evolution of a Gaussian distribution through a non-linear transformation by making use of Stirling's formula to obtain a second order approximation of the non-linear system around the current state estimate. Stirling's formula may be derived from the multi-dimensional Taylor's series expansion by replacing the derivatives with divided differences. The equations for computing the second order approximation along with those for propagating a multi-dimensional Gaussian random variable x with mean $\bar{x}$ through the approximated

function, $f_s(x, \bar{x})$, using special vectors, which are columns of the matrix square root of the covariance of x, are given in the literature [132]. Based on this formulation, the DDF filter equations for the system in (5.21) can be summarized as in (5.27)–(5.33).

The filter equations have been suitably simplified based on the model structure with a linear output equation and the assumption of additive Gaussian noise. In the following equations, all the quantities with a "bar" denote *a priori* estimates (before the actual output is observed) while the quantities with a "hat" denote *a posteriori* estimates (after the output is observed). Further, let $\bar{S}_x(k)$, $\hat{S}_x(k)$, $S_v(k)$, and $S_w(k)$ denote the matrix square roots of $\bar{P}(k)$ (the *a priori* state covariance estimate), $\hat{P}(k)$ (the *a posteriori* state covariance estimate), $Q(k)$ (the process noise covariance) and $R(k)$ (the output noise covariance), respectively. These matrices are calculated using the following equations:

$$\bar{P}(k) = \bar{S}_x(k)\bar{S}_x(k)^T \tag{5.23}$$

$$\hat{P}(k) = \hat{S}_x(k)\hat{S}_x(k)^T \tag{5.24}$$

$$Q(k) = S_v(k)S_v(k)^T \text{ and} \tag{5.25}$$

$$R(k) = S_w(k)S_w(k)^T \tag{5.26}$$

Let $\bar{x}(k)$ denote the *a priori* state estimate and let $\hat{s}_{x,p}$ denote the p^{th} column of $\hat{S}_x$. Then, the a priori state estimate calculated by the DDF is given by (5.27),

$$\begin{aligned}\bar{x}(k+1) =& \frac{h^2-n}{h^2} f_m\left(\hat{x}(k), u(k)\right) \\ &+ \frac{1}{2h^2}\sum_{p=1}^{n}\Big\{ f_m\left(\hat{x}(k) + h\hat{s}_{x,p}, u(k)\right) \\ &\qquad + f_m\left(\hat{x}(k) - h\hat{s}_{x,p}, u(k)\right)\Big\}\end{aligned} \tag{5.27}$$

Two temporary matrices are defined, $S^{(1)}_{x\hat{x}}(k)$ and $S^{(2)}_{x\hat{x}}(k)$ whose p^{th} columns are computed as

$$\begin{aligned} S^{(1)}_{x\hat{x}}(k)_{(p)} =& \frac{1}{2h}\Big\{ f_m\left(\hat{x}(k) + h\hat{s}_{x,p}, u(k)\right) \\ & - f_m\left(\hat{x}(k) - h\hat{s}_{x,p}, u(k)\right) \Big\} \end{aligned} \tag{5.28}$$

$$\begin{aligned} S^{(2)}_{x\hat{x}}(k)_{(p)} =& \frac{\sqrt{h^2-1}}{2h^2}\Big\{ f_m\left(\hat{x}(k) + h\hat{s}_{x,p}, u(k)\right) \\ & + f_m\left(\hat{x}(k) - h\hat{s}_{x,p}, u(k)\right) \\ & - 2f_m\left(\hat{x}(k), u(k)\right) \Big\} \end{aligned} \tag{5.29}$$

Then the updated square root of the *a priori* state covariance matrix, $\bar{S}_x(k+1)$, is given by

$$\bar{S}_x(k+1) = HT\left(\begin{bmatrix} S^{(1)}_{x\hat{x}}(k) & S^{(2)}_{x\hat{x}}(k) & S_v(k) \end{bmatrix}\right) \tag{5.30}$$

where $HT(S)$ denotes a Householder transformation [133] to convert the matrix S into a square triangular form such that $HT(S)HT(S)^T = SS^T$. The gain matrix, $K(k+1)$, is calculated using (5.31), and the square root of the *a posteriori* state covariance matrix, $\hat{S}_x(k+1)$, is given by (5.32).

$$\begin{aligned} K(k+1) =& \bar{S}_x(k+1)\bar{S}_x(k+1)^T H^T \\ & \times \left(H\bar{S}_x(k+1)\bar{S}_x(k+1)^T H^T + R(k+1)\right)^{-1} \end{aligned} \tag{5.31}$$

$$\hat{S}_x(k+1) = HT\left(\begin{bmatrix} \left(I - K(k+1)H\right)\bar{S}(k+1) & S_w(k+1) \end{bmatrix}\right) \tag{5.32}$$

The innovation (output prediction error) is defined as $\gamma(k+1) = y(k+1) - H\bar{x}(k+1)$ and $\bar{x}(k+1)$ is used to denote the *a posteriori* estimate after the k^{th} observation. The state estimate is then updated using (5.33) to get the final *a posteriori* estimate

$$\hat{x}(k+1) = \bar{x}(k+1) - K(k+1)\gamma(k+1) \tag{5.33}$$

5.3.2 Fault Modeling

The internal leakage fault in a hydraulic system was modeled as a parametric fault based on equations of the flow through an orifice in Chapter 3. The product of flow

coefficient and the unknown orifice area is treated as the unknown parameter and is appended to the state space representation of the system. The DDF nonlinear filter is then used to estimate this state and, hence, the fault parameter. The internal leakage is given as:

$$Q_{leak} = C_{tm}\sqrt{|P_L|}\,\mathrm{sgn}\,P_L \tag{5.34}$$

where

$$C_{tm} \propto x_{v_{leak}} \tag{5.35}$$

$$\text{or } C_{tm} = k_q x_{v_{leak}} \tag{5.36}$$

This fault parameter is appended to the system (3.11) and converted to discrete form using simple Euler difference as follows:

$$x_1(k+1) = x_1(k) + T_s x_2(k) \tag{5.37}$$

$$\begin{aligned} x_2(k+1) = x_2(k) + T_s\theta_1\left(x_3(k) - \bar{b}x_2(k) - \bar{F}_{fc}\right) \\ + \theta_2 + x_6(k) \end{aligned} \tag{5.38}$$

$$\begin{aligned} x_3(k+1) = x_3(k) + T_s\theta_3\Big(- \bar{A}x_2(k) \\ - x_5(k)\sqrt{|x_3(k)|}\,\mathrm{sgn}(x_3(k)) \\ + \sqrt{\bar{P}_s - \mathrm{sgn}(x_4(k))x_3(k)}x_4(k)\Big) \end{aligned} \tag{5.39}$$

$$x_4(k+1) = x_4(k) + T_s\left(-\frac{1}{\tau_v}x_4(k) + \frac{\bar{K}_v}{\tau_v}u\right) \tag{5.40}$$

$$x_5(k+1) = x_5(k) \tag{5.41}$$

$$x_6(k+1) = x_6(k) \tag{5.42}$$

where $x_1, \ldots, x_4$ are states (position, velocity, load pressure and spool position), x_5 is the parameter appended as a state for the unknown internal leakage fault and x_6 is the appended state for estimating friction. It is important to estimate friction

online because of the large uncertainty associated with offline friction estimates. The addition of the friction parameter improves the estimation accuracy significantly.

To use the DDF, the system model, f_m, can be easily inferred from (5.37)-(5.42). $y(k)$ is the measured output of the system. In this case, the first four states are being measured. Hence,

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{5.43}$$

$v(k) \sim \mathcal{N}(0, Q(k))$ is the Gaussian process noise, with $Q(k)$ being the process covariance. $w(k) \sim \mathcal{N}(0, R(k))$ represents the Gaussian measurement with R representing the noise covariance. The matrices used in the filter implementation are:

$$Q = \text{diag}\left(\begin{bmatrix} 1 \times 10^{-2}, 1 \times 10^{-2}, 1, 1 \times 10^{-2}, 6 \times 10^{-6}, 1 \end{bmatrix}\right) \tag{5.44}$$

$$R = \text{diag}\left(\begin{bmatrix} 1 \times 10^{-8} & 1 \times 10^{-2} & 1 \times 10^{7} & 1 \times 10^{-2} \end{bmatrix}\right) \tag{5.45}$$

$$P = \text{diag}\left(1 \times 10^{2} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}\right) \tag{5.46}$$

5.3.3 Stability of the DDF

Xiong et. al. have shown the stability of another second order filter viz. the Unscented Kalman Filter. Here we follow a process similar to one given in [134]. There is a minor difference in the first few steps as given below. Define the estimation and prediction errors by

$$\tilde{\hat{x}}(k) = x(k) - \hat{x}(k) \tag{5.47}$$

$$\tilde{\bar{x}}(k) = x(k) - \bar{x}(k) \tag{5.48}$$

Expanding $x(k)$ by Taylor series about $\hat{x}(k-1)$ gives

$$\begin{aligned} x(k) =& f_m(\hat{x}(k-1)) + \nabla f_m(\hat{x}(k-1))\tilde{\hat{x}}(k-1) \\ &+ \frac{1}{2}\nabla^2 f_m(\hat{x}(k-1))\tilde{\hat{x}}(k-1)^2 + \cdots + v(k) \end{aligned} \tag{5.49}$$

where

$$\nabla^i f_m(\hat{x})\tilde{\hat{x}}^i = \left(\sum_{j=1}^{L} \tilde{\hat{x}}_j \frac{\partial}{\partial x_j}\right)^i f_m(x)\Big|_{x=\hat{x}(k-1)}$$

x_j denotes the jth component of x. Expanding $\bar{x}$ given in (5.27) using the Taylor series gives:

$$\begin{aligned}\bar{x}(k) = &\frac{h^2-n}{h^2} f_m(\hat{x}(k-1), u(k-1)) \\ &+ \frac{1}{2h^2}\sum_{p=1}^{n} f_m(\hat{x}(k-1) + h\hat{s}_{x,p}, u(k-1)) \\ &+ \frac{1}{2h^2}\sum_{p=1}^{n} f_m(\hat{x}(k-1) - h\hat{s}_{x,p}, u(k-1)) \\ = &f_m(\hat{x}(k-1)) + \frac{1}{2}\nabla^2 f_m(\hat{x}(k-1))\hat{P}(k-1) + \text{h.o.t.s} \end{aligned} \tag{5.50}$$

Substituting (5.50) and (5.49) into (5.48) gives

$$\tilde{\hat{x}}(k) \approx F(k)\tilde{\hat{x}}(k-1) + v(k) \tag{5.51}$$

where

$$F(k) = \left(\frac{\partial f_m(x)}{\partial x}\Big|_{x=\hat{x}(k-1)}\right)$$

Since only the linear term is used in (5.51) for approximating the *posteriori* error, to obtain an exact equality, the term needs to be multiplied by an unknown term. A diagonal matrix with suitable dimension is used to satisfy the equality, $\beta(k) = \text{diag}\,(\beta_1(k), \beta_2(k), \ldots, \beta_n(k))$ so that

$$\tilde{\hat{x}}(k) \approx \beta(k)F(k)\tilde{\hat{x}}(k-1) + v(k) \tag{5.52}$$

After this the process followed is exactly similar to the one given in [134] since it only requires the measurement update equations. The measurement update is linear and the equations are the same in both cases except for a minor algorithmic difference in the calculation of the gain matrix. For brevity, the main Theorem derived in [134] is reproduced here:

Theorem 5.1 ([134]). *Consider a nonlinear stochastic system given by (5.37)–(5.42) and the DDF algorithm as stated by (5.27)–(5.33). Let the following assumptions hold:*

- *There are real numbers $f_{min}, h_{min}, \beta_{min} \neq 0$, $f_{max}, h_{max}, \beta_{max} \neq 0$ such that the following bounds on various matrices are fulfilled for every $k \neq 0$:*

$$f_{min}^2 I \leq F(k)F(k)^T \leq f_{max}^2 I \tag{5.53}$$

$$h_{min}^2 I \leq H(k)H(k)^T \leq h_{max}^2 I \tag{5.54}$$

$$\beta_{min}^2 I \leq \beta(k)\beta(k)^T \leq \beta_{max}^2 I \tag{5.55}$$

- *There are real numbers $q_{max}, \hat{q}_{min}, \hat{q}_{max}, r_{min}, p_{max}, p_{min} > 0$ such that following bounds are fulfilled:*

$$Q(k) \leq q_{max} I \tag{5.56}$$

$$\hat{q}_{min} \leq \hat{Q}(k) \leq \hat{q}_{max} \tag{5.57}$$

$$r_{min} \leq R(k) \tag{5.58}$$

$$p_{min} \leq \hat{P}(k) \leq p_{max} \tag{5.59}$$

Then the following will hold:

A) For a stochastic positive definite Lyapunov type function

$$V_k(\tilde{\hat{x}}(k)) = \tilde{\hat{x}}(k)^T \hat{P}(k)^{-1} \tilde{\hat{x}}(k), \tag{5.60}$$

its discrete time derivative satisfies

$$E\left[V_k(\tilde{\hat{x}}(k))\right] - V_{k-1}(\tilde{\hat{x}}(k-1)) \leq \mu_{max} - \lambda_{min} V_{k-1}(\tilde{\hat{x}}(k-1)) \tag{5.61}$$

B) The estimation error $\tilde{x}_k$ is bounded in the mean square sense as follows

$$\begin{aligned} E\left[||\tilde{\hat{x}}(k)||^2\right] \leq & \frac{p_{max}}{p_{min}} E\left[||\tilde{\hat{x}}(0)||^2\right](1-\lambda_{min})^k \\ & + \frac{p_{max}}{p_{min}} \sum_{i=1}^{k-1} (1-\lambda_{min})^i \end{aligned} \tag{5.62}$$

where $E[\bullet]$ represents the expected value. The expressions for the terms μ_{max}, λ_{min} are

$$\begin{aligned}\lambda_{min} &\triangleq p_{min}(h_{min}\beta_{min}f_{min})^2[p_{max}(h_{max}\beta_{max}f_{max})^2 \\ &\quad + \hat{q}_{max}h_{max}^2 + r_{max}]^{-1} \\ &> 0\end{aligned} \tag{5.63}$$

$$\mu_{max} \triangleq \frac{q_{max}}{\hat{q}_{min}} \cdot n + \frac{h_{max}^2 p_{max}}{r_{min}} \cdot p \tag{5.64}$$

where n is the number of states and p is the number of measurements.

5.4 Experimental Fault Identification Result

To ensure the presence of a persistent excitation condition during the execution of the estimation algorithm, all of the estimations were performed while the actuator was tracking a square wave reference signal with 10 mm amplitude and 5 second period. This square wave signal is passed through a fourth order filter given in equation (5.65) to obtain a smooth reference signal with continuous fourth derivative. This smooth reference is used by the controller. The square wave signal, the reference signal, and the response are shown in Figure 5.11.

$$F_T(s) = \frac{1}{(s+9.99)(s+9.99)(s+10.01)(s+10.01)} \tag{5.65}$$

The states are (from equation (5.43)) x_1, x_2, x_3, and x_4 i.e. position of the actuator, velocity of the actuator, load pressure, and the spool position. The velocity is not measured using a sensor but is calculated using the Euler difference equation. The DDF gives an estimate for these measured states as well. The comparison between the calculated and measured velocity improves confidence in the numerical values for velocity used in the calculations. The estimated states are x_5 and x_6 i.e. the leakage coefficient and the friction. The states covariance estimate can be used to calculate the 95% confidence interval of the estimated values.

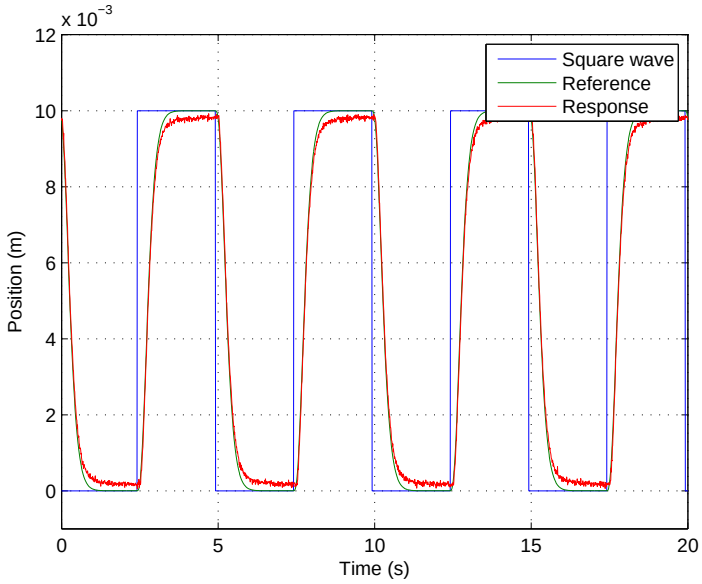
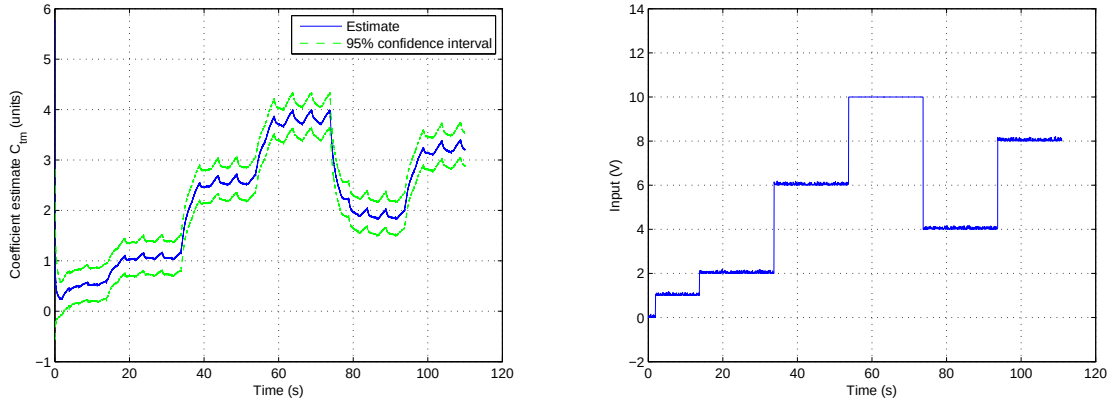


Figure 5.11. Position reference signal used for fault identification and response of the robust controller.

5.4.1 Random Step Fault

First example considered in fault identification is for a random step changes in the leakage fault. Figure 5.12(a) shows the estimation of the leakage coefficient for the fault model described in section 3.6. The voltage input supplied to the directional proportional valve controlling the leakage fault is also shown in Figure 5.12(b).

The estimates for the four states and the 95% confidence intervals are given in Figure 5.13. The confidence intervals are based on the noise covariance matrix given in equation (5.45). As can be seen, the uncertainty for states x_1, x_3, and x_4 are very small compared to that for x_2. This is because these state are obtained from actual measurements where as the state x_2 is calculated using Euler difference passed through a second order filter. The velocity calculations are not accurate at high frequencies, but the computational requirements are low and accuracy is sufficient for the current application. It is possible to artificially obtain low uncertainty for the x_2 state estimate, but this will result in small uncertainty for the estimate of the leakage



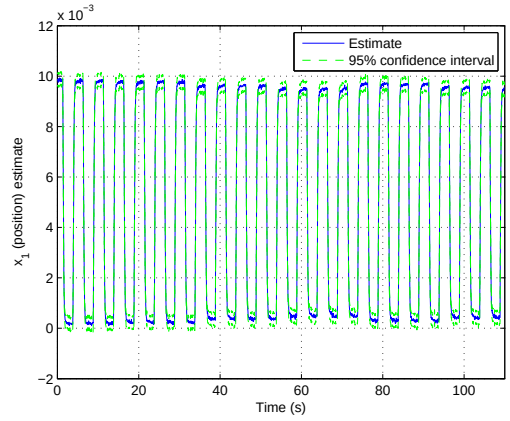
(a) Estimate of internal leakage coefficient C_{tm} (b) Voltage input to the leakage control valve equation (5.34)

Figure 5.12. Estimation of leakage coefficient and corresponding input to the directional proportional valve controlling internal leakage for a random step fault input.

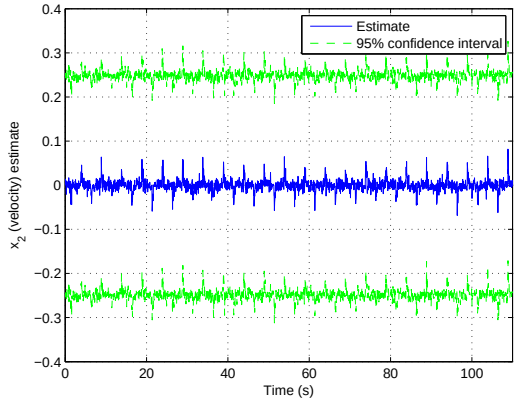
fault (Figure 5.12(a)) and friction coefficient leading to incorrect level of confidence in these estimate.

Figure 5.14 shows the comparison of the mean estimates for the states and the sensor measurements. The estimates match the measurements very closely. This is expected because one part of the update equations in the DDF is the measurement update equations (5.27)–(5.33), uses these sensor measurements explicitly.

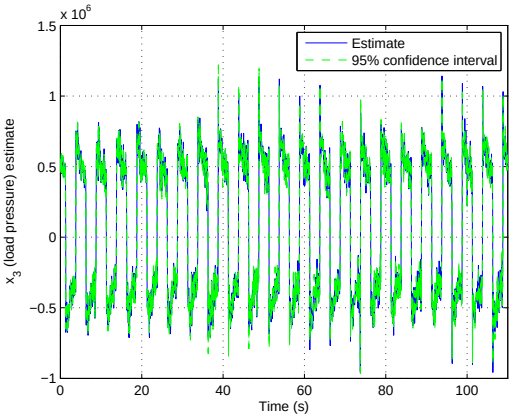
The mean estimate of the leakage rate based on the leakage parameter presented in Figure 5.12(a) and the measured leakage rate are shown in Figure 5.15. The estimate matches the measured leakage very well. The inset in Figure 5.15 showing a snapshot of comparison between 50 s and 60 s time interval confirms the accuracy of the estimate. The periodic spikes in the leakage estimate and measurement are observed because of the change in direction of the hydraulic piston. As the piston changes direction, the direction of the leakage also changes however, the flowmeter can only measure the magnitude of the leakage rate. The direction is inferred from the velocity. The flowmeter is a positive displacement type meter. So when the leakage



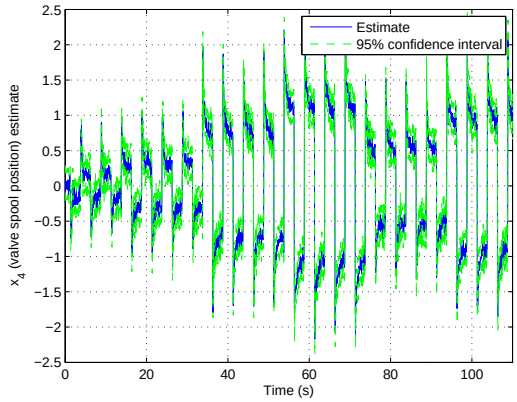
(a) Estimate of piston position x_1



(b) Estimate of piston velocity x_2



(c) Estimate of load pressure x_3

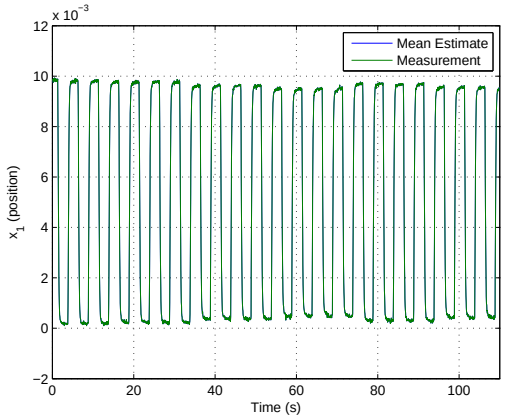


(d) Estimate of valve spool position x_4

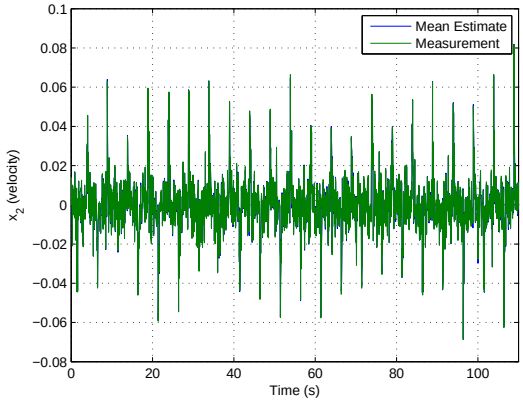
Figure 5.13. States estimated by DDF algorithm with corresponding 95% confidence intervals for random step fault.

changes direction, the measurement goes to zero instantaneously and ramps back up. Thus physical phenomenon is also estimated very well by the DDF algorithm. Furthermore, after every "jump" the leakage rate decreases. This is because once the piston comes to rest, the pressures in the two chambers of the cylinder start to equalize resulting in decrease in load pressure P_L and consequently the leakage flow rate. This behaviour is also estimated well by the filtering algorithm.

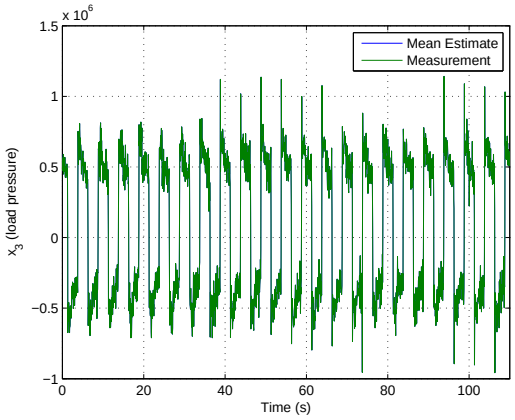
The corresponding friction estimate with 95% confidence interval is shown in Figure 5.16. The uncertainty is very large because the process noise covariance chosen



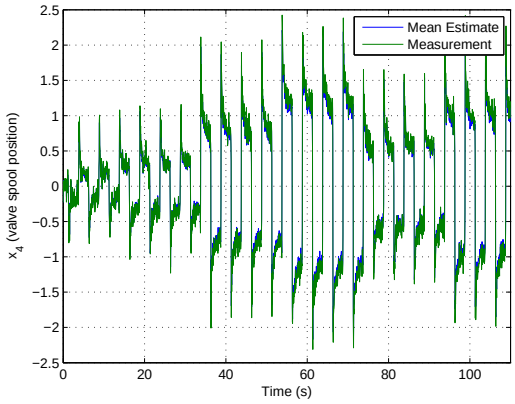
(a) Comparison of mean estimate and measurement of piston position x_1



(b) Comparison of mean estimate and measurement of piston velocity x_2



(c) Comparison of mean estimate and measurement of load pressure x_3



(d) Comparison of mean estimate and measurement of valve spool position x_4

Figure 5.14. Comparison of mean estimates of states from DDF algorithm and sensor measurements for random step fault.

for the friction state equation (5.44) is very high. The uncertainty can be reduced by decreasing the value of the process noise covariance from 1, but this leads to increasingly poor estimate of velocity (state x_2). This high value also compensates for other modeling errors.

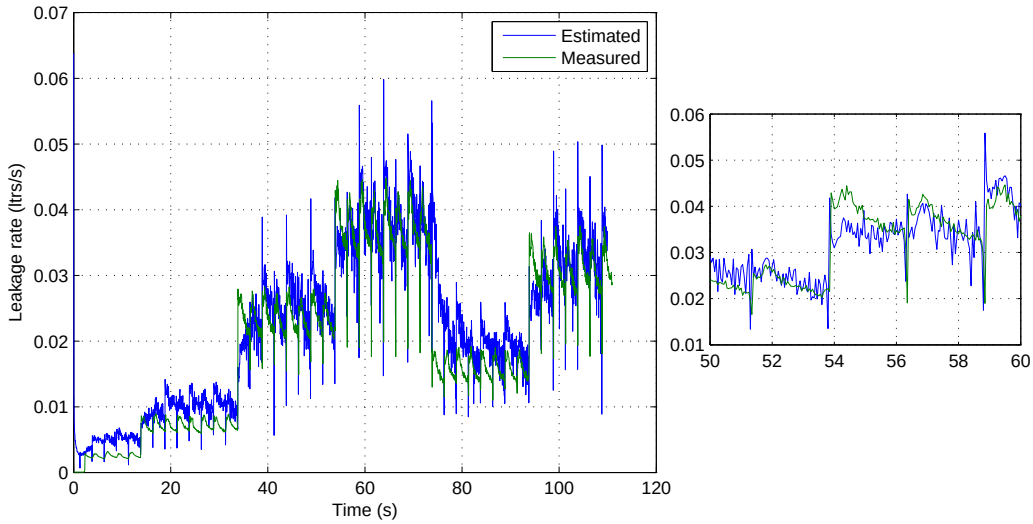


Figure 5.15. Comparison of estimated (mean) and measured leakage rate for random step changes in the fault level.

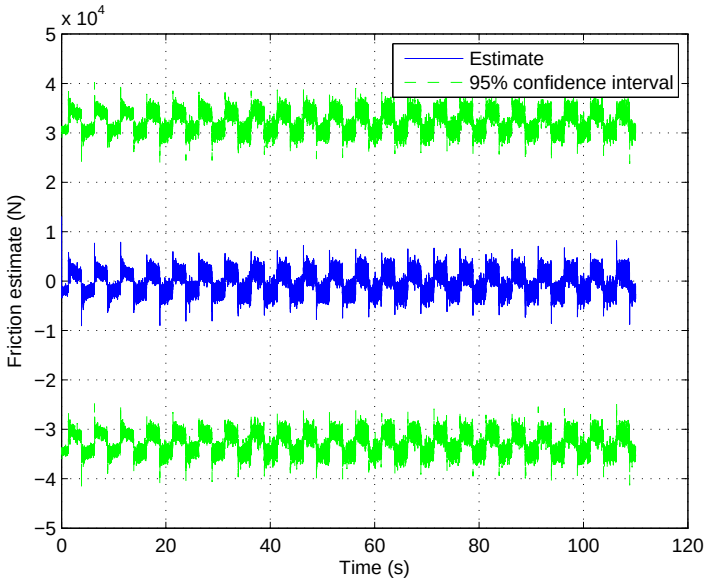


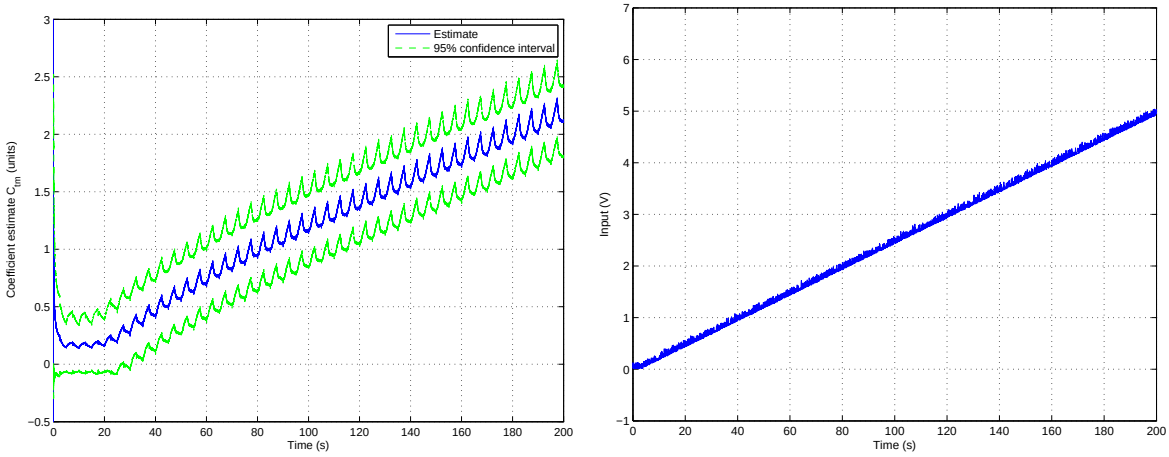
Figure 5.16. Estimate of friction between the piston and the cylinder for random step changes in the fault level.

5.4.2 Ramp Fault

The second example considered is a slow 200 s ramp fault. This example serves two main purposes:

1. to demonstrate ability to detect incipient faults and

2. to demonstrate ability to track fault as it grows over time



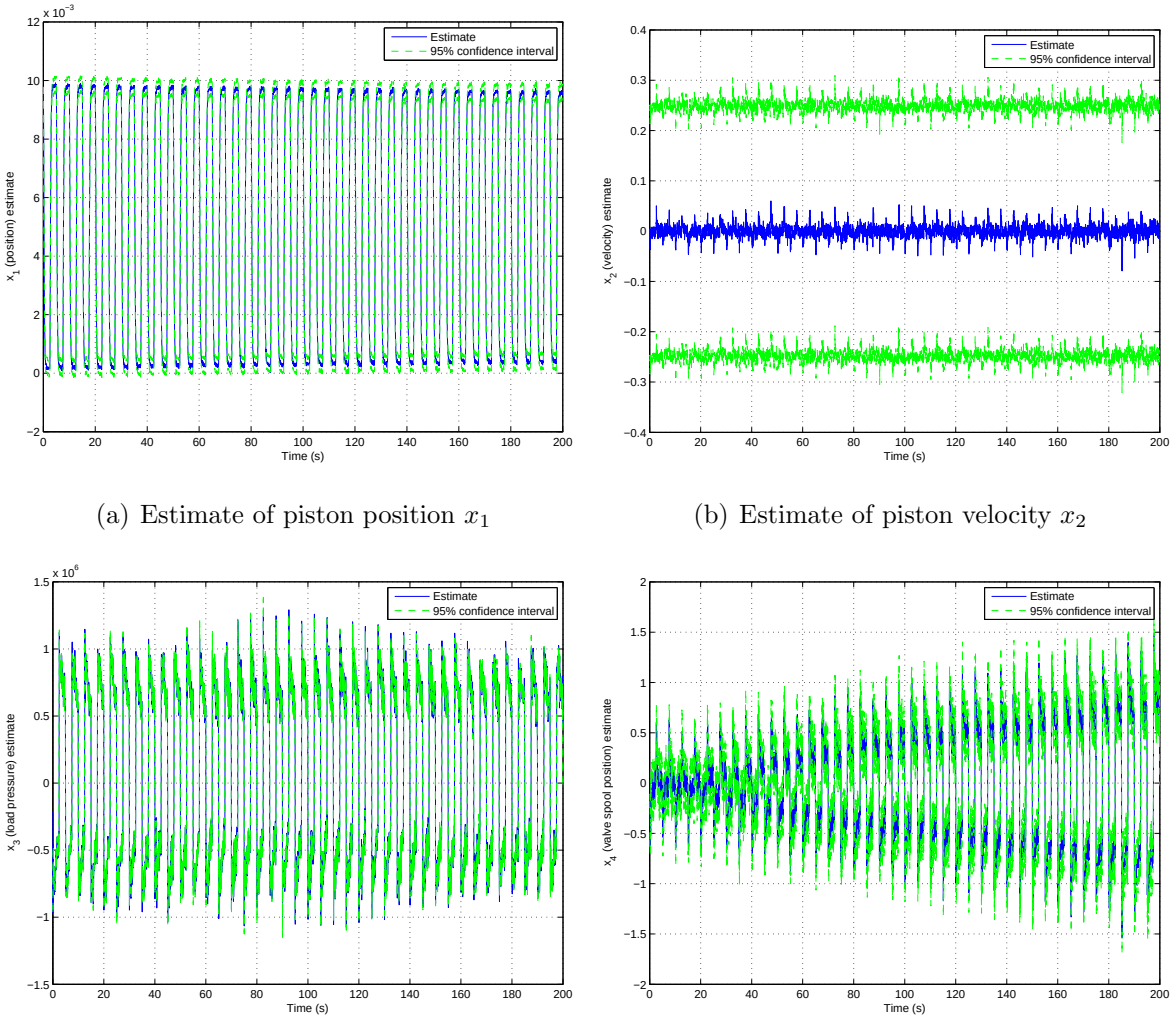
(a) Estimate of internal leakage coefficient C_{tm} equation (5.34)

(b) Voltage input to the leakage control valve

Figure 5.17. Estimation of leakage coefficient and corresponding input to the directional proportional valve controlling internal leakage for a ramp fault input.

The estimate of the leakage coefficient with 95% confidence interval is shown in Figure 5.17(a) and the voltage input supplied to the directional proportional valve controlling the leakage fault is also shown in Figure 5.17(b)

The estimates for the four states and the 95% confidence intervals are shown in Figure 5.18. The confidence intervals are based on the noise covariance matrix given in equation (5.45). Again, observation similar to those in section 5.4.1 regarding the uncertainty in estimates for states x_1, x_3, x_2, and x_4 can be made. Uncertainty in states x_1, x_3, and x_4 is very small compared to that for x_2. Figure 5.19 shows the



(a) Estimate of piston position x_1

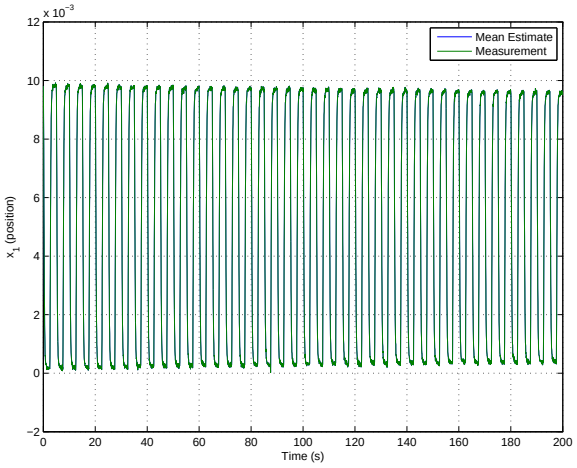
(b) Estimate of piston velocity x_2

(c) Estimate of load pressure x_3

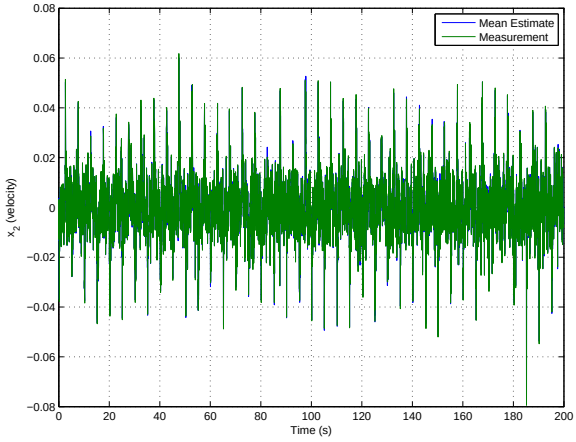
(d) Estimate of valve spool position x_4

Figure 5.18. States estimated from DDF algorithm with corresponding 95% confidence intervals for a ramp fault input.

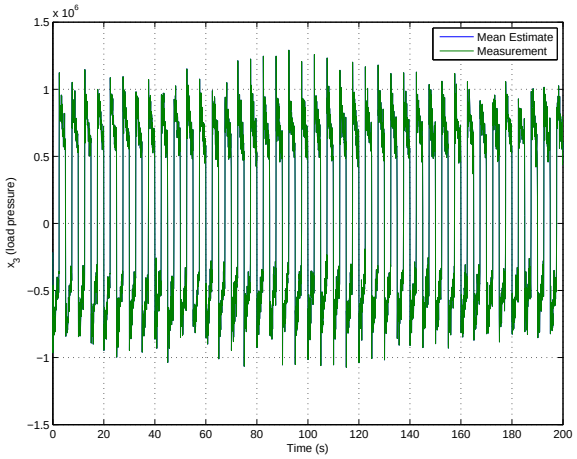
comparison of the mean estimate for the states and the sensor measurements. The estimates again match very closely with the measurements. An important observation regarding the effect of fault on the system can be made from Figure 5.19(d). Internal leakage fault essentially acts as an increased damping in the system, hence as the fault size increases, the displacement of the spool valve also goes up. This results from the robust control strategy trying to compensate for the decreased performance.



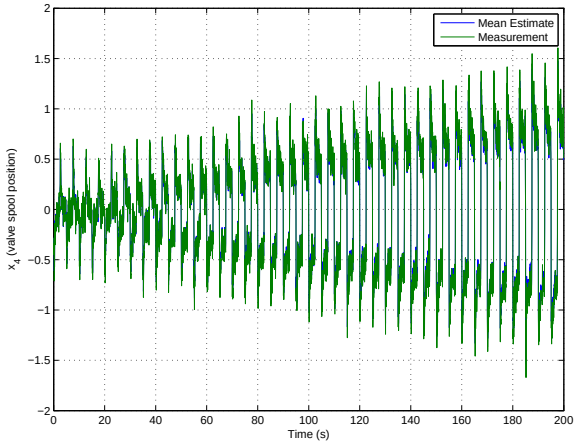
(a) Comparison of mean estimate and measurement of piston position x_1



(b) Comparison of mean estimate and measurement of piston velocity x_2



(c) Comparison of mean estimate and measurement of load pressure x_3



(d) Comparison of mean estimate and measurement of valve spool position x_4

Figure 5.19. Comparison of mean estimates of states from DDF algorithm and sensor measurements for a ramp fault input.

Figure 5.17(b) shows the results for a slow 200 s ramp input to the leakage control valve and Figure 5.21 and 5.20 show the estimation of the friction force and the leakage rate, respectively. These results indicate that the estimation algorithm is accurate and is able to locate incipient failures. The estimation algorithm has also been found to be very consistent over different runs and over different functions for

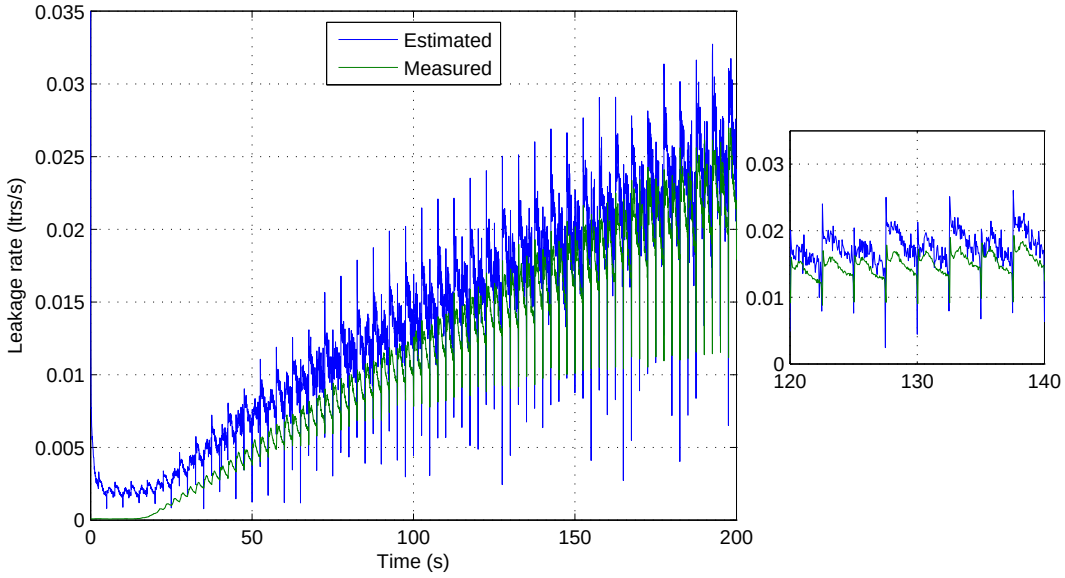


Figure 5.20. Comparison of estimated (mean) and measured leakage rate for ramp increase in fault level.

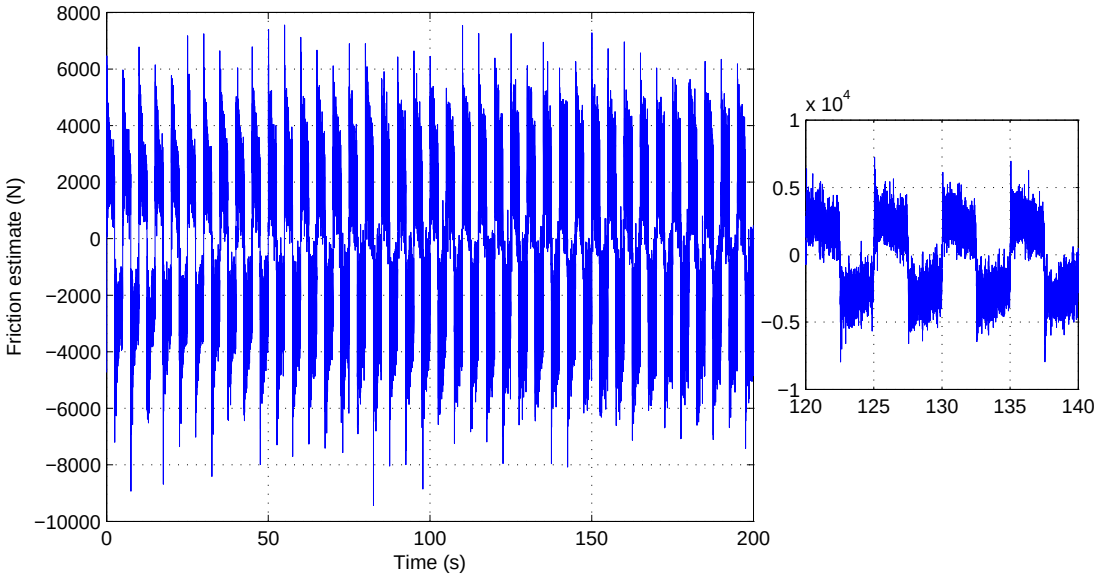
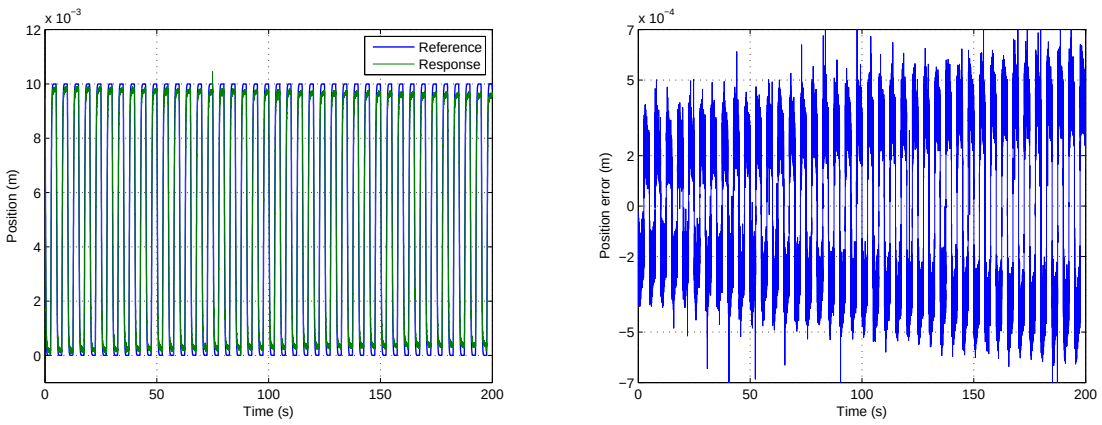


Figure 5.21. Estimate of friction between the piston and the cylinder for ramp increase in fault level.

the input signal to the leakage control valve. Very low leakage rates as low as .025 ltrs/s were identified successfully. However, there is an error in estimating extremely small leakage rates lower than .002 ltrs/s due to the presence of unmodeled load dynamics, which affect the system states and parameters, but are not considered in the estimation algorithm. This causes the algorithm to output a small leakage rate even though the leakage control valve is closed. Moreover, as leakage becomes greater then .01 ltrs/s, the estimates become very accurate as can be seen from the inset in Figure 5.20. Thus it can concluded that incipient failures can be detected as well as the algorithm is able to track the fault growth as a function of time

Remark 5.2. The main drawback of this methodology is that if there are two faults which can be represented as simple additive parameters in the same state equation, it would be difficult to distinguish between them. These two faults can be represented as a single parameter and its variation will be a combination of both faults.



(a) Performance degradation due to leakage

(b) Position error increases as leakage increases

Figure 5.22. Reference signal and degradation of performance of robust position control strategy with increasing internal leakage rate.

In the experimental runs, as the leakage was increased from 0 to .03 ltrs/s, the position error went up from .2 mm to .5 mm. The failure leakage rate for such actuators is estimated to be around .13 ltrs/s, which would lead to unacceptable performance

degradation. Hence, there is a need to compensate for the lost performance without affecting the rate of leakage. The next section details development of one such controller based on the robust control structure described in section 5.2.1.

5.5 Adaptive Fault Tolerant Control

Fast response times and high size to power ratios have made hydraulic systems very popular in the industrial sector for delivering large forces and torques [95]. Their industrial applications include positioning [96–98], active suspensions [99–101], material testing, aircraft, and industrial hydraulic systems. With increasing economic demand for high plant availability, safety of hydraulic systems, and risks associated with faults, reliability becomes an important factor in hydraulic applications.

A cost effective way of obtaining increased reliability and dependability in automated systems is through fault tolerant control (FTC). These requirements, when compounded with hazardous work conditions, necessitate the development of reliable and efficient fault identification techniques. Fault diagnosis of electro-hydraulic systems has been a subject of numerous studies. A variety of methods have been proposed in the literature for early detection of faults in dynamical systems. These methods are broadly divided into two main types: hardware redundancy based and analytical redundancy based methods [37]. Analytical redundancy based methods are further divided into signal model based methods and process model based methods [38, 39, 135]. An excellent review of fault detection and diagnosis methods is provided in several survey papers [10, 40–43]. Fault diagnosis and identification is a supervisory process and it alone cannot maintain performance and functionality of the faulty system in absence of fault tolerant control.

Fault tolerant control is a control method that can accommodate system faults and failures automatically and maintain overall system stability and performance. A fault tolerant controller can be classified into two main types: passive and active.

Passive approaches make use of robust control techniques to ensure that the system remains insensitive to certain class of faults with bounded magnitudes and do not use any online fault information. The faulty system continues to operate with the same controller and the same control structure and should continue to maintain the designed performance under a set known faults and failures. However, since the design is developed for a certain class of faults, no performance guarantees can be made in the presence of unforeseen faults [136]. Furthermore, since the designed control system must accommodate a class of faults, the design is usually conservative. Methods that have been traditionally used in passive fault tolerant control include quantitative feedback theory [122], frequency domain design [137], sliding mode control [60, 61], and linear matrix inequalities [63]. The advantages that make passive approaches appealing are that they do not require fault identification schemes and are usually straightforward to implement.

Active approaches on the other hand are usually variable in their structure. They reconfigure the control actions to maintain stability and performance within acceptable limits. Active fault tolerant methods can be further classified into projection based and online redesign methods depending on the way the post fault controller is developed. Some of popular techniques include the pseudo-inverse method [24], eigenstructure assignment [46], multiple model method [5,25], model following [9,70], adaptive control [4, 7, 65] etc. An extensive bibliographical review on active fault tolerant methods is available in Zhang [21] and Kanev [138].

In this section, a fault tolerant control strategy is derived for faults that can be modeled parametrically. The fault estimate from section 5.3 is used in an adaptive control scheme to compensate for the level of fault. The original controller given in equation (5.14) is modified as follows:

$$x_v = x_{v_a} + x_{v_s} \tag{5.66}$$

The control action is broken into two parts: x_{v_a} is the model compensation term while x_{v_s} is the robustifying term. The robustifying term is further divided into two terms: $x_{v_{s_1}}$, which is the robust feedback term and $x_{v_{s_2}}$ which ensures that the adaptation

is passive. In other words, the adaptation does not affect the robustness properties of the control action [112].

$$x_{v_a} = \frac{1}{\varsigma}\left(\dot{F}_d + \frac{4\beta_e}{V_t}A^2\dot{x}_L - \hat{Q}_{leak}\frac{4\beta_e}{V_t}A\right), \tag{5.67}$$

$$\varsigma = \frac{C_d w}{\sqrt{\rho}}\frac{4\beta_e}{V_t}A\sqrt{P_s - \mathrm{sgn}(x_v)P_L}$$

$$x_{v_s} = x_{v_{s_1}} + x_{v_{s_2}} \tag{5.68}$$

$$x_{v_{s_1}} = -K_{s_1}(F - F_d) \qquad \text{robust feedback} \tag{5.69}$$

$x_{v_{s_2}}$ which satisfies following conditions

$$\begin{cases} (F - F_d)\varsigma x_{v_{s_2}} \leq 0 \\ (F - F_d)\left(-\tilde{Q}_{leak}\frac{4\beta_e}{V_t} + \varsigma x_{v_{s_2}} + d\right) \leq \epsilon_2 \end{cases} \tag{5.70}$$

Consider the following Lyapunov function:

$$V = \frac{1}{2}(F - F_d)^2 \tag{5.71}$$

then,

Theorem 5.2 (Main Result)**.** *Let the leakage parameter estimate be updated by the stochastic adaptation law as given by (5.27)–(5.33), then the control law (5.66)–(5.69) guarantees that all signals are bounded. Furthermore, V as given in (5.71), is bounded above by*

$$V(t) \leq \exp(-2K_{s_1}t)|V(0)|^2 + \frac{\epsilon_2}{2K_{s_1}}\left[1 - \exp(-2K_{s_1}t)\right] \tag{5.72}$$

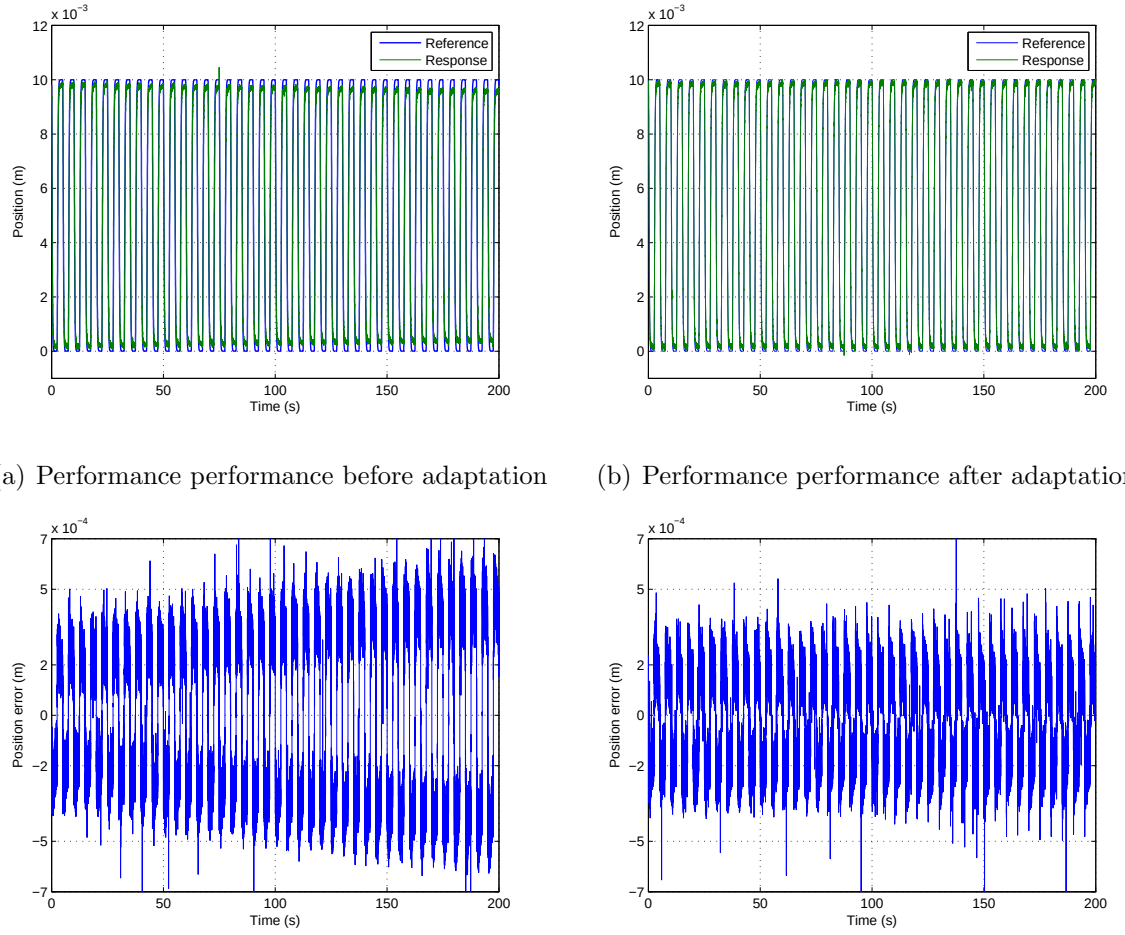
Proof.

$$\begin{aligned} \dot{V} &= (F - F_d)(\dot{F} - \dot{F}_d) \\ &= (F - F_d)\left(-\tilde{Q}_{leak}\frac{4\beta_e}{V_t}A - K_{s_1}(F - F_d) + \varsigma x_{v_{s_2}}\right) \\ &\leq -K_{s_1}(F - F_d)^2 + \epsilon_2 \\ &= -2K_{s_1}V + \epsilon_2 \end{aligned}$$

and the result follows. It remains to show that $\tilde{Q}_{leak}$ is bounded, which clearly follows from Theorem 5.1.

The supermartingale property of the current stochastic system can also be used to prove similar exponential stability of the estimation process as noted in [139]. □

5.6 Adaptive Control Experimental Results



(a) Performance performance before adaptation

(b) Performance performance after adaptation

(c) Position error before adaptation

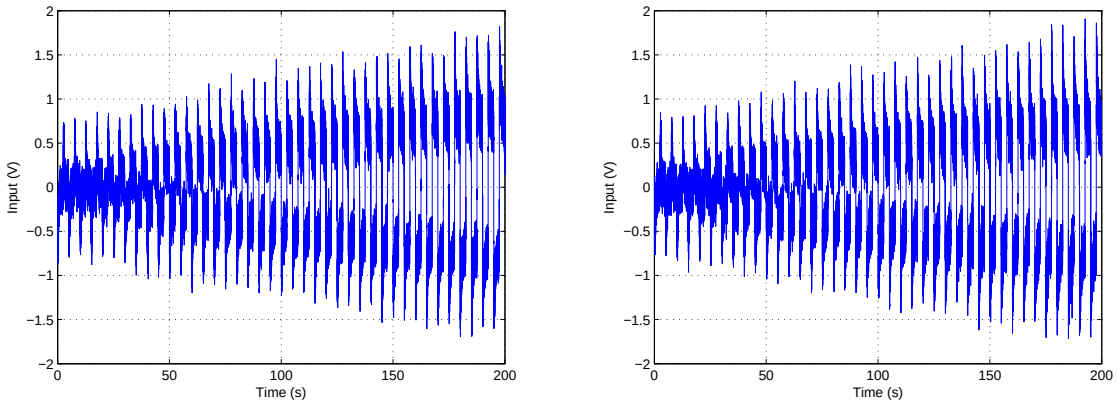
(d) Position error after adaptation

Figure 5.23. Comparison of positioning performance in absence and presence of adaptive control.

The performance improvement obtained due to the new adaptive controller can be clearly inferred from the position performance graph shown in Figure 5.23. The figure shows the comparative results for the response to a square wave reference in absence and presence of adaptation. Figures 5.23(a) and 5.23(b) show the response of the position controller without and with adaptation respectively. Figures 5.23(c) and 5.23(d) show the corresponding errors. For this simulation study, the input the valve controlling the internal leakage was varied from 0 to 5 V over a period of 200 s as shown in Figure 5.17(b). As can be seen in Figure 5.23(c), in absence of adaptation, the position error keeps increasing as the fault size increases. Starting at zero leakage, the $\pm.2$ mm error has increased to almost $\pm.5$ mm for .03 ltrs/s leakage rate. For aircraft actuators, the failure leakage rate is estimated to be around .13 ltrs/s, for which the performance degradation would be unacceptable large. On the other hand in presence of adaptation, as seen in Figure 5.23(d), the error remain bounded throughout. The comparison of input profiles in absence and presence of adaptation are shown in Figure 5.24. The difference in the inputs is barely perceptible. As is clear from this figure, the addition of adaptation doesn't command extra control effort and in turn does not cause saturation of the control inputs, which is a critical drawback of adaptive control strategies. As a result, the new controller does not affect the measured/estimated leakage significantly as can be seen from Figure 5.25.

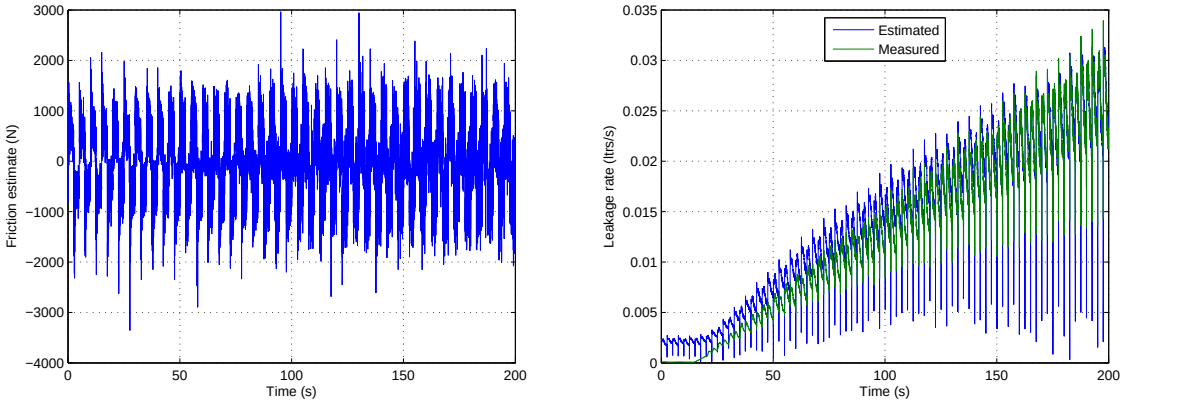
The improvement in performance is more striking when depicted in a single simulation run as shown in Figure 5.26. This simulation was performed for a leakage input of 5 V corresponding to a .03 ltrs/s leakage rate. There is an instantaneous reduction in tracking error as soon as the adaptation is turned on at 38 s as seen in Figure 5.26(a). Figure 5.26(b) shows that the effect of adaptation on the control input is negligible. There is a slight disturbance to the leakage estimate at 38 s but the level and pattern of leakage remains the same as evidenced in Figure 5.26(d).

Remark 5.3. Using a stochastic fault identification strategy is advantageous compared to the deterministic strategies that are typically used in adaptive control [140] because they can better deal with modeling errors and unmodeled dynamics. A comparative



(a) Input profile before adaptation (b) Input profile after adaptation

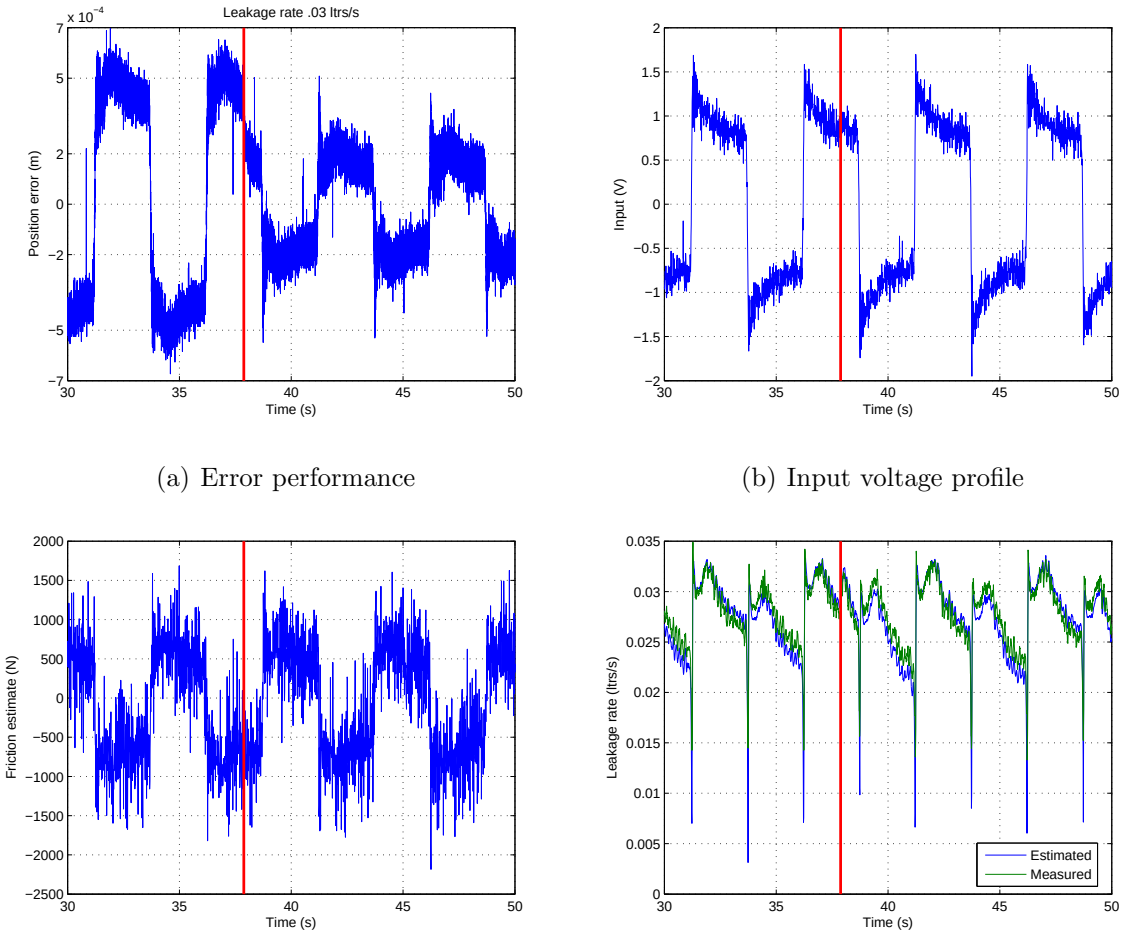
Figure 5.24. Comparison of inputs to the directional proportional valve in absence and presence of adaptation.



(a) Estimation of friction (b) Comparison of estimated and measured leakage

Figure 5.25. Estimation of friction and leakage rate for a slow ramp input to the leakage control valve in presence of adaptation.

study was performed between least squares estimate (LSE) and DDF. It was found that the LSE was unable to estimate the fault at all. This also manifests itself in the inability of the Kalman Filter and Extended Kalman Filters to estimate the fault since least squares is essentially just a special deterministic case of Kalman Filtering [127]. It is also possible to establish whether there are unmodeled dynamics in the system.



(a) Error performance

(b) Input voltage profile

(c) Friction estimate

(d) Leakage measurement and estimate

Figure 5.26. Performance improvement due to introduction of adaptation at time 38 s (after the red line).

This can be done by adding a constant parameter to each of the state space equations and estimating the parameter in the same manner as the fault parameter. If the new parameter estimates are Gaussian noise, it implies that no unmodeled dynamics are present in that particular state equation.

5.7 Sensitivity to Process Noise Covariance

The process noise matrix in equation (5.44) was chosen heuristically based on confidence in the mathematical model used for the system given in equation (3.11). For example for the leakage state estimate, a noise covariance value of 8×10^{-6} was used assuming that the leakage fault is caused by wear which is a very slow process. This value determines the speed at which the leakage coefficient estimate can respond to changes in the fault and the variability of the estimate. Figure 5.27 shows the variation of response for different process covariance values. The average steady state value for a 5 V input is around 2.66. Figure 5.28 shows the variation of time required to reach 90% of steady state value with process covariance value used for the leakage coefficient state estimate. The basic principle of DDF filter follows the Kalman filter with a prediction update and a measurement update. The behaviour can be explained by looking at the simple update equations of a discrete Kalman filter. The first set of equations (5.73)–(5.74 represent the prediction update and the second set (5.75)–(5.77 represent the measurement update.

$$\bar{x}(k+1) = A\hat{x}(k) + Bu(k) \tag{5.73}$$

$$\bar{P}(k+1) = A\hat{P}(k)A^T + Q \tag{5.74}$$

$$K(k+1) = \bar{P}(k+1)H^T \left(H\bar{P}(k+1)H^T + R\right)^{-1} \tag{5.75}$$

$$\hat{x}(k+1) = \bar{x}(k+1) + K(k+1)\left(y(k+1) - H\bar{x}(k+1)\right) \tag{5.76}$$

$$\hat{P}(k+1) = \left(I - K(k+1)H\right)\bar{P}(k+1) \tag{5.77}$$

As the *a priori* estimate error covariance $\bar{P}(k+1)$ approaches zero, or after the error covariances have converged and if Q is small, the gain K weighs the residual less heavily. Specifically,

$$\lim_{\bar{P}(k+1)\to 0} K(k+1) = 0 \tag{5.78}$$

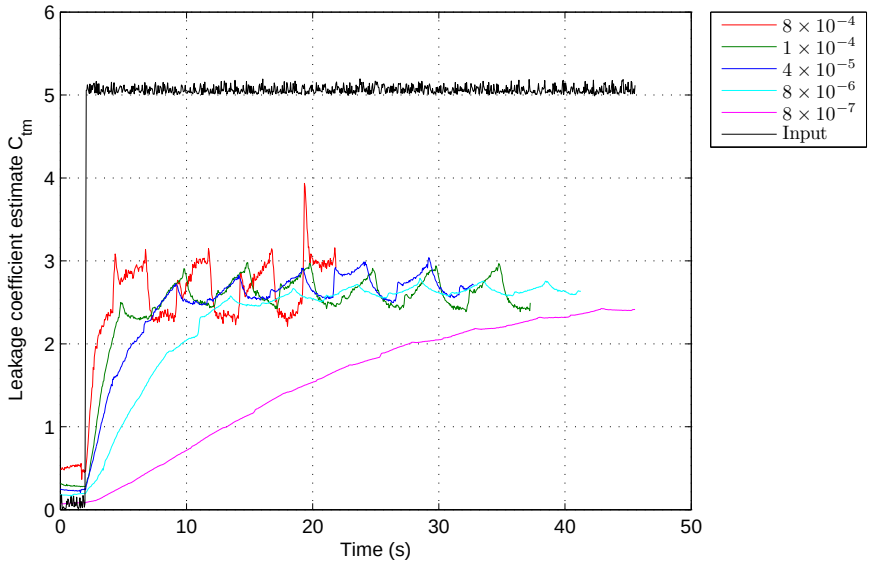


Figure 5.27. Behaviour leakage parameter C_{tm} estimation for different process covariance values.

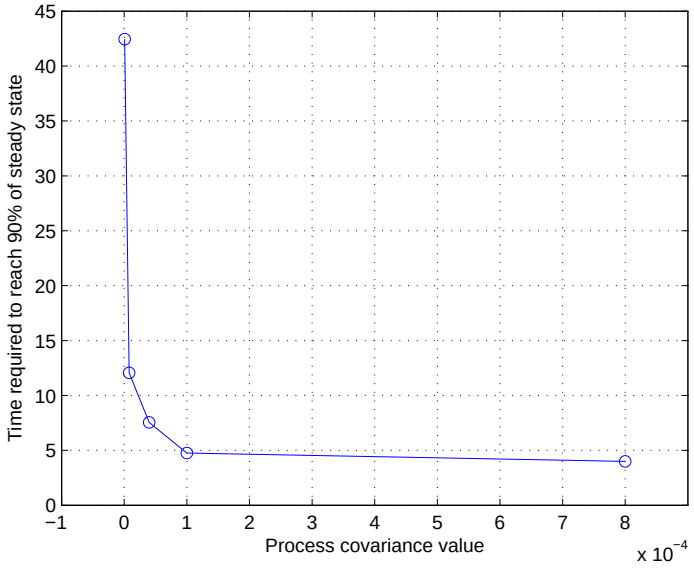


Figure 5.28. Sensitivity of leakage parameter estimate response to corresponding process covariance values.

When this happens, the actual measurement is trusted less and less, while the prediction is trusted more and more. This results in a large inertia resulting in slow filter response to changes in measurements. Hence, as the process noise covariance is decreased, the time taken to reach the 90% value increases as seen clearly in Figure 5.28. This also results in smoother response.

6. PROGNOSIS BASED CONTROL

In the previous chapters, models for the aircraft, the hydraulic actuator, and the degradation were presented. Control structure for the aircraft model was derived and an adaptive fault tolerant control strategy was proposed for the hydraulic actuator powering the control surface. In this chapter the rest of the modules described in the approach will be developed.

6.1 Prognosis

The location of the prognosis module in the implementation scheme is shown in Figure 6.1. The prognosis module is a static module obtained through extensive optimization. The rational for developing this module came from Model Predictive Control (MPC) literature.

6.1.1 Model Predictive Control

MPC has a long history in the field of control engineering. It is one of the few areas that has received on-going interest from researchers in both the industrial and academic communities. Three major aspects of model predictive control make the design methodology attractive to both engineers and academics. The first aspect is the design formulation which lends itself naturally to a multivariable system framework. The second aspect is the ability of the method to handle both soft constraints and hard constraints in a multivariable control framework. And the third aspect is the ability to perform online optimization. These aspects results in number of advantages such as ability to explicitly consider actuator and state saturation, handle non-minimumphase and unstable processes, and handle structural changes.

The conceptual structure of MPC is depicted in Figure 6.2 [141].

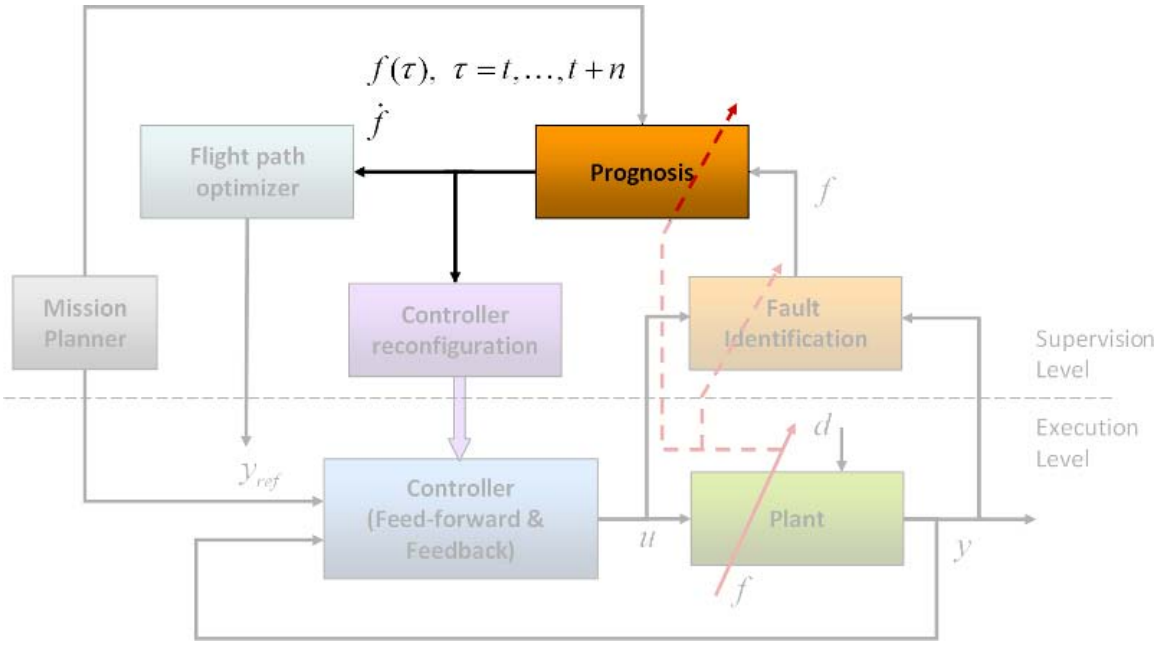


Figure 6.1. Block diagram depicting location of the prognosis module.

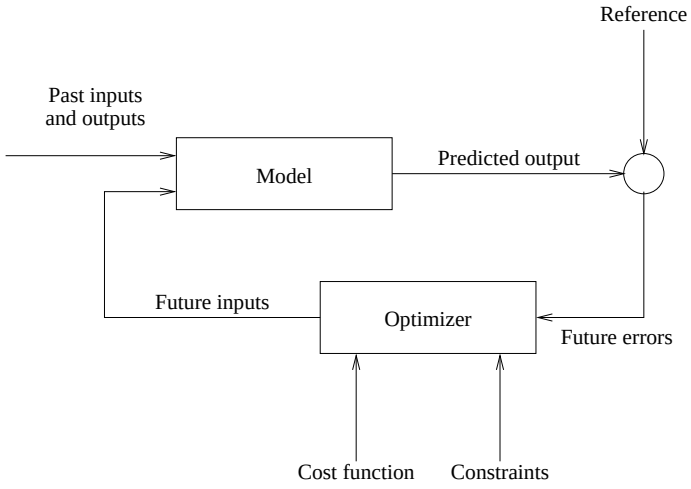


Figure 6.2. Basic structure of MPC.

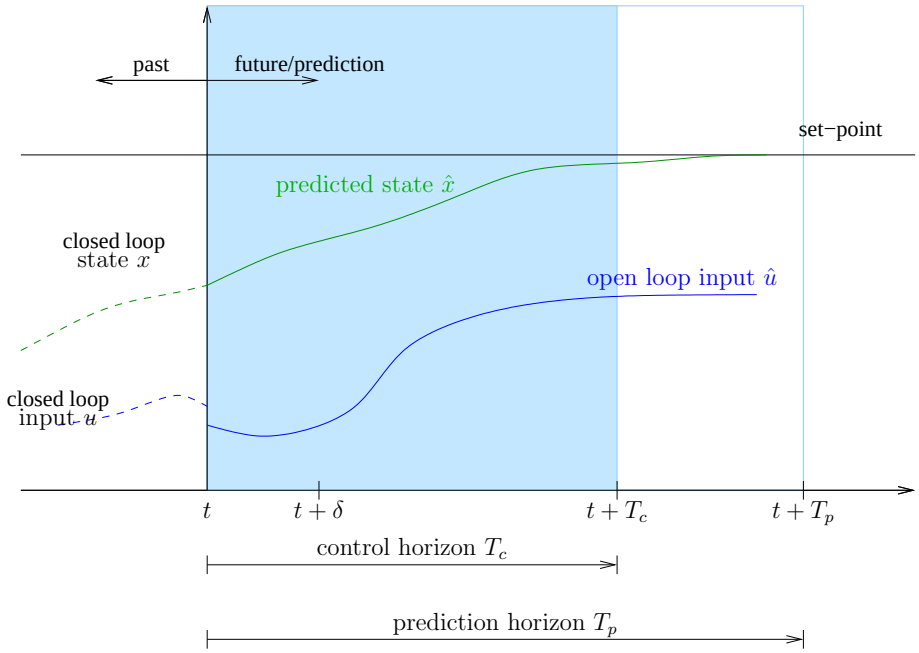


Figure 6.3. Basic principle of MPC.

The structure employs an explicit model of the plant to be controlled to predict future output behaviour. This prediction capability is used to formulate and solve an optimal control problem online. The objective function minimized is the error between the predicted output and desired reference. In general, the MPC problem is formulated to solve a finite horizon open-loop optimal control problem subject to system dynamics and constraints involving states and controls. Figure 6.3 shows the general principle of MPC. Given measurements at time t, the controller predicts the future dynamic behaviour of the system over a prediction horizon T_p and calculates (over a control horizon $T_c < T_p$) the input required to minimize a predetermined control objective. In absence of any disturbances and model mismatch, and if the optimization problem could be solved for infinite horizons, then the control input calculated at time t can be applied to the system for all time $\tau \geq t$. However this is not true in general due to modeling errors and disturbances. This results in predicted behaviour being different from actual. Hence it is imperative to incorporate some feedback mechanism. To do this, the calculated input is implemented only until the

next measurement is available at which time, the prediction–optimization process is repeated.

The objective of the prognosis based control strategy is to minimize the degradation of the actuator while extracting best possible performance from the aircraft. In other words, allow graceful degradation of performance resulting in extension of component and system life. This problem lends itself very well into the MPC framework especially because of presence of nonlinear plant dynamics and various hard and soft constraints. The problem under consideration was formulated as a nonlinear MPC problem as follows:

$$\min J = w_r(t_f)^T W w_r(t_f) + \int_0^{t_f} e_y^T Q e_y dt \tag{6.1}$$

$$\text{subject to } \dot{x} = f(x,u,t) \qquad \text{system equations} \tag{6.2}$$

$$u \in \begin{bmatrix} u_{min} & u_{max} \end{bmatrix} \qquad \text{input constraints} \tag{6.3}$$

$$x \in \begin{bmatrix} x_{min} & x_{max} \end{bmatrix} \qquad \text{state limits} \tag{6.4}$$

$$\lambda \in \begin{bmatrix} \lambda_{min} & \lambda_{max} \end{bmatrix} \qquad \text{parameter limits} \tag{6.5}$$

$$g(x,u,t) < 0 \qquad \text{other constraints} \tag{6.6}$$

$$x(0) = x_0 \qquad \text{initial conditions} \tag{6.7}$$

where, J is the objective function to be minimized. w_r is the degradation of the actuator, t_f is the final time, e_y is the output error. Other constraints in eq. (6.6) include constraints on degradation level, time domain constraints such as rise time t_r, settling time t_s, maximum overshoot m_p etc. The first part of the objective function in equation (6.1) ensures that the degradation at the end of the mission is minimized and second part ensures that the system follows the trajectory as closely as possible. The system equations $f(x,u,t)$ include all the plants as described in Chapter 3 and given in equations (3.11), (3.14), (3.23), (3.24), and Figure 3.10.

As mentioned before, the ideal input to an MPC problem is the solution of an infinite horizon optimal control formulation. When a finite prediction horizon is used, the actual closed loop input and state trajectories will differ from the predicted open-

loop trajectories. Hence an optimization has to be solved every time new information becomes available. This repeated minimization over a finite horizon results in a different solution than an infinite horizon optimal problem. Secondly, no guarantees for closed loop stability can be made. Furthermore, in application were the systems are sufficiently complex, the optimization is very computationally expensive. This is especially true where sampling instances are very small. For example for a hydraulic actuator sampling of the order of 1–2 ms is required. Hence to work within the available computational limitations and to ensure stability during a mission, the optimization was broken up into two components:

1. The off-line component was a map generated using local optimization and
2. The online component which performs the actual reconfiguration

6.1.2 Off-line Optimization

Optimization problem formulation given in equation (6.1), can be used to generate an extensive map of performance–degradation tradeoff. This map provides an empirical function to calculate compromise between the best possible performance achievable for a given level of degradation. This optimization was performed for simple maneuvers which could then be clubbed together to generate a mission profile. For example, in the longitudinal plane, the map can be generated for a altitude changes of different sizes. The optimization essentially generates a series of gains for the aircraft controller designed in (5.1)–(5.7). Figure 6.4 shows a tradeoff for 1000 feet change in altitude at different response speeds. The constraints imposed are step response constraints in terms of rise time, settling time, and maximum overshoot/undershoot. Actuator saturation and rate limit constraints are also imposed. Full nonlinear hydraulic actuator model was used during the optimization.

A block diagram of the combined model used during generation of this tradeoff is shown in Figure 6.5. The stability for each of these individual maneuvers is assured

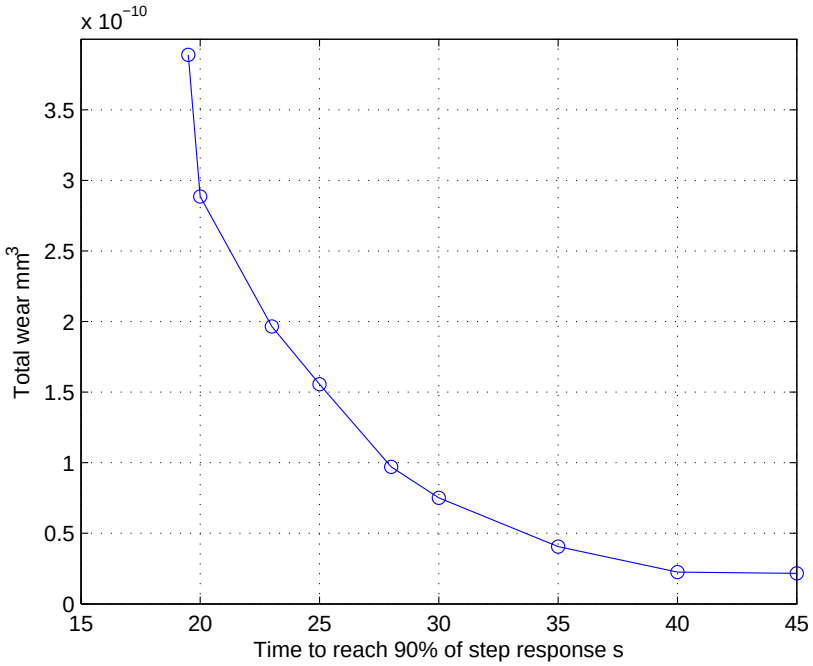


Figure 6.4. Trade-off between response speed (rise time) and degradation.

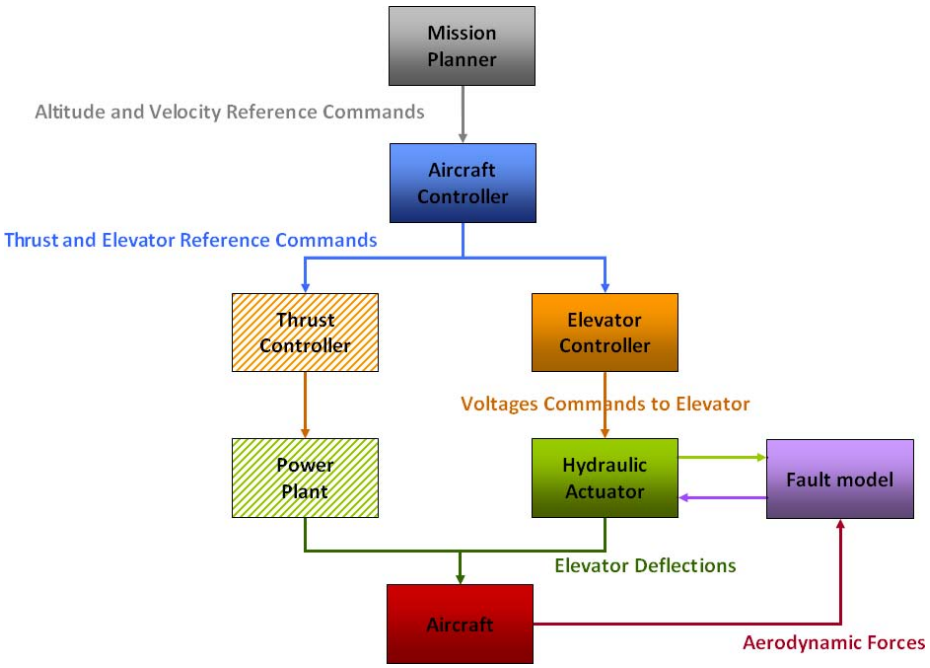


Figure 6.5. Block diagram of the combined model of the aircraft, the hydraulic actuator, and the fault; and their interactions.

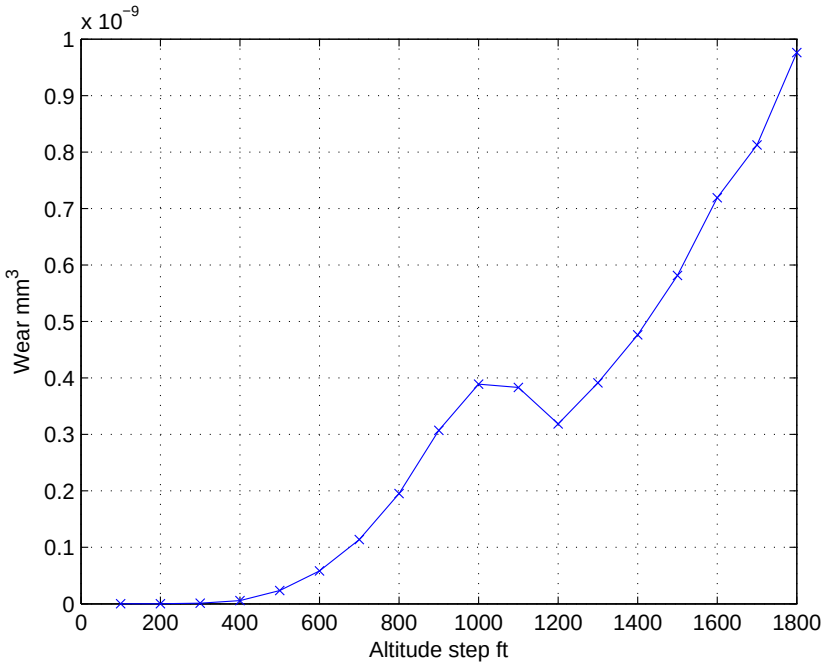


Figure 6.6. Trade-off between degradation observed as a function of altitude reference step.

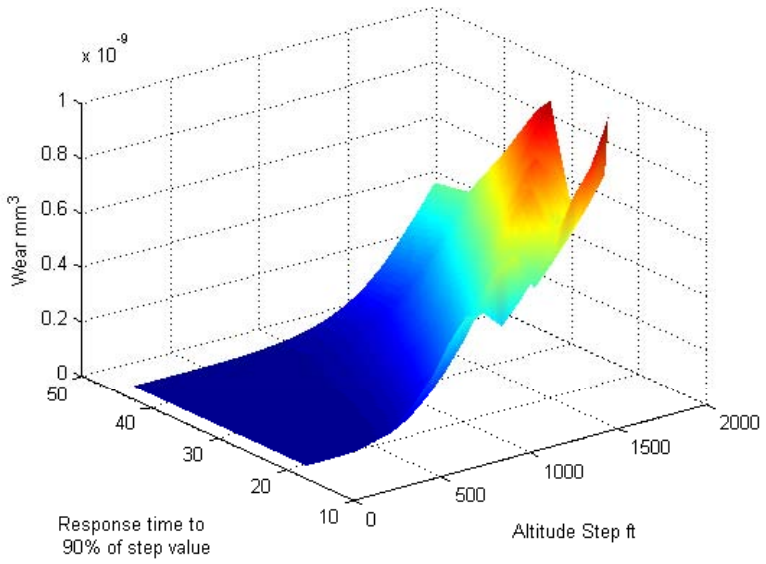


Figure 6.7. Trade-off map generated using simulations between degradation on z-axis, response speed on y-axis, and altitude steps on x-axis. In other words, trade-off between prognosis, performance, and mission profile.

by the control design which can be considered as full state feedback LQR design. An LQR design has been proven to possess following stability properties [142]:

1. Upward gain margin is infinite
2. Downward gain margin is atleast 1/2
3. Phase margin is at least $\pm 60^{\circ}$

For multivariable systems, these gain and phase margins occur independently and simultaneously in all control channels, thus guaranteeing stability over the entire mission. A similar trade-off can be generated between the size of the step and degradation as shown in Figure 6.6. Figures 6.4 and 6.6 can then be combined to generate a 3-D map of performance v/s mission v/s prognosis as shown in Figure 6.7. This map was generated using simulations with the wear model described in section 3.6.1.

6.1.3 Online Optimization: Supervisory Layer

This section describes the two blocks from the approach block diagram highlighted in Figure 6.8. From Figure 6.7, it is evident at there are two ways to reduce degradation

1. through the selection of different response speeds, or
2. through the modification of the "subgoals" of the mission: for example, by breaking a large altitude step requirement into smaller step requirements

The supervisory control layer does both of the above in an optimal fashion through a mixed integer programming strategy to find the best possible response while keeping the degradation for a particular mission below a pre-specified level. This can also be interpreted as minimizing the degradation given the mission and performance constraints. This was achieved using the following objective function:

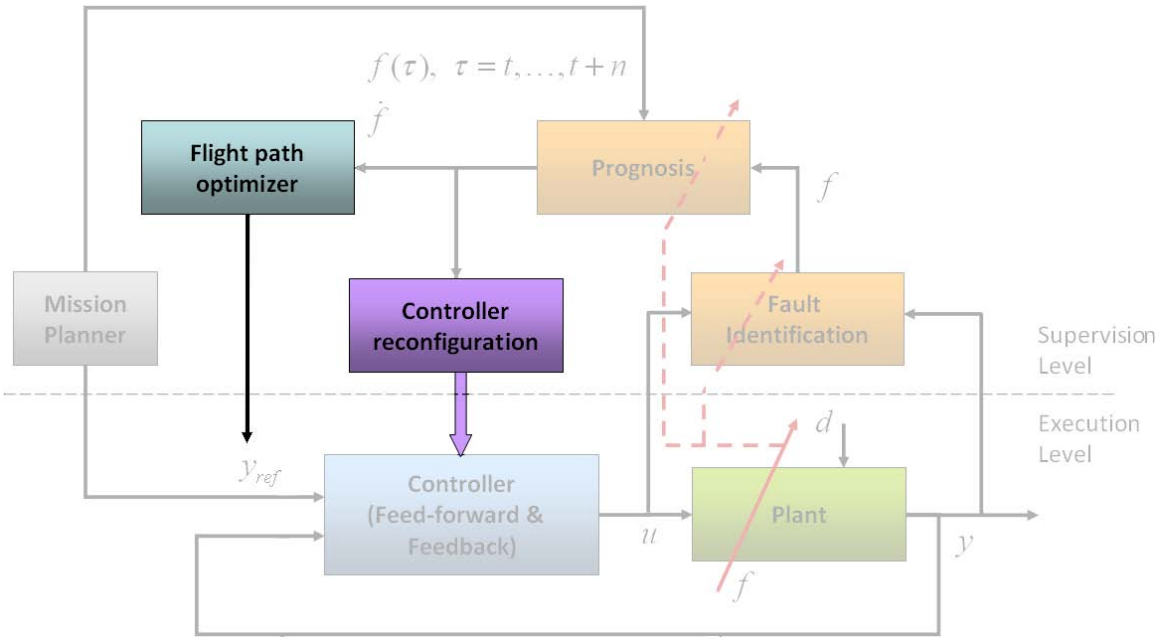


Figure 6.8. Block diagram depicting the location and interactions of the supervisory layer.

$$\min_{t_r, \mathrm{n_{steps}}} J = \sum_{i=1}^{\mathrm{n_{steps}}} t_r\left(h\left(i\right)\right) \tag{6.8}$$

$$\text{subject to } w_r\left(t_r\right) \leq w_{r,\mathrm{commanded}} \tag{6.9}$$

The optimization minimizes the sum of rise times (performance) of step response of the aircraft. This optimization is also performed over the the number of steps in the trajectory (assuming that the trajectory is supplied in terms of step changes in altitude for a longitudinal aircraft model). Hence as an altitude change is broken into one or more smaller altitude changes, the size of the optimization function changes. This results in a nonlinear mixed integer programming problem. For sufficiently small number of cases, a part of the optimization can be searched exhaustively and for each search a relaxed linear programming problem can be solved. The altitude steps which are allowed to be divided into smaller step were thresholded to restrict the

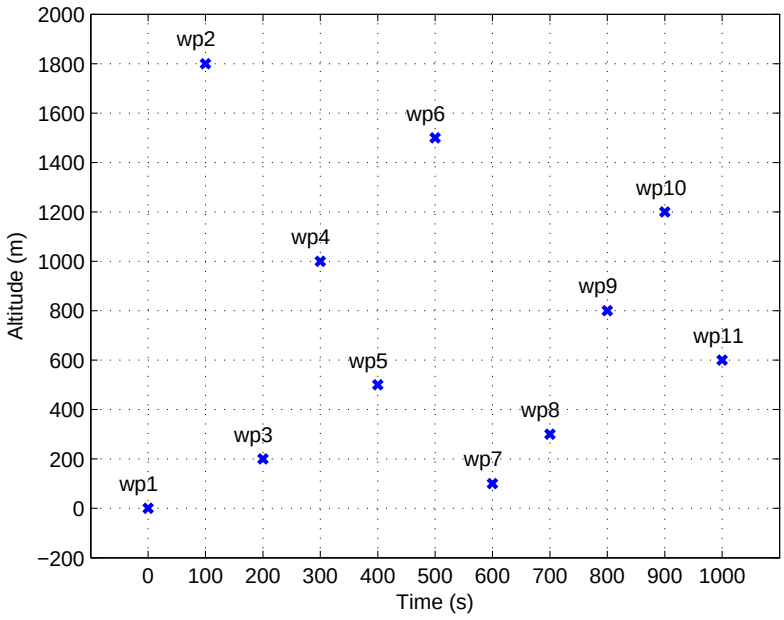


Figure 6.9. Reference waypoint map used for simulation.

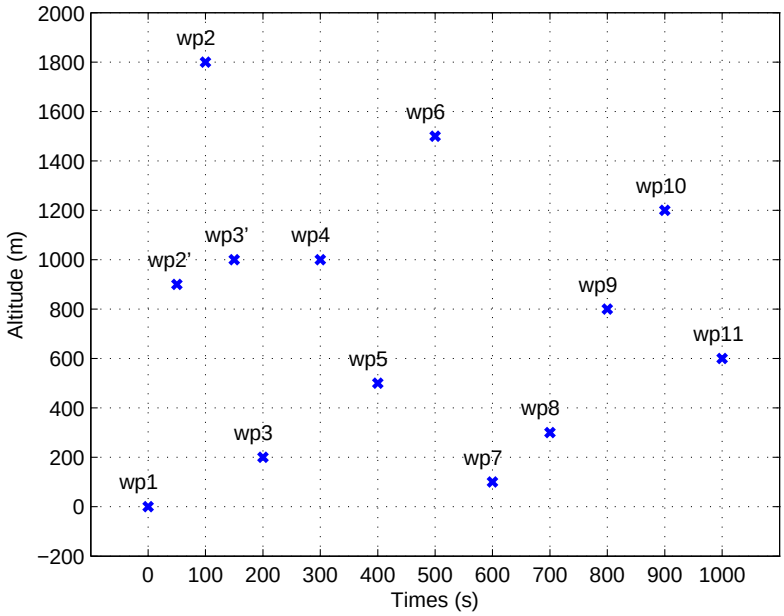


Figure 6.10. Waypoint map obtained after reconfiguration through optimization.

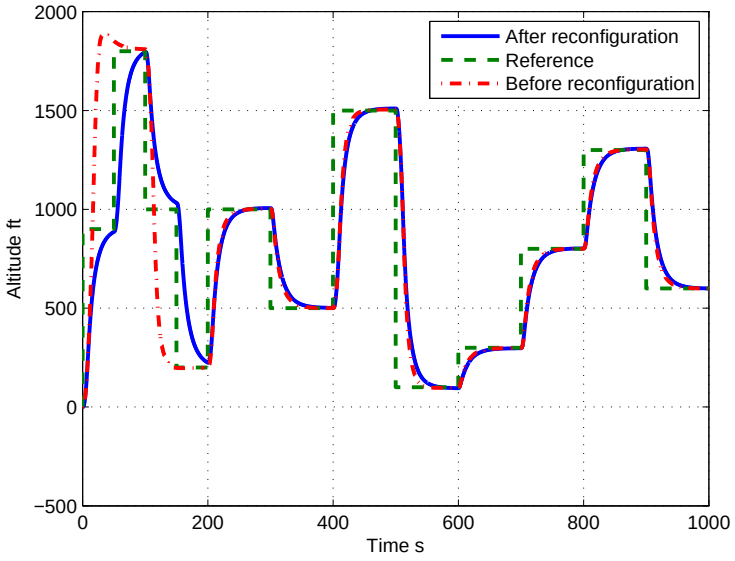
dimensionality of the optimization problem. As an example, consider the reference trajectory given in terms of a series of waypoints in the longitudinal plane shown in Figure 6.9.

The step response was used as performance measure in this case so that the aircraft can get to the desired altitude as fast as possible. This gives a buffer in case there is fault allowing more time to get to the desired waypoint. After the optimization was run, the trajectory got modified to the one shown in Figure 6.10. The performance

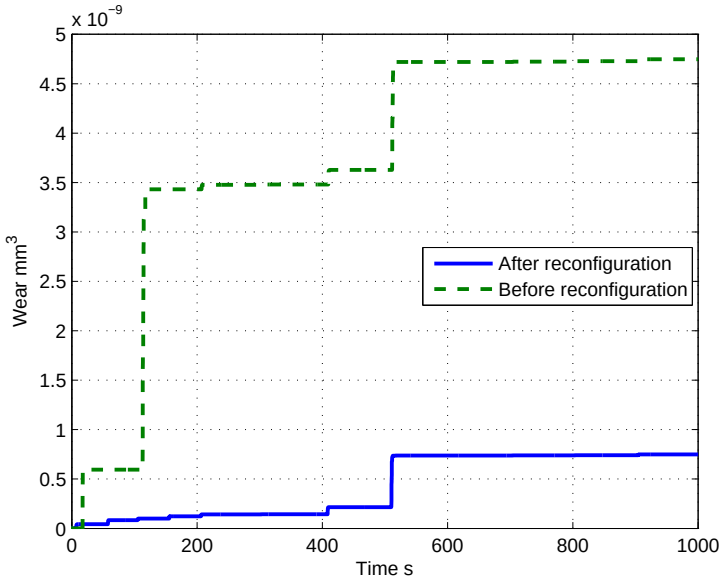
Table 6.1 Rise times for waypoints before and after optimization.

	Before optimization		After optimization	
Waypoint	Altitude (ft)	Rise time (s)	Altitude (ft)	Rise time (s)
$2'$			900	24.4
2	1800	19.5	1800	24.4
$3'$			1000	23.0
3	200	19.5	200	23.0
4	1000	19.5	1000	23.0
5	500	19.5	500	20.0
6	1500	19.5	1500	20.0
7	100	19.5	100	19.5
8	300	19.5	300	20.0
9	800	19.5	800	20.0
10	1200	19.5	1200	20.0
11	600	19.5	600	20.0

times (rise times) are modified as given in Table 6.1. The optimization generated two more waypoints listed as $2'$ and $3'$. The altitudes and rise times corresponding to the waypoints are also given in Table 6.1. Figure 6.11 shows the response before and after reconfiguration.



(a) Response of the aircraft before and after reconfiguration



(b) Wear response observed before and after reconfiguration

Figure 6.11. Waypoint tracking and wear response of the aircraft before and after reconfiguration.

6.2 Experimental Results and Discussion

Internal leakage due to wear was simulated on the experimental setup by connecting the two chambers of the cylinder using a directional proportional valve. The

leakage rate can be controlled electronically by giving voltage input to the valve. The voltage input to the valve has a range from -10 to 10 V with an accuracy of .01 V. Considering all these factors and the fact that wear is a slow process, a random polynomial wear function was used to allow observable and implementable degradation over a relatively short period of time as shown in Figure 6.12.

This function was used to generate a performance trade-off map similar to one shown in Figure 6.7. Figure 6.13 shows the map that was obtained using simulation model of the hydraulic actuator connected with the aircraft. Similar map was generated for the experimental hydraulic actuator as shown in Figure 6.14(a). The experimental map was generated by noting the level of degradation for different waypoint altitudes at different response speeds. The maps for increasing altitude are marginally different from decreasing altitude due to the presence gravitational forces. The trade-off for "going up" and "coming down" are shown in Figures 6.14(a) and 6.14(b) respectively. The map obtained from simulations (Figure 6.13) and that from experiments (Figure 6.14(a)) are very similar thus validating the simulation model again. The differences are due to unmodelled dynamics such as Coulomb friction etc. The direction of altitude change was also considered during the optimization process.

The optimization was not implemented on the real-time processor because the time step of the implementation is .001 s, where as the time taken by the optimization function depends on following factors:

1. The length of the mission under consideration
2. The tolerance on the constraints
3. The number of steps each waypoint altitude is allowed to be divided into
4. The threshold for altitude steps
5. The threshold for allowed degradation

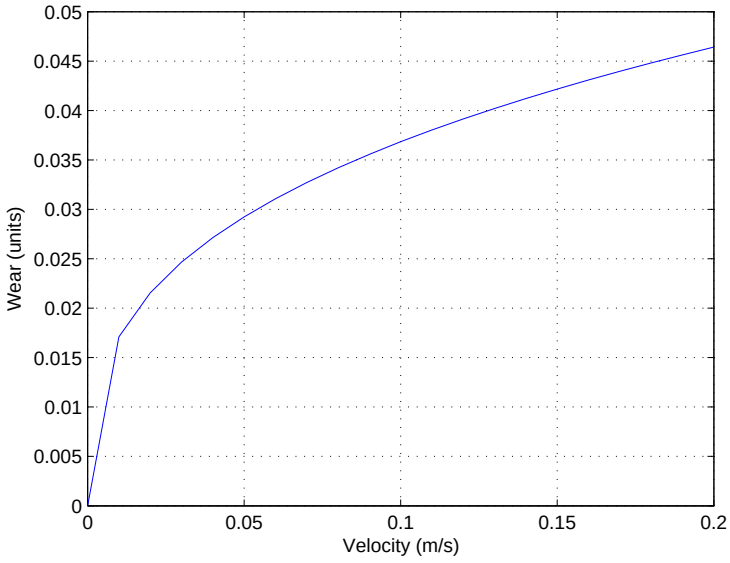


Figure 6.12. Random polynomial wear function implemented on the directional proportional valve controlling the internal leakage rate.

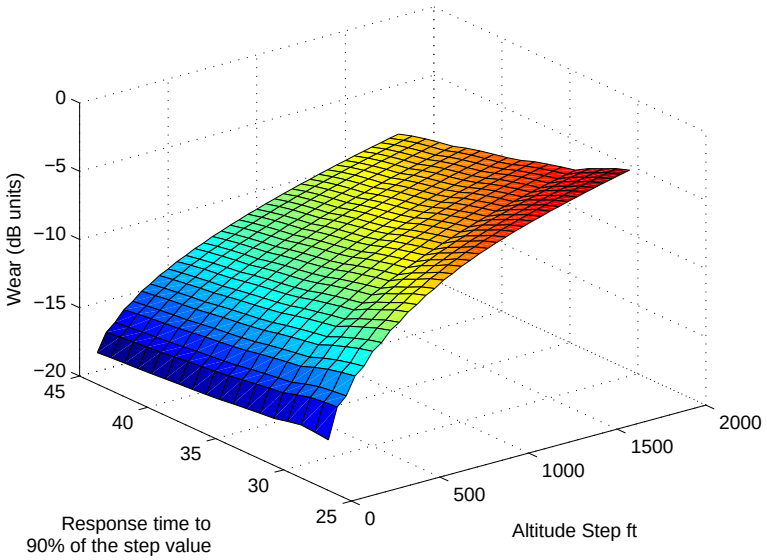
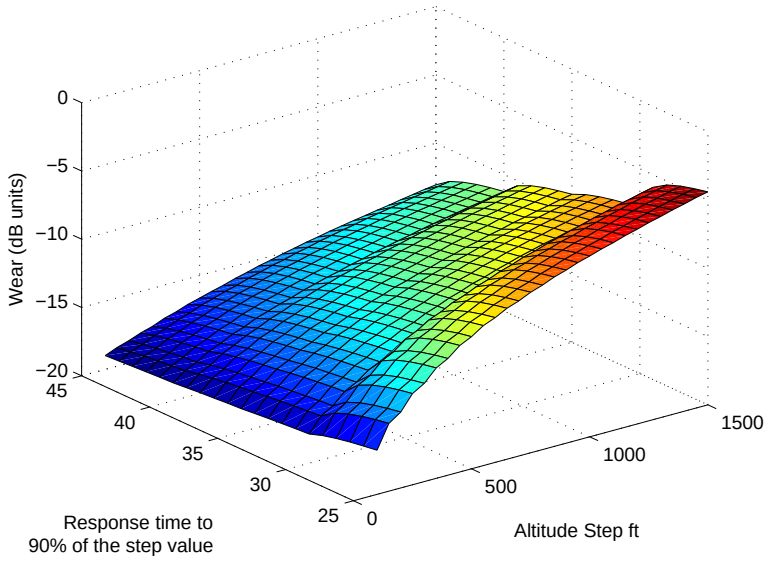
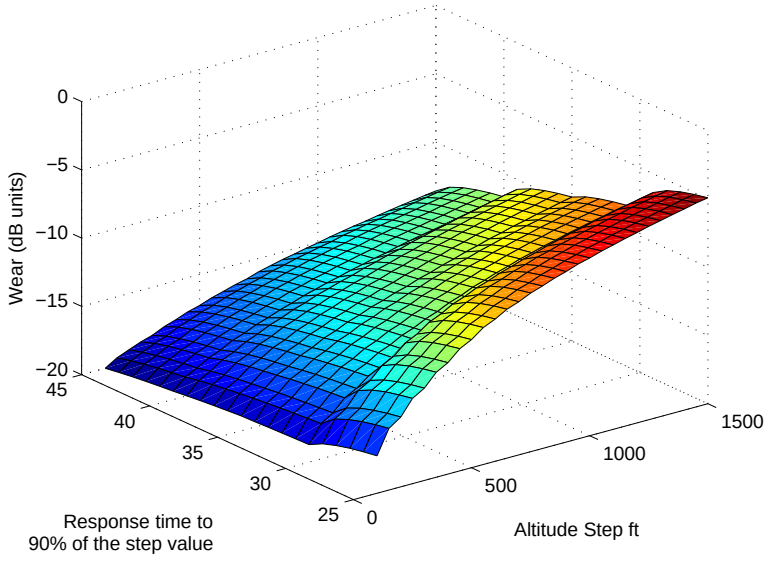


Figure 6.13. Map generated from simulation model of the hydraulic actuator representing trade-off between the degradation, the response speed, and mission profile.



(a) Increasing altitude



(b) Decreasing altitude

Figure 6.14. Map generated from simulation model of the hydraulic actuator representing trade-off between the degradation, the response speed, and mission profile.

Although the optimization is faster than real-time, the total time taken can be much longer than the simulation time step. To overcome this problem, the optimization was run on a separate computer. The implemented algorithm continuously searches for a trigger, which is generated by the prognosis module, to start the optimization. Once the optimization is complete, the information is updated on the real-time processor. Four different Case runs were performed to demonstrate the effectiveness of the optimization strategy:

(i) The first was the baseline case for a series of waypoint shown in Figure 6.9. No optimization was performed and hence it represents the worst case scenario with the best performance without any constraints on the degradation. Figure 6.15 shows the reference and the baseline response. The incremental degradation observed over this mission is 0.025 as seen in Figure 6.16.

(ii) In the second case, the degradation was upper bound by 0.002,

(iii) In the third case, the degradation was upper bound by 0.007, and

(iv) In the fourth case, the upper bound was set to 0.015

Case (i) is plotted in blue. Case (ii) with upper bound 0.002 is plotted in green, case (iii) with upper bound 0.007 is plotted in magenta, and case (iv) with upper bound 0.015 is shown in cyan. The reduction in degradation due to reconfiguration with different acceptable upper bounds is presented in Figure 6.16. It is evident from this figure, that the reconfiguration works well. It should however be noted that the degradation upper bound is a soft upper bound i.e. small violation of this upper bound constraint is allowed. This requirement stems from the fact that the optimization function has some discrete variables (e.g. the number of parts in which an altitude change is broken). Ordinarily, in continuous optimization, the optimum usually lies on the constraint boundary. Due to discrete nature of this problem, some optima may be located inside the feasible region as was observed with the case (ii),

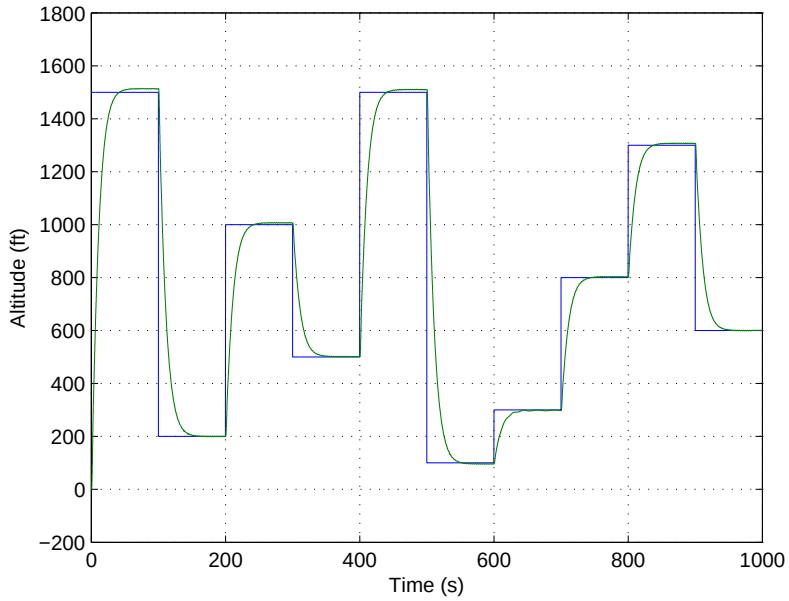


Figure 6.15. Reference altitude trajectory and response of the baseline Case (i).

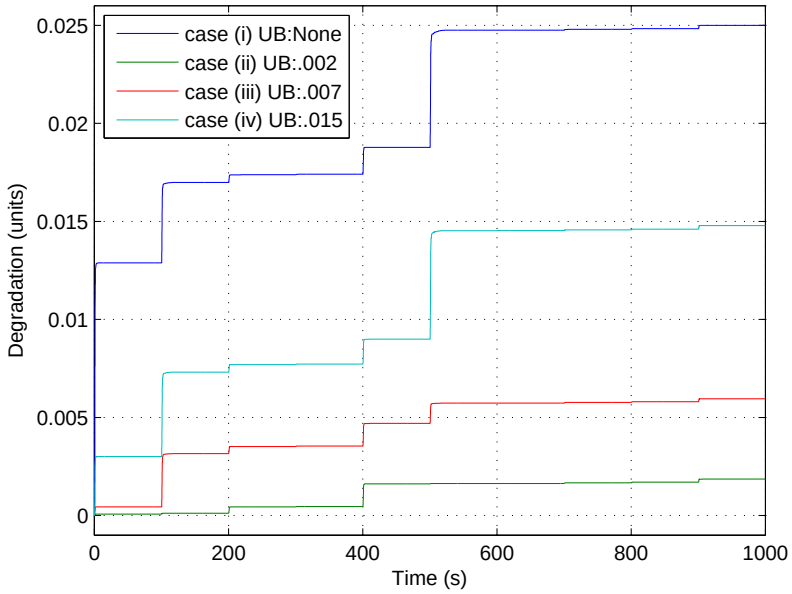
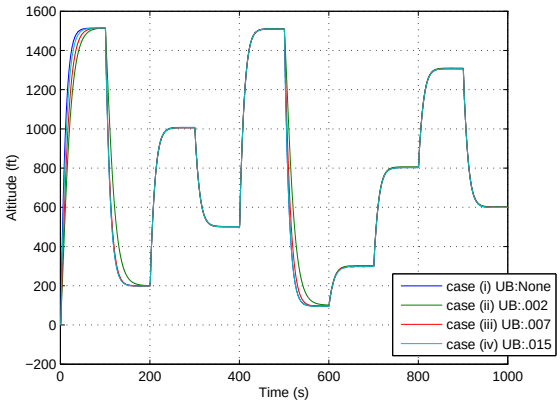
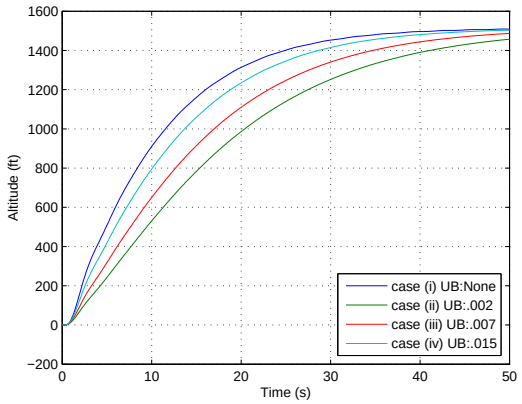


Figure 6.16. Comparison of degradation of the elevator actuator over the mission profile due to the four Cases.

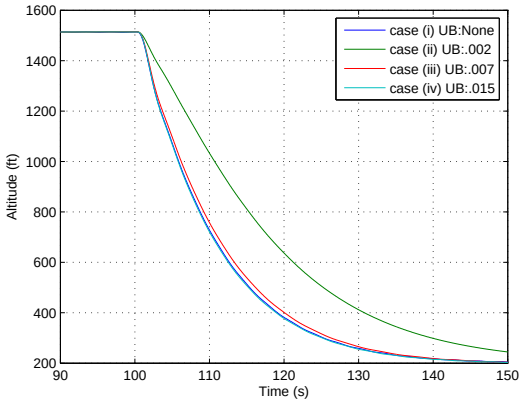
where the upper bound is 0.007, but the final degradation observed is close to 0.006 units.



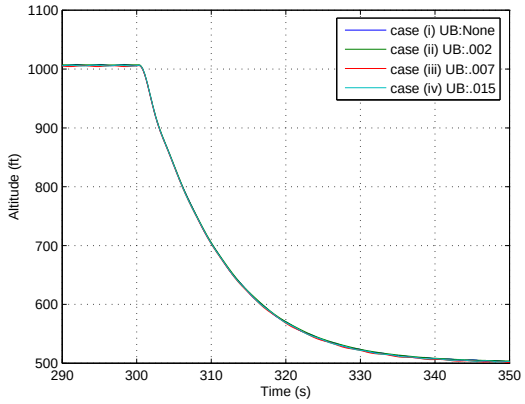
(a) Complete altitude trajectory



(b) Time 0 s to 50 s. All Cases have different response



(c) Time 90 s to 150 s. Cases (i) and (iv) are identical, (ii) and (iii) are different



(d) Time 290 s to 350 s. All Cases are identical

Figure 6.17. Comparison of altitude response of all four cases demonstrating differences and similarities at different locations in the time history.

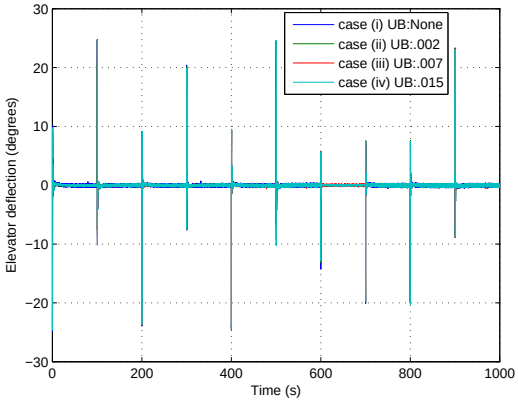
Figure 6.17 shows the comparison of altitude tracking response for the four cases. Three plots are presented which demonstrate the differences in response characteristics during certain time intervals and similarities during others. For example, between

0 s and 50 s all cases have different response characteristics as shown in Figure 6.17(b), whereas in Figure 6.17(d), the response is identical. Table 6.2 gives the response speeds demanded to satisfy the degradation upper bound constraint. As is evident from the table, only the response times for three largest altitude changes were affected. This is a consequence of the trade-off maps given in Figure 6.14(a) and 6.14(b).

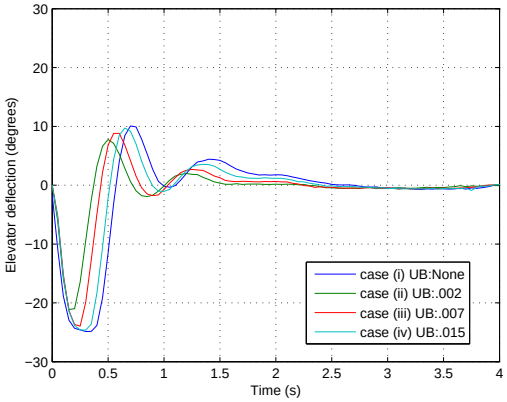
Table 6.2 Rise times for the four Cases.

		Upper bound			
		None	0.002	0.007	0.015
WP	Altitude (ft)	Rise time (s)	Rise time (s)	Rise time (s)	Rise time (s)
2	1500	25.00	34.99	32.13	27.52
3	200	25.00	31.00	27.75	25.00
4	1000	25.00	25.00	25.00	25.00
5	500	25.00	25.00	25.00	25.00
6	1500	25.00	25.00	25.00	25.00
7	100	25.00	33.01	29.31	25.00
8	300	25.00	25.00	25.00	25.00
9	800	25.00	25.00	25.00	25.00
10	1200	25.00	25.00	25.00	25.00
11	600	25.00	25.00	25.00	25.00

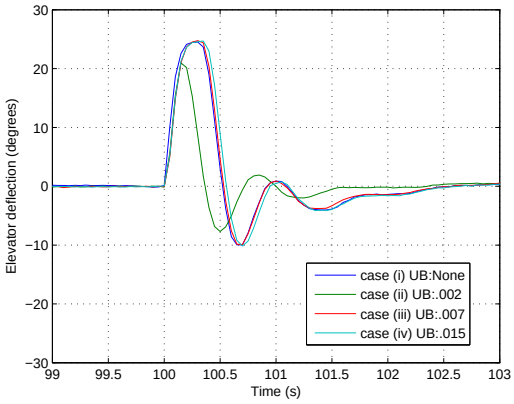
Similar behaviour is observed in the elevator deflection for the four cases as shown in Figure 6.18. The differences in elevator deflections in Figure 6.18(b) clearly identify the reason for the differences in the degradation for the four cases during the time interval 0 s to 4 s. In Figure 6.18(c) cases (i) and (iv) have same while cases (ii) and (iii) have different response for the interval 99 s to 103 s. All cases in Figure 6.18(d) have identical response. This is reflected in same incremental degradation in all four cases as seen from Figure 6.16.



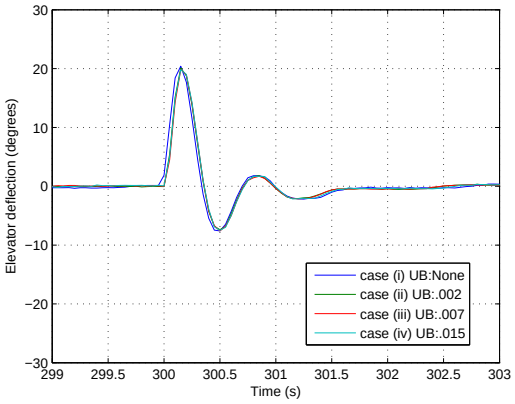
(a) Complete elevator tim history



(b) Time 0 s to 4 s. All Cases have different response



(c) Time 99 s to 103 s. Cases (i) and (iv) are identical, (ii) and (iii) are different



(d) Time 299 s to 303 s. All Cases are identical

Figure 6.18. Comparison of elevator deflections of all four Cases demonstrating differences and similarities at different locations in the time history.

The observed degradation of the elevator actuator was implemented on the leakage control valve. Figure 6.19 shows the leakage coefficient estimate obtained from the fault identification algorithm given in section 5.3, for the four cases. Due to largest degradation, the baseline Case has the largest magnitude for the estimate followed by case (iv), case (iii), and case (ii). The estimate for case (ii) is largely incorrect

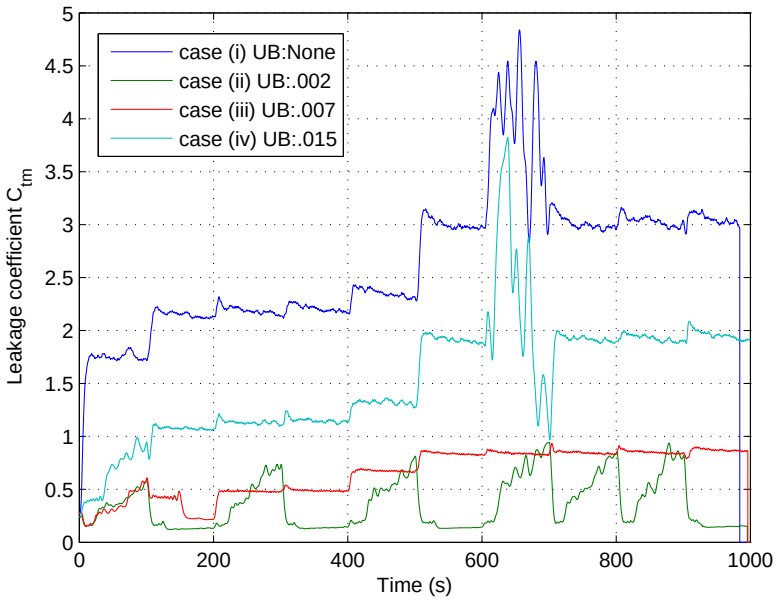
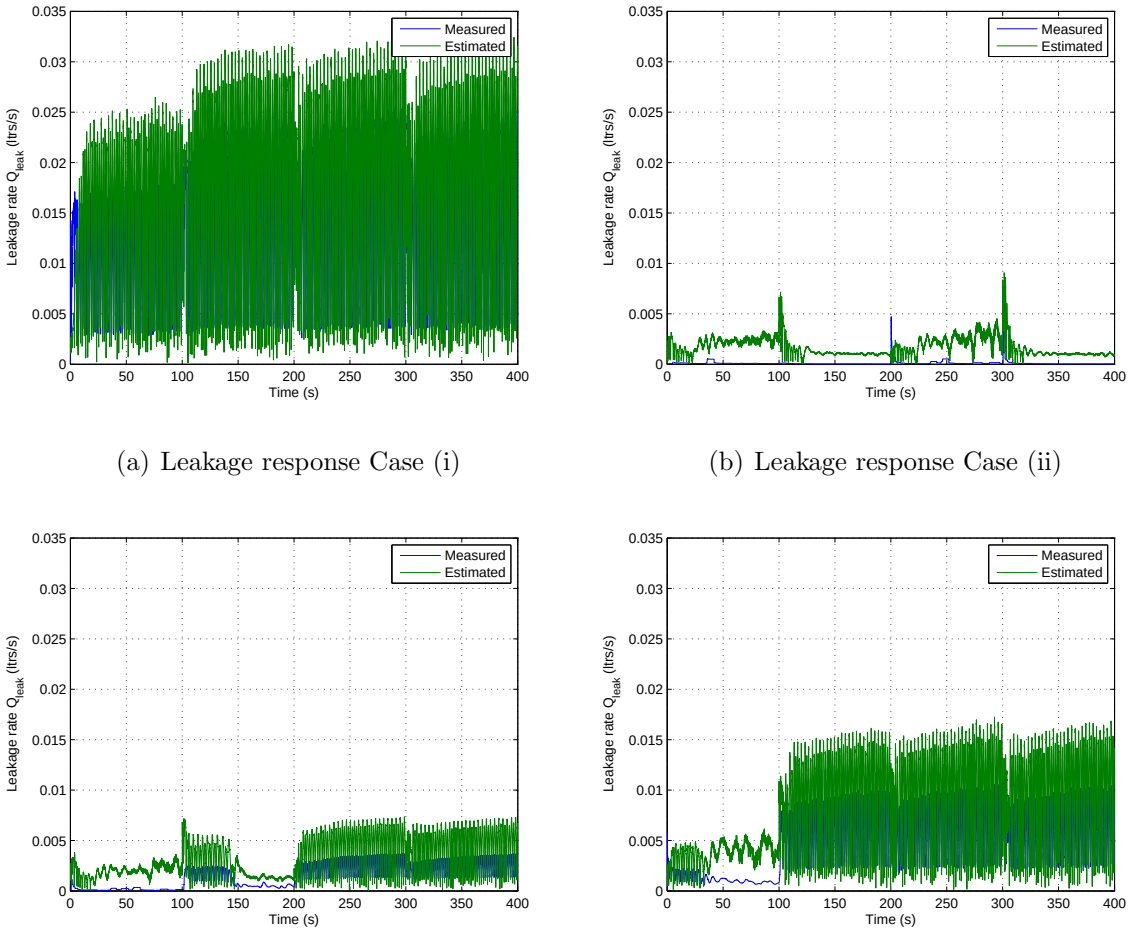


Figure 6.19. Comparison of estimation of the leakage coefficient C_{tm} for the the four Cases.

because the maximum degradation observed in case (ii) was less than the least count of the leakage control valve. The reason why the estimate is not close to zero is due to lack of persistent excitation condition whenever the aircraft reaches steady state, leading to a steady state elevator behaviour. It should however be noted that whenever there is an altitude change, the PE condition gets satisfied and the estimate immediately drops indicating zero or very small leakage. The effect of absence of PE condition can also be seen in the C_{tm} estimate for cases (i) and (iv), resulting in large random variations in the estimate. The situation is corrected as soon as there is an altitude change restoring the PE condition. One would expect to observe behaviour similar to case (ii) in other cases as well, but this is not the observed because as leakage increases, the turbulent flow causes vibrations in the system resulting in slight increase in "noise" in the hydraulic cylinder position sensor. If this noise is large enough, the PE condition get satisfied, resulting in good estimates. Figure 6.20 shows the measured and estimated leakages due to the degradation. As mentioned

earlier, in case (ii), there is negligible leakage, however Figure 6.20(b) shows a non zero leakage rate estimate because of incorrect C_{tm} estimate due to reasons mentioned earlier. The estimates for other cases are in good agreement with the measured rate and the error becomes even less wherever there is an altitude change, which increases information content of the signals improving the PE condition. This behaviour is observed in Figures 6.20(a), 6.20(c), and 6.20(d) at times 100 s, 200 s, etc.



(a) Leakage response Case (i)

(b) Leakage response Case (ii)

(c) Leakage response Case (iii)

(d) Leakage response Case (iv)

Figure 6.20. Comparison of measured and estimated leakage for the four Cases.

A prognosis based control strategy was developed in this chapter and its effectiveness demonstrated through HIL implementation, which successfully mitigated the

degradation at the end of a mission given performance and actuator constraints. The next chapter summarizes all the contribution and makes suggestions for future work.

7. CONTRIBUTIONS AND RECOMMENDATIONS

Prognosis based control reconfiguration is quickly becoming new paradigm in the systems health management. This is especially important in case of air-vehicles which have been operating beyond their design life. This technology not only improves safety but also increases the life of components and structures. For example, if a leakage fault in the hydraulic actuator in an aircraft due to wear is not "managed", it may lead to unacceptable degradation of performance leading to safety issues and eventually unresponsive actuator. The previous chapters demonstrated the ability to identify incipient faults in hydraulic actuators and a fault tolerant control strategy to minimize the effect of fault on the performance of the actuator. This ability is utilized in a supervisory control to modify the aircraft control structure and mission profile to minimize degradation while allowing performance. Specific contributions made by this work include:

1. Nonlinear models of following components were developed and integrated in a common simulation environment

   (a) Six degree of freedom aircraft model,

   (b) Double rod double acting hydraulic actuator,

   (c) Hydraulic piston seal wear, and

   (d) Aerodynamic forces acting on the aircraft control surfaces.

   These models were simplified to allow real-time implementation.

2. An autopilot was developed to allow waypoint navigation for the aircraft in the three dimensional space.

3. An experimental setup was designed for Hardware-in-the-loop simulation to allow verification and validation developed models and control strategies. The

setup was designed to allow introduction of different faults typically found in hydraulic systems such as internal/external leakage faults, supply pressure and flow rate faults, faults in the directional proportional valves, stuck actuators, etc.

4. Detailed system identification of the components of the experimental such as directional proportional valves, hydraulic parameters, and friction was performed using pseudo random multi-level signals.

5. A robust controller was developed for the hydraulic actuator. This controller was implemented on both the simulation model and the experimental setup. Good agreement between the experimental and simulation results demonstrated the validity of the model and the controller.

6. The fault considered in this application was the internal leakage fault which occurs due the seal wear. An approach based on derivative free nonlinear filtering was developed for detecting these internal leakage faults. The efficacy of this method was demonstrated by implementing it on an experimental testbed. The algorithm was able to estimate leakage rates as low as $.01 ltrs/s$ demonstrating the ability to identify incipient failures. The sensitivity of the algorithm to process noise covariance was demonstrated. Frictional forces acting on the hydraulic piston were also estimated using the same algorithm which exhibited the traditional friction model used in the literature.

7. The performance degradation of the position control due to the internal leakage fault was demonstrated.

8. An adaptive position control strategy for the hydraulic actuator with an internal leakage fault was proposed based on the robust controller and the stochastic fault identification algorithm. The effectiveness of the controller showcased by implementing it on the experimental setup. Stability of the adaptive control strategy and the fault identification algorithm was also proven.

9. A prognosis module was developed to predict degradation of the fault hydraulic actuator in an aircraft during a mission. An optimization based strategy was proposed to minimize the degradation over a mission while extracting best performance from the aircraft. The optimization was broken into an online component and an offline component to allow real-time implementation. The effectiveness of the strategy was demonstrated through experiments.

7.1 Recommendations

There are many potential areas for further research in this field.

1. Fault identification and fault tolerant control is a very mature field theoretically, however practical applications have been few. This is true especially when there are multiple faults in the system. Challenges exist in both passive and active fault tolerant techniques. As number of fault scenarios increase, the effectiveness of controller decreases. Maintaining stability while incorporating performance aspects is a wide open area.

2. Another major issue common to fault tolerant control techniques is the lack of systematic approaches to deal with actuator saturation. The has been some work on design and analysis of such control systems [143]. Some preliminary theoretical results on control of hydraulic actuator under input and state saturation were derived and are presented in Appendix C. These results can be extended in a manner similar to that presented in this dissertation to obtain adaptive fault tolerant control in presence of actuator saturation.

3. The fault considered in this dissertation is the internal leakage fault. The experimental setup has the capability to simulate various other faults such as external leakage, valve faults etc. The same fault identification algorithm can be used to identify these faults if they can be represented parametrically. The optimization strategy presented for prognosis based control can also be modified accordingly.

For example, for external leakage, the objective function can be appended with the oil loss function to minimize loss or allow mission completion.

4. Model simplification was performed to allow real-time implementation. Developing prognosis based control for full nonlinear model will require algorithm changes to allow faster simulations and real-time implementation.

5. The fault identification strategy used is stochastic in nature and thus takes into account various uncertainties such as sensor noise, modeling errors, etc., but the prognosis module developed does not take into account these uncertainties. Furthermore, prognosis itself has large uncertainty associated with it which needs to be accounted for.

A. EQUATIONS OF MOTION FOR A 6 DOF AIRCRAFT

This appendix list all the equations required for modeling and simulating a F-16 aircraft. All the details have been taken from [144].

1. the airframe is assumed to be a rigid body in the motion under consideration,
2. the airplane's mass is assumed to be constant during the time interval in which its motions are studied,
3. the earth is assumed to be fixed in space, i.e. its rotation is neglected,
4. the curvature of the earth is neglected.
5. the atmosphere is assumed to be steady, constant velocity wind

A.1 Translation motion

Translational equations of motion are given as:

$$\dot{V}_b = F - \omega \times V_b \qquad \text{where, } F = F_a + F_t + F_{gr} + F_w \tag{A.1}$$

where V_b are the body axis components of the center of mass velocity with respect to an inertial frame. F_a, F_t, F_{gr}, F_w are the forces due to aerodynamics, thrust, gravity and wind respectively and $\omega = \begin{pmatrix} p & q & r \end{pmatrix}^T$ is the body axis angular velocity.

A.2 Rotational motion

The moment equation can be written as:

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = I^{-1} \left\{ -\omega \times [I\omega + H_{eng}] + \begin{pmatrix} L \\ M \\ N \end{pmatrix} \right\} \tag{A.2}$$

where L, M, and N are the moments acting about x, y, and z axes respectively and I is the inertia matrix and is given by:

$$I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{A.3}$$

and H_{eng} is the angular moment of the engine:

$$H_{eng} = \begin{pmatrix} h_{eng} \\ 0 \\ 0 \end{pmatrix} \tag{A.4}$$

A.3 Kinematics

The kinematic equations can be written as follows:

$$\dot{\phi} = p + \tan\theta(q\sin\phi + r\cos\phi) \tag{A.5a}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{A.5b}$$

$$\dot{\psi} = (q\sin\phi + r\cos\phi)/\cos\theta \tag{A.5c}$$

where, ϕ is the roll angle, θ is the pitch angle and ψ is the yaw angle. They are also known as attitude angles. p, q and r are respective rates. These equations can be writing these in quarternion form to avoid singularities as follows:

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{pmatrix} = \begin{bmatrix} 0 & -r & -q & -p \\ r & 0 & -p & q \\ q & p & 0 & -r \\ p & -q & r & 0 \end{bmatrix} \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} \tag{A.6}$$

where,

$$q_1 = \cos\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} + \sin\frac{\psi}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2} \tag{A.7a}$$

$$q_2 = \sin\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} - \cos\frac{\psi}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2} \tag{A.7b}$$

$$q_1 = \cos\frac{\psi}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2} + \sin\frac{\psi}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2} \tag{A.7c}$$

$$q_1 = \cos\frac{\psi}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2} - \sin\frac{\psi}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2} \tag{A.7d}$$

A.4 Inertial location

Finally the inertial location of the aircraft can be calculated from the velocity components and attitude angles as follows:

$$\dot{x}_b = \{u_b\cos\theta + (v_b\sin\phi + w_b\cos\phi)\sin\theta\}\cos\psi - (v_b\cos\phi - w_b\sin\phi)\sin\psi \tag{A.8}$$

$$\dot{y}_b = \{u_b\cos\theta + (v_b\sin\phi + w_b\cos\phi)\sin\theta\}\sin\psi + (v_b\cos\phi - w_b\sin\phi)\cos\psi \tag{A.9}$$

$$\dot{z}_b = -u_b\sin\theta + (v_b\sin\phi + w_b\cos\phi)\cos\theta \tag{A.10}$$

A.5 Transformation

It is more convenient to write equations (A.1) in terms of the relative velocity, the angle of attack α and the side slip angle β, because these quantities can be readily measured during flight:

$$V_b = V_r + V_w \tag{A.11}$$

V_w is the wind velocity and V_r is the relative velocity of the aircraft with respect to the wind.

$$u_r = V_r\cos\alpha\cos\beta \tag{A.12a}$$

$$v_r = V_r\sin\beta \tag{A.12b}$$

$$w_r = V_r\sin\alpha\cos\beta \tag{A.12c}$$

thus,

$$\begin{aligned} V_r &= \sqrt{u_r^2 + v_r^2 + w_r^2} \\ \alpha &= \arctan\left(\frac{w_r}{u_r}\right) \\ \beta &= \arctan\left(\frac{v_r}{\sqrt{u_r^2 + w_r^2}}\right) \end{aligned} \tag{A.13}$$

differentiating,

$$\begin{aligned} \dot{V}_r &= \frac{u_r\dot{u}_r + v_r\dot{v}_r + w_r\dot{w}_r}{V_r} \\ \Rightarrow \dot{V}_r &= \dot{u}_r\cos\alpha\cos\beta + \dot{v}_r\sin\beta + \dot{w}_r\sin\alpha\cos\beta \qquad \text{from equation (A.12)} \end{aligned} \tag{A.14}$$

using equations (A.1), (A.14) and (A.11) and grouping the terms due to the the derivative of wind velocity in F_w, we get

$$\begin{aligned} \dot{V}_r = \frac{1}{m}\Big[&(X_{gr} + X_t + X_a + X_w)\cos\alpha\cos\beta + (Y_{gr} + Y_t + Y_a + Y_w)\sin\beta \\ &+(Z_{gr} + Z_t + Z_a + Z_w)\sin\alpha\cos\beta\Big] \end{aligned} \tag{A.15}$$

for angle of attack we have,

$$\begin{aligned} \dot{\alpha} =& \frac{u_r\dot{w}_r - \dot{u}_r w_r}{u_r^2 + w_r^2} \\ \Rightarrow \dot{\alpha} =& \frac{\dot{w}_r\cos\alpha - \dot{u}_r\sin\alpha}{V_r\cos\beta} \\ \Rightarrow \dot{\alpha} =& \frac{1}{V_r\cos\beta}\left\{\frac{1}{m}\Big[(Z_{gr} + Z_t + Z_a + Z_w)\cos\alpha - (X_{gr} + X_t + X_a + X_w)\sin\alpha\Big]\right\} \\ &+ q - (p\cos\alpha + r\sin\alpha)\tan\beta \end{aligned} \tag{A.16}$$

similarly for the side slip angle,

$$\begin{aligned} \dot{\beta} =& \frac{\dot{v}_r(u_r^2 + w_r^2) - v_r(u_r\dot{u}_r + w_r\dot{w}_r}{V_r^2\sqrt{u_r^2 + w_r^2}} \\ \dot{\beta} =& \frac{1}{V_r}\left\{\frac{1}{m}\Big[-(X_{gr} + X_t + X_a + X_w)\cos\alpha\sin\beta + (Y_{gr} + Y_t + Y_a + Y_w)\cos\beta \right. \\ &\left. - (Z_{gr} + Z_t + Z_a + Z_w)\sin\alpha\sin\beta\Big] - p\sin\alpha - r\cos\alpha \right. \end{aligned} \tag{A.17}$$

The forces due to gravity are:

$$X_{gr} = -mg\sin\theta \tag{A.18}$$

$$Y_{gr} = mg\cos\theta\sin\psi \tag{A.19}$$

$$Z_{gr} = mg\cos\theta\cos\psi \tag{A.20}$$

The aerodynamic forces $F_a = \begin{pmatrix} X_a & Y_a & Z_a \end{pmatrix}^T$ are given by following expression:

$$X_a = \bar{q}SC_{x,t} \tag{A.21}$$

$$Y_a = \bar{q}SC_{y,t} \tag{A.22}$$

$$Y_a = \bar{q}SC_{z,t} \tag{A.23}$$

and the moments are given as:

$$L = \bar{q}SbC_{l,t} \tag{A.24}$$

$$M = \bar{q}SbC_{m,t} \tag{A.25}$$

$$N = \bar{q}SbC_{n,t} \tag{A.26}$$

$C_{x,t}, C_{y,t}, C_{z,t}, C_{l,t}, C_{m,t}$, and $C_{n,t}$ are the aerodynamic coefficients. These coefficients are function of angle of attack, side slip angle, deflection of control surfaces, Mach number and the dynamic pressure. The aircraft specific stability derivatives are given as follows [144]:

$$\begin{aligned} C_{x,t} =& C_X(\alpha,\beta,\delta_h) + \Delta C_{X,lef}\left(1-\frac{\delta_{lef}}{25}\right) + \Delta C_{X,sb}(\alpha)\left(\frac{\delta_{sb}}{60}\right) \\ &+ \frac{\bar{c}q}{2V}\left[C_{X_q}(\alpha) + \Delta C_{X_{q,lef}}(\alpha)\left(1-\frac{\delta_{lef}}{25}\right)\right] \end{aligned} \tag{A.27a}$$

$$\begin{aligned} C_{z,t} =& C_Z(\alpha,\beta,\delta_h) + \Delta C_{Z,lef}\left(1-\frac{\delta_{lef}}{25}\right) + \Delta_{Z,sb}(\alpha)\left(\frac{\delta_{sb}}{60}\right) \\ &+ \frac{\bar{c}q}{2V}\left[C_{Z_q}(\alpha) + \Delta C_{Z_{q,lef}}(\alpha)\left(1-\frac{\delta_{lef}}{25}\right)\right] \end{aligned} \tag{A.27b}$$

$$
\begin{aligned}
C_{m,t} =& C_m(\alpha,\beta,\delta_h)\eta_{\delta_h}\delta_h + C_{z,t}(x_{\text{cg,ref}} - x_{\text{cg}}) + \Delta C_{m,lef}\left(1 = \frac{\delta_{lef}}{25}\right) \\
&+ \Delta C_{m,sb}(\alpha)\left(\delta_{sb}\right) + \frac{\bar{c}q}{2V}\left[C_{m_q}(\alpha) + \Delta C_{m_{q,lef}}(\alpha)\left(1 - \frac{\delta_{lef}}{25}\right)\right] \\
&+ \Delta C_m(\alpha) + \Delta C_{m,ds}(\alpha,\delta_h) \qquad \text{(A.27c)} \\
C_{y,t} =& C_Y(\alpha,\beta) + \Delta C_{Y,lef}\left(1 - \frac{\delta_{lef}}{25}\right) \\
&+ \left[\Delta C_{Y,\delta_{a=20^\circ}} + \Delta C_{Y,\delta_{a=20^\circ},lef}\left(1 - \frac{\delta_{lef}}{25}\right)\right]\left(\frac{\delta_a}{20}\right) \\
&+ \Delta C_{Y,\delta_{r=30^\circ}}\left(\frac{\delta_r}{30}\right) + \frac{b}{2V}\left\{\left[C_{Y_r}(\alpha) + \Delta C_{Y_{r,lef}}(\alpha)\left(1 - \frac{\delta_{lef}}{25}\right)\right] r\right. \\
&\left.+ \left[C_{Y_p}(\alpha) + \Delta C_{Y_{p,lef}}(\alpha)\left(1 - \frac{\delta_{lef}}{25}\right)\right] p\right\} \qquad \text{(A.27d)} \\
C_{n,t} =& C_n(\alpha,\beta,\delta_h) + \Delta C_{n,lef}\left(1 - \frac{\delta_{lef}}{25}\right) - C_{y,t}(x_{\text{cg,ref}} - x_{\text{cg}})\frac{\bar{c}}{b} \\
&+ \left[\Delta C_{n,\delta_{a=20^\circ}} + \Delta C_{n,\delta_{a=20^\circ},lef}\left(1 - \frac{\delta_{lef}}{25}\right)\right]\left(\frac{\delta_a}{20}\right) \\
&+ \Delta C_{n,\delta_{r=30^\circ}}\left(\frac{\delta_r}{30}\right) + \frac{b}{2V}\left\{\left[C_{n_r}(\alpha) + \Delta C_{n_{r,lef}}(\alpha)\left(1 - \frac{\delta_{lef}}{25}\right)\right] r\right. \\
&\left.+ \left[C_{n_p}(\alpha) + \Delta C_{n_{p,lef}}(\alpha)\left(1 - \frac{\delta_{lef}}{25}\right)\right] p\right\} + \Delta C_{n_\beta}(\alpha)\beta \qquad \text{(A.27e)} \\
C_{l,t} =& C_l(\alpha,\beta,\delta_h) + \Delta C_{l,lef}\left(1 - \frac{\delta_{lef}}{25}\right) \\
&+ \left[\Delta C_{l,\delta_{a=20^\circ}} + \Delta C_{l,\delta_{a=20^\circ},lef}\left(1 - \frac{\delta_{lef}}{25}\right)\right]\left(\frac{\delta_a}{20}\right) \\
&+ \Delta C_{l,\delta_{r=30^\circ}}\left(\frac{\delta_r}{30}\right) + \frac{b}{2V}\left\{\left[C_{l_r}(\alpha) + \Delta C_{l_{r,lef}}(\alpha)\left(1 - \frac{\delta_{lef}}{25}\right)\right] r\right. \\
&\left.+ \left[C_{l_p}(\alpha) + \Delta C_{l_{p,lef}}(\alpha)\left(1 - \frac{\delta_{lef}}{25}\right)\right] p\right\} + \Delta C_{l_\beta}(\alpha)\beta \qquad \text{(A.27f)}
\end{aligned}
$$

The leading edge flap of the aircraft cannot be controlled by the pilot and its displacement is scheduled as follows

$$\delta_{lef} = 1.38\frac{25 + 7.25}{s + 7.25}\alpha - 9.05\frac{\bar{q}}{P_s} + 1.45 \qquad \text{(A.28)}$$

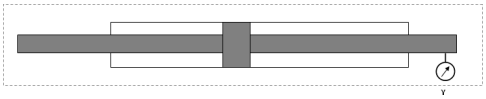
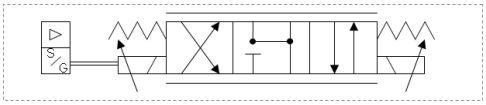
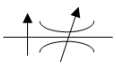
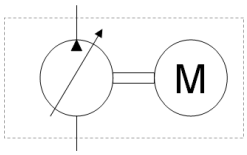
Thus, the state variables for the aircraft system are:

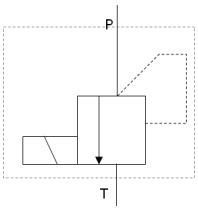
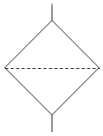
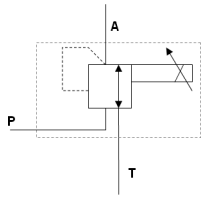
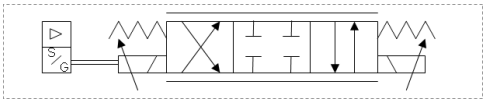
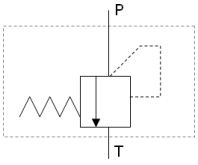
$$x = \begin{pmatrix} V_r \\ \alpha \\ \beta \\ p \\ q \\ r \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ x_b \\ y_b \\ z_b \end{pmatrix} \tag{A.29}$$

and the inputs can be written as

$$u = \begin{pmatrix} \delta_t & \delta_{lef} & \delta_e & \delta_a & \delta_r \end{pmatrix}^T \tag{A.30}$$

B. GLOSSARY OF HYDRAULIC SYMBOLS

Symbols	Explanation
	Hydraulic actuator with a linear position sensor
	Direct operated directional proportional valve with electrical spool position feedback and integrated electronics (example: Parker D1FH or D1FM series proportional valves with spool type E50/E53). This valve has a position in which both ports can drain to the tank and supply port is blocked
	Flowmeter
	Pressure compensated flow control valve
	Hydrailic punp with electric motor

Symbols	Explanation
	Solenoid operated proportional rpressure relief valve
	Pressure transducer
	Termperature sensor
	Hydraulic filter
	Solenoid operated proportional pressure reducing valve
	Direct operated directional proportional valve with electrical spool position feedback and integrated electronics (example: Parker D1FX Series). This valve has a position in which both ports can drain to the tank and supply port is blocked
	Simple spring operated pressure relief valves

C. COMMAND FILTERED BACKSTEPPING

C.1 Command Filtered Backstepping for Hydraulic Actuators

System equations

$$m\ddot{x}_L = P_L A - b\dot{x}_L - F_{fc} + \tilde{f} \tag{C.1}$$

$$\frac{V_t}{4\beta_e}\dot{P}_L = -A\dot{x}_L - C_{tm}P_L + Q_L \tag{C.2}$$

$$\tau_v \dot{x}_v = -x_v + k_v i \tag{C.3}$$

Define

$$\bar{P}_L = \frac{P_L}{S_{c_3}} \qquad \bar{x}_v = \frac{x_v}{S_{c_4}}$$

$$\theta_1 = \frac{AS_{c_3}}{m} \qquad \theta_2 = d_n$$

$$\theta_3 = \frac{4\beta_e}{V_t}\frac{S_{c_4}}{\sqrt{\rho}}\frac{C_d w}{\sqrt{S_{c_3}}}$$

System dynamics can now be written as

$$\dot{x}_1 = x_2 \tag{C.4}$$

$$\dot{x}_2 = \theta_1\left(x_3 - \bar{b}x_2 - \bar{F}_{fc}\right) + \theta_2 + \tilde{d} \tag{C.5}$$

$$\dot{x}_3 = \theta_3\left(-\bar{A}x_2 - \bar{C}_{tm}x_3 + g_3 x_4\right) \tag{C.6}$$

$$\dot{x}_4 = -\frac{1}{\tau_v}x_v + \frac{\bar{k}_v}{\tau_v}u \tag{C.7}$$

C.1.1 Step 1

Define $x_{1_c} = x_d$ and $\dot{x}_{c_1} = \dot{x}_d$, where x_d is the desired trajectory. The tracking error is then given by

$$\tilde{z}_1 = x_1 - x_{1_c} \tag{C.8}$$

Next define compensated tracking error due to command filtered signal as

$$v_1 = \tilde{z}_1 - \zeta_1 \tag{C.9}$$

where,

$$\dot{\zeta}_1 = -k_1\zeta_1 + (x_{2_c} - \alpha_1) + \zeta_2 \tag{C.10}$$

Command filtering is used to avoid computing the derivatives of virtual inputs during the recursive backstepping procedure

$$\dot{z}_{11} = \omega_n z_{12} \qquad x_{c_2} = z_{11} \tag{C.11}$$

$$\dot{z}_{12} = -2\xi\omega_n z_{12} - \omega_n (z_{11} - \alpha_1) \qquad \dot{x}_{c_2} = z_{21} \tag{C.12}$$

The tracking error dynamics is given by

$$\dot{\tilde{z}}_1 = \dot{x}_1 - \dot{x}_{1_c} \tag{C.13}$$

$$= x_2 - \dot{x}_{1_c} \tag{C.14}$$

$$= \alpha_1 - \dot{x}_{1_c} + (x_{2_c} - \alpha_1) + (x_2 - x_{2_c}) \tag{C.15}$$

$$= \alpha_1 - \dot{x}_{1_c} + (x_{2_c} - \alpha_1) + \tilde{z}_2 \tag{C.16}$$

$$\tag{C.17}$$

Define the first virtual input as

$$\alpha_1 = \dot{x}_{1_c} - k_1\tilde{z}_1 \tag{C.18}$$

Thus, the compensated tracking error dynamics can be written as

$$\dot{v}_1 = \dot{\tilde{z}}_1 - \dot{\zeta}_1 \tag{C.19}$$

$$= \alpha_1 - \dot{x}_{1_c} + (x_{2_c} - \alpha_1) + \tilde{z}_2 - (-k_1\zeta_1 + (x_{2_c} - \alpha_1) + \zeta_2) \quad = -k_1\tilde{z}_1 + \tilde{z}_2 + k_1\zeta_1 + \zeta_2 \tag{C.20}$$

$$= -k_1 v_1 + v_2 \tag{C.21}$$

C.1.2 Step 2

$$\tilde{z}_2 = x_2 - x_{c_2} \tag{C.22}$$

$$v_2 = \tilde{z}_2 - \zeta_2 \tag{C.23}$$

$$\zeta_2 = -k_1\zeta_2 + \theta_1\left(x_{3_c} - \alpha_2\right) + \theta_1\zeta_3 \tag{C.24}$$

with command filtering

$$\dot{z}_{21} = \omega_n z_{22} \qquad x_{c_3} = z_{21} \tag{C.25}$$

$$\dot{z}_{22} = -2\xi\omega_n z_{22} - \omega_n\left(z_{21} - \alpha_1\right) \qquad \dot{x}_{3_c} = z_{22} \tag{C.26}$$

Define the second virtual input as

$$\alpha_2 = \frac{1}{\theta_1}\left(-k_2\tilde{z}_2 + \dot{x}_{2_c} - \left(-\bar{b}\theta_1 x_2 - \theta_1\bar{F}_{fc} + \theta_2 + \tilde{d}\right) - v_1\right) \tag{C.27}$$

This gives the the compensated error dynamics as

$$\dot{v}_2 = \dot{\tilde{z}}_2 - \dot{\zeta}_2 \tag{C.28}$$

$$= \theta_1\left(x_3 - \bar{b}x_2 - \bar{F}_{fc}\right) + \theta_2 + \tilde{d} - \dot{x}_{2_c} - \left(-k_2\zeta_2 + \theta_1\left(x_{3_c} - \alpha_2\right) + \zeta_3\right) \tag{C.29}$$

$$= -k_2\tilde{z}_2 + \theta_1\left(x_{3_c} - \alpha_2\right) + \theta_1\tilde{z}_3 - v_1 - \left(-k_2\zeta_2 + \theta_1\left(x_{3_c} - \alpha_2\right) + \zeta_3\right) \tag{C.30}$$

$$= -k_2 v_2 - v_1 + \theta_1 v_3 \tag{C.31}$$

C.1.3 Step 3

$$\tilde{z}_3 = x_3 - x_{c_3} \tag{C.32}$$

$$v_3 = \tilde{z}_3 \tag{C.33}$$

The final input is given by

$$\alpha_3 = \frac{1}{\theta_3 g_3}\left(-k_3\tilde{z}_3 + \dot{x}_{3_c} - \theta_3\left(-\bar{A}x_2 - \bar{C}_{tm}x_3\right) - \theta_1 v_1\right) \tag{C.34}$$

The compensated error dynamics is given by

$$\dot{v}_3 = \dot{\tilde{z}}_3 \tag{C.35}$$

$$= \theta_3 \left(-\bar{A}x_2 - \bar{C}_{tm}x_3 + g_3\alpha_3\right) - \dot{x}_{3_c} \tag{C.36}$$

$$= -k_3\tilde{z}_3 - \theta_1 v_2 \tag{C.37}$$

$$= -k_3 v_3 - \theta_1 v_2 \tag{C.38}$$

Consider a Lyapunov function candidate given below

$$V = \frac{1}{2}\sum_{i=1}^{3} v_i^2 \tag{C.39}$$

Then,

$$\dot{V} = \sum_{i=1}^{3} v_i \dot{v}_i \tag{C.40}$$

$$= \sum_{i=1}^{3} -k_i v_i^2 \tag{C.41}$$

$$\leq -2k_v V \qquad\qquad k_v = \min\left(k_1, k_2, k_3\right) \tag{C.42}$$

where K is a positive definite diagonal matrix. Hence the equilibrium $v = 0$ is exponentially stable. Furthermore, the state $\tilde{z}_3$ converges exponentially to 0 because $\tilde{z}_3 = v_3$. It is also easy to show that $v_i \in \mathcal{L}_2$ using integration. The properties of $\tilde{z}_i, z_i$, can be proved using the two time scale property of the considered system and the command filtered dynamics and applying the Tikhinov's Theorem.

C.2 Adaptive Robust Constrained Control using Backstepping

Consider the following system of equations to represent a general plant in strict feed-forward form

$$\dot{x}_i = x_{i+1} + \theta^T \phi_I(x_i, x_2, \ldots, x_i) + \Delta_i \tag{C.43}$$

$$\dot{x}_n = \sigma(x)u + \theta^T \phi_n(x, t) + \Delta_n \tag{C.44}$$

$$y = x_1 \tag{C.45}$$

we will design an adaptive robust control while explicitly considering actuator saturation and rate limits. For this we will utilize previous result on command filtered backstepping. Assume all disturbances and parameters are bounded above.

C.2.1 Step 1

$$\dot{x}_1 = x_2 + \theta^T \phi_1 + \Delta_1 \tag{C.46}$$

Define tracking error as

$$z_1 = x_1 - x_{1d} \tag{C.47}$$

The modified tracking error due to saturation is given by

$$\bar{z}_1 = z_1 - \xi_1 \tag{C.48}$$

where ξ_i are the filtered versions of the effect of state constraints on the tracking error z_i.

$$\dot{\xi}_i = -k_i \xi_i + (\alpha_i - \alpha_i^0) \tag{C.49}$$

The nominal virtual control inputs are represented as α_i^0 and are filtered to produce magnitude and rate limited virtual control signals α_i that are within prescribed limits. The command filter can be chosen for instance as

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \end{pmatrix} = \begin{pmatrix} q_2 \\ 2\zeta\omega_n \left(S_r \left(\frac{\omega_n}{2\zeta\omega_n} \left[S_m(\alpha_i^0) - q_1 \right] \right) - q_2 \right) \end{pmatrix} \tag{C.50}$$

$$\begin{pmatrix} \alpha_i \\ \dot{\alpha}_i \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} \tag{C.51}$$

where S_r and S_m are magnitude and rate limit functions for e.g.

$$S_m(x) = \begin{cases} M & \text{if } x \geq M, \\ x & \text{if } |x| < 0. \\ -M & \text{if } x \leq -M. \end{cases} \tag{C.52}$$

The effect of implementing achievable virtual control signals instead of the desired ones is estimates by the ξ_i filters. These filter help in defining the modified tracking errors. Choose the first virtual control input as follows:

$$\alpha_1^0 = \alpha_{1f} + \alpha_{1s} + \alpha_{1c} \tag{C.53}$$

where, α_{1f} is the model compensation, α_{1s} is the robust feedback term and α_{1c} is the compensation due to saturation effects. These three are given as follows

$$\alpha_{1f} = \dot{x}_{1d} - \hat{\theta}^T \phi_1 \tag{C.54}$$

$$\alpha_{1s} = \alpha_{1s_1} + \alpha_{1s_2} \qquad \alpha_{1s_1} = -k_{1s} z_1 \tag{C.55}$$

$$\alpha_{1c} = -\xi_1 \tag{C.56}$$

α_{1s_2} is function satisfying conditions

i. $$\bar{z}_1(-\tilde{\theta}^T \phi_1 + \Delta_1 + \alpha_{1s_2}) \leq \epsilon_1 \qquad \forall\ \hat{\theta} \in \Omega_\theta \tag{C.57}$$

ii. $$\bar{z}_1 \alpha_{1s_2} \leq 0 \tag{C.58}$$

with this, the modified tracking error dynamics can be written as

$$\dot{\bar{z}}_1 = \dot{z}_1 - \dot{\xi}_1 \tag{C.59}$$

$$= \dot{x}_1 - \dot{x}_{1d} - \left(-k_1 \xi_1 + (\alpha_1 - \alpha_1^0)\right) \tag{C.60}$$

$$= x_2 + \theta^T \phi_1 + \Delta_1 + \alpha_{1s_2} - \dot{x}_{1d} - \left(-k_1 \xi_1 + (\alpha_1 - \alpha_1^0)\right) \tag{C.61}$$

$$= z_2 - \xi_2 - k_1 \bar{z}_1 - \tilde{\theta}^T \phi_1 + \Delta_1 + \alpha_{1s_2} \tag{C.62}$$

$$= \bar{z}_2 - k_1 \bar{z}_1 - \tilde{\theta}^T \phi_1 + \Delta_1 + \alpha_{1s_2} \tag{C.63}$$

C.2.2 Step i

At the i^{th} step, we have the tracking and modified tracking errors are

$$z_i = x_i - \alpha_i \tag{C.64}$$

$$\bar{z}_i = z_i - \xi_i \tag{C.65}$$

The ξ_i, which are the filtered versions of effects of state constraints are

$$\dot{\xi}_i = -k_i\xi_i + (\alpha_i - \alpha_i^0) \tag{C.66}$$

The virtual control input is defined as

$$\alpha_i^0 = \alpha_{if} + \alpha_{is} + \alpha_{ic} \tag{C.67}$$

The components are given as

$$\alpha_{if} = \dot{\alpha}_{i-1} - \hat{\theta}^T\phi_i \tag{C.68}$$

$$\alpha_{is} = \alpha_{is_1} + \alpha_{is_2} \qquad \alpha_{is_1} = -k_{is}z_i - \bar{z}_{i-1} \tag{C.69}$$

$$\alpha_{ic} = -\xi_i \tag{C.70}$$

α_{is_2} is function satisfying conditions

$$\text{i.} \qquad \bar{z}_i(-\tilde{\theta}^T\phi_i + \Delta_i + \alpha_{is_2}) \leq \epsilon_i \qquad \forall\ \hat{\theta} \in \Omega_\theta \tag{C.71}$$

$$\text{ii.} \qquad \bar{z}_i\alpha_{is_2} \leq 0 \tag{C.72}$$

and the $\dot{\alpha}_{i-1}$ is obtained from the command filter at the previous step. The modified error dynamics can now be written as

$$\dot{\bar{z}}_i = \dot{z}_i - \dot{\xi}_i \tag{C.73}$$

$$= \dot{x}_i - \dot{\alpha}_{i-1} - \dot{\xi}_i \tag{C.74}$$

$$= x_{i+1} + \theta^T\phi_i + \Delta_i - \dot{\alpha}_{i-1} - \left(-k_i\xi_i + (\alpha_i - \alpha_i^0)\right) \tag{C.75}$$

$$= z_{i+1} + \alpha_i + \theta^T\phi_i + \Delta_i - \dot{\alpha}_{i-1} - \left(-k_i\xi_i + (\alpha_i - \alpha_i^0)\right) \tag{C.76}$$

$$= \bar{z}_{i+1} - k_i\bar{z}_i - \bar{z}_{i-1} - \tilde{\theta}^T\phi_i + \Delta_i + \alpha_{is_2} \tag{C.77}$$

C.2.3 Step n

At the final step, let

$$x_{n+1} = \sigma(x)u \tag{C.78}$$

Let the ideal control input be

$$u^0 = \frac{1}{\sigma(x)}\left(-\hat{\theta}^T\phi_n + \dot{\alpha}_{n-1} - k_n z_n - \bar{z}_{n-1} + \alpha_{ns_2}\right) \tag{C.79}$$

and the filter defined as

$$\dot{\xi}_n = -k_n\xi_n + (u - u^0) \tag{C.80}$$

The modified error dynamics are

$$\begin{aligned}
\dot{\bar{z}}_n &= \dot{z}_n - \dot{\xi}_n && \text{(C.81)}\\
&= \dot{x}_n - \dot{\alpha}_{n-1} - \dot{\xi}_n && \text{(C.82)}\\
&= \sigma(x)u + \theta^T\phi_n + \Delta_n - \dot{\alpha}_{n-1} - \left(-k_n\xi_n + (u - u^0)\right) && \text{(C.83)}\\
&= -k_n\bar{z}_n - \bar{z}_{n-1} - \tilde{\theta}^T\phi_n + \Delta_n + \alpha_{ns_2} && \text{(C.84)}
\end{aligned}$$

Adaptation

$$\dot{\hat{\theta}} = Proj_{\hat{\theta}}\left(\Gamma\sum_{i=1}^{n}\phi_i\bar{z}_i\right) \tag{C.85}$$

where, the projection is defined as

$$Proj_{\hat{\theta}}(\bullet) = \begin{cases} 0 & \text{if } \hat{\theta}_i = \hat{\theta}_{i_{max}} \text{ and } \bullet > 0 \\ \bullet & \text{otherwise} \\ 0 & \text{if } \hat{\theta}_i = \hat{\theta}_{i_{min}} \text{ and } \bullet < 0 \end{cases} \tag{C.86}$$

Such a projection displays following properties

$$\text{P1} \qquad \hat{\theta} \in \Omega_\theta = \{\hat{\theta} : \theta_{min} \le \hat{\theta} \le \theta_{max}\} \tag{C.87}$$

$$\text{P2} \qquad \tilde{\theta}^T\left(\Gamma^{-1}Proj_{\hat{\theta}}(\Gamma\bullet) - \bullet\right) \le 0, \qquad \forall\bullet \tag{C.88}$$

Now consider the candidate Lyapunov function

$$V = \frac{1}{2}\sum_{i=1}^{n}\bar{z}_i^2 \tag{C.89}$$

Differentiating

$$\dot{V} = \sum_{i=1}^{n} \dot{\bar{z}}_i \bar{z}_i \tag{C.90}$$

$$= \sum_{i=1}^{n} -k_i \bar{z}_i^2 + \sum_{i=1}^{n} (-\tilde{\theta}\phi_i + \Delta_i + \alpha_{is_2}) \tag{C.91}$$

$$\leq -2k_v V + \epsilon_v \qquad k_v \triangleq min\{k_1, \ldots, k_n\} \text{ and } \epsilon_v \triangleq \sum_{i=1}^{n} \epsilon_i \tag{C.92}$$

which leads to

$$V(t) \leq e^{(-2k_v t)} V(0) + \frac{\epsilon_v}{2k_v} \left[1 - e^{(-2k_v t)}\right] \tag{C.93}$$

LIST OF REFERENCES

[1] NTSB, "Aviation accident statistics," 2007.

[2] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault Tolerant Control*. Berlin / Heidelberg: Springer, 2 ed., 2006.

[3] J. D. Boskovic and R. K. Mehra, *Failure Detection, Identification and Reconfiguration in Flight Control*, vol. 1 of *Springer Tracts in Advanced Robotics*. Berlin / Heidelberg: Springer, January 2003.

[4] F. Ahmed-Zaid, P. Ioannou, K. Gousman, and R. Rooney, "Accommodation of failures in the F-16 aircraft using adaptive control," *IEEE Control Systems Magazine*, vol. 11, pp. 73–78, Jan 1991.

[5] J. Boskovic and R. Mehra, "A multiple model-based reconfigurable flight control system design," *Proceedings of the 37th IEEE Conference on Decision and Control*, vol. 4, pp. 4503–4508, Dec 1998.

[6] J. Boskovic, S.-M. Li, and R. Mehra, "Reconfigurable flight control design using multiple switching controllers and online estimation of damage-related parameters," *Proceedings of the 2000 IEEE International Conference on Control Applications*, pp. 479–484, 2000.

[7] M. Bodson and J. Groszkiewicz, "Multivariable adaptive algorithms for reconfigurable flight control," *IEEE Transactions on Control Systems Technology*, vol. 5, pp. 217–229, Mar 1997.

[8] "Reconfigurable systems for tailless fighter aircraft-restore (first draft)," System Design Report Contract No. F33615-96-C-3612, Boeing Phantom Works, CDRL Sequence Number A007, St. Louis Missouri, 1998.

[9] Y. Zhang and J. Jiang, "Active fault-tolerant control system against partial actuator failures," *IEEE Proceedings Control Theory and Applications*, vol. 149, pp. 95–104, Jan 2002.

[10] R. J. Patton, "Fault-tolerant control systems: The 1997 situation," *IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes*, vol. 3, pp. 1033–1054, August 1997.

[11] J. S. Brinker and K. A. Wise, "Flight testing of reconfigurable control law on the x-36 tailless aircraft," *Journal Of Guidance, Control, And Dynamics*, vol. 24, pp. 903–909, September 2001.

[12] J. Lunze and T. Steffen, "Control reconfiguration after actuator failures using disturbance decoupling methods," *IEEE Transactions on Automatic Control*, vol. 51, pp. 1590–1601, October 2006.

[13] T. Steffen, *Control Reconfiguration of Dynamical Systems*. Lecture Notes in Control and Information Sciences, Berlin / Heidelberg: Springer, 2005.

[14] M. Bodson, "A reconfigurable nonlinear autopilot," in *AIAA Guidance, Navigation, and Control Conference Exhibit*, (Monterey, California), August 2002.

[15] Y. Zhang and J. Jiang, "Fault tolerant control system design with explicit consideration of performance degradation," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 39, pp. 838–848, July 2003.

[16] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*. Wiley, 2003.

[17] R. F. Stengel, "Towards intelligent flight control," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, pp. 1699–1717, November/December 1993.

[18] J. J. Topsakal, *Robust Motion Planning in Presence of Uncertainties Using a Maneuver Automaton*. PhD thesis, Georgia Institute of Technology, May 2005.

[19] M. Blanke, R. Izadi-Zamanabadi, S. A. Bogh, and C. P. Lunau, "Fault-tolerant control systems a holistic view," *Control Engineering Practice*, vol. 5, pp. 693–702, May 1997.

[20] M. Blanke, M. Staroswiecki, and N. E. Wu, "Concepts and methods in fault-tolerant control," in *Proceedings of the American Control Conference*, (Arlington, VA), June 2001.

[21] Y. M. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Proceedings of the 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (SAFEPROCESS '03)*, pp. 265–276, June 2003.

[22] M. R. Napolitano, Y. An, and B. A. Seanor, "A fault tolerant flight control system for sensor and actuator failures using neural networks," *Aircraft Design*, vol. 3, pp. 103–128, June 2000.

[23] D. McLean and S. Aslam-Mir, "Reconfigurable flight control systems," *International Conference on Control*, pp. 234–242 vol.1, Mar 1991.

[24] Z. Gao and P. J. Antsaklis, "On the stability of the pseudo-inverse method for reconfigurable control systems," *Proceedings of the IEEE National Aerospace and Electronics Conference*, vol. 1, pp. 333–337, May 1989.

[25] P. S. Mayback and R. D. Stevens, "Reconfigurable flight control via multiple model adaptive control methods," *IEEE Transactions On Aerospace And Electronic Systems*, vol. 27, pp. 470–479, May 1991.

[26] R. Stengel, "Intelligent failure-tolerant control," *IEEE Control Systems Magazine*, vol. 11, pp. 14–23, Jun 1991.

[27] D. P. Looze, J. L. Weiss, J. S. Etrno, and N. M. Barrett, "An automatic redesign approach for restructurable control systems," *IEEE Control Systems Magazine*, vol. 5, pp. 16–22, May 1985.

[28] Y. Ochi and K. Kanai, "Design of restructurable flight control systems using feedback linearization," *Journal Of Guidance*, vol. 14, pp. 903–911, September 1991.

[29] Y. Ochi and K. Kanai, "Application of restructurable flight control system to large transport aircraft," *Journal Of Guidance, Control, and Navigation*, vol. 18, pp. 365–370, March 1995.

[30] C. Y. Huang and R. F. Stengel, "Restructurable control using proportional-integral implicit model following," *Journal of Guidance*, vol. 13, pp. 303–309, March 1990.

[31] J. S. Eterno, J. L. Weiss, D. P. Looze, and A. Willsky, "Design issues for fault tolerant-restructurable aircraft control," in *Proceedings of 24th Conference on Decision and Control*, (Ft. Lauderdale, Florida), December 1985.

[32] R. Eslinger and P. Chandler, "Self-repairing flight control system program overview," *Aerospace and Electronics Conference, 1988. NAECON 1988., Proceedings of the IEEE 1988 National*, pp. 504–511 vol.2, May 1988.

[33] X. Zhang, M. M. Polycarpon, and T. Parisini, "Integrated design of fault diagnosis and accomodation schemes for a class of nonlinear systems," in *Proceedings of the 40th IEEE Conference on Decision and Control*, (Orlando, Florida), pp. 1448–1453, December 2001.

[34] S. Thomas, H. G. Kwatny, B.-C. Chang, and C. Belcastro, "Regulator design for control surface failure accomocation in an F-16," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (San Francisco, CAlifornia), August 2005.

[35] G. Tao, S. Chen, and S. Joshi, "An adaptive actuator failure compensation controller using output feedback," *IEEE Transactions on Automatic Control*, vol. 47, pp. 506–511, Mar 2002.

[36] G. Tao, S. Chen, J. Fei, and S. M. Joshi, "An adaptive actuator failure compensation scheme for controlling a morphing aircraft model," in *Proceedings of 42th Conference on Decision and Control*, (Maui, Hawai), December 2003.

[37] J. Watton, *Condition Monitoring and Fault Diagnosis in Fluid Power Systems*. Wllis Horwood, 1992.

[38] R. J. Patton, P. M. Frank, and R. N. C. (eds.), *Issues of Fault Diagnosis for Dynamic Systems*. Berlin: Springer-Verlag, 2000.

[39] R. Isermann, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Berlin: Springer-Verlag, 2006.

[40] R. Isermann, "Fault diagnosis of machines via parameter estimation and knowledge processing: tutorial paper," *Automatica (Journal of IFAC)*, vol. 29, pp. 815–835, July 1993.

[41] R. J. Patton, "Fault detection and diagnosis in aerospace systems using analyticalredundancy," *Computing and Control Engineering Journal*, pp. 127–136, May 1991.

[42] S. Dash and V. Venkatasubhamanian, "Challenges in industrial applications of fault diagnosis systems," *Computers and Chemical Engineering*, vol. 24, no. 2, pp. 785–791, 2000.

[43] H. Hammouri, P. Kabore, S. Othman, and J. Biston, "Failure diagnosis and nonlinear observer. application to a hydraulic process," *Journal of the Franklin Institute*, vol. 339, no. 4-5, pp. 455 – 478, 2002.

[44] R. Isermann, "Model-based fault-detection and diagnosis - status and applications," *Annual Reviews in Control*, vol. 29, no. 1, pp. 71 – 85, 2005.

[45] R. J. Patton and J. Chen, "Robust fault detection of jet engine sensor systems using eigenstructure assignment," *Journal Of Guidance, Control, And Dynamics*, vol. 15, pp. 1491–1497, November 1992.

[46] R. J. Patton and J. Chen, "Robust fault detection using eigenstructure assignment: A tutorial consideration and some new results," *Control Engineering Practice*, vol. 5, no. 5, pp. 671–682, 1997.

[47] P. Kudva, N. Vishwanadham, and A. Ramakrishna, "Observers for linear systems with unknown inputs.," *Automatica*, vol. 22, no. 3, pp. 333–344, 1980.

[48] R. Patton, R. Clark, and P. Frank, *Fault Diagnosis in Dynamic Systems: theory and applications*. Prentice Hall, 1989.

[49] G. J. Preston, D. N. Shields, and S. Daley, "Application of robust nonlinear fault detection observer to a hydraulic process," in *Proceedings of UKACC Internaltional Conference on Control*, pp. 1484–1489, 1996.

[50] D. Yu, "Fault diagnosis for a hydraulic drive system using a parameter-estimation method," *Control Engineering Practice*, vol. 5, no. 9, pp. 1283–1297, 1997.

[51] R. Isermann, "Process fault detection based on modeling and estimation methods – a survey," *Automatica*, vol. 20, no. 4, pp. 387–404, 1984.

[52] L. An and N. Sepehri, "Hydraulic actuator circuit fault detection using extended kalman filters," in *Proceedings of the American Control Conference*, pp. 4261–4266, 2003.

[53] Y. M. Cho and R. Rajamani, "A systematic approach to adaptive observer synthesis for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 42, pp. 534 –537, apr 1997.

[54] R. Rajamani and J. Hedrick, "Adaptive observers for active automotive suspensions: theory and experiment," *IEEE Transactions on Control Systems Technology*, vol. 3, pp. 86 –93, mar 1995.

[55] P. Garimella and B. Yao, "Model based fault detection of an electro-hydraulic cylinder," in *American Control Conference*, pp. 484–489, June 2005.

[56] U. Horowitz, "Application of quantitative feedback theory to flight control problems," in *Proceedings of the 29th IEEE Conference on Decision and Control*, (Honolulu, Hawai), pp. 2593–2598, December 1980.

[57] M. Keating, M. Pachter, and C. Houpis, "Qft applied to fault tolerant flight control system design," *American Control Conference, 1995. Proceedings of the*, vol. 1, pp. 184–188 vol.1, Jun 1995.

[58] H. Rauch, "Autonomous control reconfiguration," *IEEE Control Systems Magazine*, vol. 15, pp. 37–48, Dec 1995.

[59] R. A. Hess and S. R. Wells, "Sliding mode control applied to reconfigurable flight control design," *Journal of Guidance, Control, and Dynamics*, vol. 26, pp. 452–462, May 2003.

[60] F. K. Ufuk Demirci, "Fault tolerant control with re-configuring sliding-mode schemes," *Turk J Elec Engin*, vol. 13, no. 1, pp. 175–187, 2005.

[61] F. Liao, J. L. Wang, E. K. Poh, and D. Li, "Reliable automatic landing control against actuator stuck faults," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Providence, Rhodes Island), 2004.

[62] F. Liao, J. L. Wang, E. K. Poh, and D. Li, "Fault-tolerant robust automatic landing control design," *Fault-Tolerant Robust Automatic Landing Control Design*, vol. 28, pp. 854–871, Spetember 2005.

[63] G. R. Drozeski, *A Fault-Tolerant Control Architecture for Unmanned Aerial Vehicle*. PhD thesis, Georgia Institute of Technology, December 2005.

[64] M. Huzmezan and J. Maciejowski, "Reconfiguration and scheduling in flight using quasi-lpv high-fidelity models and mbpc control," *American Control Conference, 1998. Proceedings of the 1998*, vol. 6, pp. 3649–3653 vol.6, Jun 1998.

[65] S. Ganguli and M. G. Balas, "Reconfigurable lpv control design for boeing 747-100/200 longitudinal axis," in *Proceedings of the American Control Conference*, (Anchorage AK), May 2002.

[66] J. Shamma and M. Athans, "Gain scheduling: potential hazards and possible remedies," *IEEE Control Systems Magazine*, vol. 12, pp. 101–107, Jun 1992.

[67] D. D. Moerder, N. Halyo, J. R. Broussard, and A. K. Caglayan, "Application of precomputed control laws in a reconfigurable aircraft flight control system," *Journal of Guidance*, vol. 12, no. 3, pp. 325–333, 1989.

[68] C. Zheng, R. J. Patton, and J. Chen, "Robust fault-tolerant systems synthesis via lmi," *SAFEPROCESSES'97*, pp. 347–352, 1997.

[69] G. Meyer and L. Hunt, "Application of non-linear transformations to automatic flight control," *Automatica*, vol. 20, no. 1, pp. 130–107, 1984.

[70] W. D. Morse and K. A. Ossmant, "Model following reconfigurable flight control system for the afti/F-16," *Journal of Guidance*, vol. 13, pp. 303–309, November 1990.

[71] Z. Yang and M. Blanke, "The robust control mixer module method for control reconfiguration," *American Control Conference, 2000. Proceedings of the 2000*, vol. 5, pp. 3407–3411 vol.5, 2000.

[72] G.-S. Wang, Q. Lv, B. Liang, and G.-R. Duan, "Design of reconfiguring control systems via state feedback eigenstructure assignment," *International Journal of Information Technology*, vol. 11, no. 7, 2005.

[73] H.-S. Kim and Y. Kim, "Partial eigenstructure assignment algorithm in flight control system design," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 35, pp. 1403–1409, Oct 1999.

[74] Y. Zhang and J. Jiang, "Design of integrated fault detection, diagnosis and reconfigurable control systems," *Proceedings of the 38th IEEE Conference on Decision and Control*, vol. 4, pp. 3587–3592 vol.4, 1999.

[75] B. M. and P. W. A., "Command limiting in reconfigurable flight control," *Journal Of Guidance, Control, and Dynamics*, vol. 21, no. 4, pp. 639–646, 1998.

[76] M. Morari and J. H. Lee, "Model predictive control: Past, present and future," *Computers and Chemical Engineering*, vol. 23, pp. 667–682, September 1999.

[77] J. M. Maciejowski, "Modelling and predictive control: Enabling technologies for reconfiguration," *Annual Reviews in Control*, vol. 23, no. 1, pp. 13–23, 1999.

[78] M. Huzmezan and J. Maciejowski, "Reconfigurable flight control of a high incidence research model using predictive control," in *UKACC International Conference on CONTROL 98*, September 1998.

[79] E. C. Kerrigan and J. M. Maciejowski, "Fault-tolerant control of a ship propulsion system using model predictive control," 1999.

[80] T. Keviczky and G. J. Balas, "Receding horizon control of an F-16 aircraft: A comparative study," *Control Engineering Practice*, vol. 14, pp. 1023–1033, June 2005.

[81] D. G. Ward and R. L. Barron, "A self-designing receding horizon optimal flight controller," in *Proceedings of American Control Conference*, (Seattle, Washington), June 1995.

[82] S. Sastry and M. Bodson, *Failure Detection, Identification and Reconfiguration in Flight Control*. Prentice-Hall Advanced Reference Series, Prentice Hall, 1994.

[83] M. Bodson, "Identification with modeling uncertainty and reconfigurable control," *Proceedings of the 32nd IEEE Conference on Decision and Control*, pp. 2242–2247 vol.3, Dec 1993.

[84] R. B. Miller and M. Pachter, "Maneuvering flight control," *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 4, 1997.

[85] Y. Shtessel, J. Buffington, and S. Banda, "Multiple timescale flight control using recon. gurable sliding modes," *Journal of Guidance, Control, and Dynamics*, vol. 22, pp. 873–883, November 1999.

[86] J. Hung, W. Gao, and J. Hung, "Variable structure control: a survey," *IEEE Transactions on Industrial Electronics*, vol. 40, pp. 2–22, Feb 1993.

[87] E. N. Johnson, *Limited Authority Adaptive Flight Control*. PhD thesis, Georgia Institute of Technology, November 2000.

[88] C. C. Jorgensen, "Direct adaptive aircraft control using dynamic cell structure neural networks," Tech. Rep. TM 112198, NASA, Ames Research Center, Moffett Field, California, May 1997.

[89] A. J. C. Ilkay Yavrucuk, J. V. R. Prasady, "Adaptive limit detection and avoidance for carefree maneuvering," in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, (Montreal, California), August 2001.

[90] N. A. Sahani and J. F. Horn, "Command limiting for full-envelope guidance and control of rotorcraft," in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, (San Francisco, California), August 2005.

[91] E. M. Atkins, "Dynamic waypoint generation given reduced flight performance," in *42nd AIAA Aerospace Sciences Meeting and Exhibit*, 2004.

[92] A. Richards and J. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," *American Control Conference, Proceedings of the 2002*, vol. 3, pp. 1936–1941, 2002.

[93] T. Schouwenaars, B. Mettler, E. Feron, and J. P. How4, "Robust motion planning using a maneuver automaton with built-in uncertainties," in *Proceedings of the 2003 American Control Conference*, vol. 3, pp. 2211–2216, June 2003.

[94] J. Schierman, N. Gandhi, J. Hull, and D. Ward, "Flight test results of an adaptive guidance system for reusable launch vehicles," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Providence, Rhode Island), August 2004.

[95] M. Jelali and A. Kroll, *Hydraulic Servo-systems: Modeling Identification and Control*. Springer, 2004.

[96] G. A. Sohl and J. E. Bobrow, "Experimenst and simulations on the nonlinear control of a hydraulic servosystem," *IEEE Transactions on Control Systems Technology*, vol. 7, pp. 238–247, March 1999.

[97] M.-C. Shih and Y.-R. Sheu, "The adaptive position control of an electro hydraulic servo cylinder," *JSME International Journal*, vol. 34, no. 3, pp. 370–376, 1991.

[98] S. Tafazoli, C. de Silva, and P. Lawrence, "Tracking control of an electrohydraulic manipulator in the presence of friction," *IEEE Transactions on Control Systems Technology*, vol. 6, pp. 401 –411, may 1998.

[99] A. Alleyne and R. Liu, "A simplified approach to force control for electrohydraulic systems," *Control Engineering Practice*, vol. 8, no. 12, pp. 1347 – 1356, 2000.

[100] A. Alleyne and K. J. Hedrick, "Nonlinear adaptive control of active suspension," *IEEE Transactions on Control System Technology*, vol. 3, no. 1, pp. 94–101, 1995.

[101] J.-S. Lin and I. Kanellakopoulos, "Nonlinear design of active suspensions," *IEEE Control Systems Magazine*, vol. 3, no. 1, pp. 45–49, 1995.

[102] M. Münchhof, *Model-Based Fault Detection for a Hydraulic Servo Axis*. PhD thesis, Technischen Universitt Darmstadt, May 2006.

[103] T. A. Blanchet and F. E. Kennedy, "Sliding wear mechanism of polytetrafluoroethylene (PTFE) and PTFE composites," *Wear*, vol. 153, no. 5-6, pp. 229–243, 92.

[104] A. Ray and S. Tangirala, "Stochastic modeling of fatigue crack dynamics for online failure prognostics," *IEEE Transactions on Control Systems Technology*, vol. 4, pp. 443–451, July 1996.

[105] R. Patankar, A. Ray, and A. Lakhtakia, "A state-space model for fatigue crack growth," *International Journal of Fracture*, vol. 90, pp. 235–249, April 1998.

[106] P. Kallappa, M. S. Holmes, and A. Ray, "Life-extending control of fossil fuel power plants*," *Automatica*, vol. 33, no. 6, pp. 1101–1118, 1997.

[107] J. Khedkar, I. Negulescu, and E. I. Meletis, "Sliding wear behaviour of ptfe composities," *Wear*, vol. 252, no. 5-6, pp. 361–369, 2002.

[108] S. Tangirala, J. Caplin, E. Keller, and A. Ray, "Life extending control of gas turbine engines," in *American Control Conference*, vol. 1, pp. 614–617, 1999.

[109] M. O. Rauw, "A simulink environment for flight dynamics and control analysis - application to the dhc-2 'beaver' part i. implementation of a model library in simulink," Master's thesis, Delft University of Technology, September 1993.

[110] M. O. Rauw, "A simulink environment for flight dynamics and control analysis - application to the dhc-2 'beaver' part ii. nonlinear analysis of the 'beaver' autopilot," Master's thesis, Delft University of Technology, September 1993.

[111] H. E. Merritt, *Hydraulic Control Systems*. John Wiley and Sons, 1967.

[112] B. Yao and M. Tomizuka, "Adaptive robust control of mimo nonlinear systems in semi-strict feedback forms," *Automatica*, vol. 37, no. 9, pp. 1305 – 1321, 2001.

[113] B. S. Nau, "An historical review of studies of polymeric seals in reciprocating hydraulic systems," *Proceedings- Institution Of Mechanical Engineers Part J Journal Of Engineering Tribology*, vol. 213, pp. 215–226, January 1999.

[114] M. Karpenko and N. Sepehri, "Hardware-in-the-loop simulator for research on fault tolerant control of electrohydraulic flight control systems," in *Proceedings of 2006 American Control Conference*, (Minneapolis, Minnesota), pp. 4645–4651, June 2006.

[115] J. Kocijan and R. Karba, "The chemical process application of multivariable control hardware and algorithm testing by means of simulation," *Simulation Practice and Theory*, vol. 5, pp. 153–165, 1997.

[116] R. Isermann, J. Schaffnita, and S. Sinsela, "Hardware-in-the-loop simulation for the design and testing of engine-control systems," *Control Engineering Practice*, vol. 7, pp. 643–653, May 1999.

[117] E. N. Johnson and S. Fontaine, "Use of flight simulation to complement flight testing of low-cost uavs," in *Proceedings of the AIAA Modeling and Simulation Technology Conference*, 2001.

[118] M. S. K. Cosic, T. Kostic, and M. Volarevic, "Design and implementation of a hardware-in-the-loop simulator for a semi-automatic guided missile system," *Simulation Practice and Theory*, vol. 7, pp. 107–123, 1999.

[119] M. Linjama, T. Virvalo, J. Gustafsson, J. Lintula, V. Aaltonen, and M. Kivikoski, "Hardware-in-the-loop environment for servo system controller design, tuning and testing," *Microprocessors and Microsystems*, vol. 24, pp. 13–21, 2000.

[120] S. Mansoor, D. Jones, D. Bradley, F. Aris, and G. Jones, "Hardware-in-the-loop simulation of a pumped storage hydro station," in *Proceedings of Mechatronics 98, IASTED International Conference PES2000*, (Marbella, Spain), pp. 633–638, 1998.

[121] F. D. Yeaple, *Fluid Power Design Handbook*. Marcel Dekker, Inc., 3 ed., 1996.

[122] M. Karpenko and N. Sepehri, "Fault tolerant control of a servohydraulic positioning system with crossport leakage," *IEEE Transactions on Control Systems Technology*, vol. 13, pp. 155–161, January 2005.

[123] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ 07458: Prentice Hall, 3 ed., 2002.

[124] M. Boutayeb and D. Aubry, "A strong tracking extended kalman observer for nonlinear discrete-time systems," *IEEE Transactions on Automatic Control*, vol. 44, no. 8, pp. 1550–1556, 1999.

[125] L. Ljung, "Asymptotic behaviour of the extended kalman filter as a parameter estimator for linear systems," *IEEE Transactions on Automatic Control*, vol. 24, pp. 36–50, 1979.

[126] R. K. Gunther, S. E. Yaz, and R. Unbehauen, "Stochastic stability of the discrete-time extended kalman filter," *IEEE Transactions on Automatic Control*, vol. 44, no. 4, pp. 714–728, 1999.

[127] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Hoboken, NJ, USA: John Wiley and Sons, Inc., 2006.

[128] T. S. Schei, "A finite-difference method for linearization in nonlinear estimation algorithms," *Automatica*, vol. 33, no. 11, pp. 2051–2058, 1997.

[129] B. M. Quine, "A derivative-free implementation of the extended kalman filter," *Automatica*, vol. 42, no. 11, pp. 1927–1934, 2006.

[130] R. W. E. van der Merwe, S. Julier, A. Bogdanov, G. Harvey, and J. Hunt, "Sigma-point kalman filters for nonlinear estimation and sensor fusion: Applications to integrated navigation," in *AIAA guidance navigation and control conference*, 2004.

[131] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[132] M. Norgaard, N. K. Poulsen, and O. Ravn, "New developments in state estimation for nonlinear systems," *Automatica*, vol. 36, pp. 1627–1638, 2000.

[133] G. H. Golub and C. F. V. Loan, *Matrix Computations*. The John Hopkins University Press, third ed., 1996.

[134] K. Xiong, H. Y. Zhang, and C. W. Chan, "Performance evaluation of ukf-based nonlinear filtering," *Automatica*, vol. 42, pp. 261–270, October 2006.

[135] P. M. Frank, "Analytical and qualitative model based fault diagnosisa survey and some newresults," *European Journal of Control*, vol. 2, pp. 6–28, 1996.

[136] J. Jiang, "Fault-tolerant control systems–an introductory overview," *Acta Automatica Sinica*, vol. 31, no. 1, pp. 161–174, 2005.

[137] K. Zhou, J. C. Doyle, and K. Glover, *Robust and optimal control*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1996.

[138] S. Kanev, *Robust Fault-Tolerant Control*. PhD thesis, University of Twente, 2004.

[139] M. Mahmoud, J. Jiang, and Y. Zhang, *Active Fault Tolerant Control Systems*, vol. 287 of *Lecture Notes in Control and Information Sciences*. Berlin, Heidelberg: Springer Berlin Heidelberg, March 2003.

[140] B. Yao and M. Tomizuka, "Adaptive robust control of nonlinear systems: a general framework and new schemes," in *in Proc. of IEEE Conference on Decision and Control*, pp. 2489–2494, 1997.

[141] A. Bemporad and M. Morari, *Robustness in identification and control*, vol. 245 of *Lecture Notes in Control and Information Sciences*. London: Springer London, 1999.

[142] W. S. Levine, *The control handbook*. CRC Press, 1996.

[143] T. Hu and Z. Lin, *Control systems with actuator saturation: analysis and design*. Springer, 2001.

[144] L. T. Nguyen, M. E. Ogburn, W. P. Gilbert, K. S. Kibler, P. W. Brown, and P. L. Deal, "Simulator study of stall/post-stall characteristics of a fighter airplane with relaxed longitudinal static stability," Tech. Rep. TP 1538, NASA, Langley Research Center, Hampton, Virginia, December 1979.