A RECONFIGURATION SCHEME FOR FLIGHT CONTROL ADAPTATION TO FIXED-POSITION ACTUATOR FAILURES

By

ROBERT S. EICK

A THESIS PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

UNIVERSITY OF FLORIDA

2003

I dedicate this work to my family.

ACKNOWLEDGMENTS

This research was inspired by the Intelligent Flight Control System (IFCS) flight research project at NASA Dryden Flight Research Center (DFRC). The author thanks John Burken of NASA DFRC for his support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract of Thesis Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Master of Science

A RECONFIGURATION SCHEME FOR FLIGHT CONTROL ADAPTATION TO
FIXED-POSITION ACTUATOR FAILURES

By

Robert S. Eick

August 2003

Chair: Richard C. Lind, Jr.
Major Department: Mechanical and Aerospace Engineering

This work considers the problem of redesigning a flight control system to achieve acceptable stability and performance in the presence of a control surface failure. The particular failure considered is the fixed-position or jammed actuator failure. This is an especially difficult type of failure to overcome since the operational control surfaces must be reconfigured not only to achieve the control objective but also to compensate for a disturbance due to the failure. The proposed reconfiguration scheme relies on three interdependent systems: a fault detection and isolation (FDI) system, a stabilization system, and a fault-tolerant control system. FDI is developed using an artificial neural network that monitors the feedback measurements of the flight control system. Stabilization is based upon a least-squared optimization algorithm to determine a new trim condition for the failed aircraft. Two fault-tolerant control techniques are developed to complete the reconfiguration scheme. The first is a linear quadratic regulator (LQR) approach and the second is an H_∞ approach. In each approach the effects of the jammed surface are treated as a measurable constant disturbance to the system. For the LQR approach a controller is designed that balances the jammed surface (disturbance rejection), and provides command-tracking. For the H_∞ approach

a two-step design process is used where first the feedforward part of the controller is designed to achieve perfect trajectory following then the feedback part of the controller is designed using H_∞ regulator theory. The H_∞ approach relies on the use of a low-pass filter in controller synthesis to limit the disturbance forces and accurately simulate the effects of the jammed surface. The two methods, along with FDI and stabilization, are demonstrated on a high fidelity nonlinear six-degree-of-freedom F/A-18 simulator. Simulation results are presented with a significant control surface failure and show the benefits on stability and performance using the developed reconfiguration techniques. The LQR and H_∞ methods achieved virtually the same results for the targeted failure class with both regaining stability and restoring performance in all instances.

CHAPTER 1
INTRODUCTION

To achieve the safety goals in air travel it will be necessary to design flight control systems that can compensate for failures and damage to aircraft. All modern aircraft depend upon their flight control system to provide the handling qualities necessary for successful flight. When a component of the flight control system fails or is damaged it is desirable that the safety of the aircraft not be compromised. Reconfigurable controls attempts to address this issue by developing reconfigurable control schemes that enhance survivability and safety to allow an aircraft to be recovered in flight after it has suffered component failure or damage. The primary benefit of reconfigurable controls is the ability to significantly enhance flight safety. Beyond this, a reconfigurable controller has the potential to restore desired stability and performance characteristics so that a crippled aircraft can complete its mission and land successfully. With their clear benefit in both military and civil aircraft, reconfiguration techniques and strategies have become the focus of many investigators in recent years and are currently receiving significant attention.

A reconfiguration scheme consists of three parts: a fault detection and isolation (FDI) procedure, a reconfiguration logic, and a fault-tolerant control law. The FDI procedure detects a failure and isolates it to a specific component of the system and the reconfiguration logic adjusts the control law so that system stability and performance are restored. This work focuses on developing a fault-tolerant control law and FDI procedure for a specific class of aircraft failures. The resulting scheme provides a fast and efficient method to detect failure and procedures to overcome the effect of a control failure on stability and performance. The failure class analyzed throughout this work is a fixed-position or jammed actuator failure, which results in a flight control

surface becoming inoperable. The goal of a reconfigurable controller for this failure class is to reconfigure the control law to use the remaining operational control surfaces in such a manner that the prefailure flying qualities are restored. The objective of this work is to develop a reconfiguration scheme that is easily implementable into current flight software and offers a measurable degree of reliability for the targeted type of failure.

Analysis has shown that the probability of an actuator failure is extremely low, however, in the event failure occurs, the most probable type of failure is the fixed-position actuator failure [1]. The success of a reconfigurable controller depends crucially on the ability of the FDI module to promptly and accurately identify failure. This work proposes the development of artificial neural networks to accomplish this task. The function performed by the neural network for aircraft FDI is the mapping of aircraft measurements into fault categories that describe which surface has failed and at what position. Once the failure has been positively identified, the next step in the proposed reconfiguration scheme is the development of a fault-tolerant controller capable of using the FDI information to effectively restore stability and performance.

Fault-tolerance deals with the ability to complete a task satisfactorily (reliability) and the likehood of conducting an operation safely without endangering the human operators of the controlled system (survivability) [2]. Two fault-tolerant control (FTC) methods are developed and evaluated in this work. The first is a linear quadratic regulator (LQR)-based technique while the second is an H_∞-based technique. In each approach the effects of the jammed surface are treated as a constant disturbance to the system. While all nominal controllers have some inherent robustness to a limited failure class, an appropriately designed reconfigurable controller should have a much larger region of survivability. These proposed techniques and the ensuing reconfiguration schemes appear to meet the challenges of the fixed-position actuator failure well for both linear and nonlinear simulations.

1.1 Motivating Example

On November 12, 2001 an American Airlines Airbus Industry A300-600, Figure (1–1) [3], Flight 587 en route from John F. Kennedy International Airport (JFK), Jamaica, New York, sustained a catastrophic failure when the vertical stabilizer and rudder separated from the fuselage shortly after takeoff [4, 5]. The 2 pilots, 7 flight attendants, 251 passengers, and 5 persons on the ground lost their lives when the aircraft broke apart and crashed into the residential community of Belle Harbor, New York, Figure (1–2) [6]. The resulting investigation examined many issues including the adequacy of the certification standards for transport-category airplanes, the structural requirements and integrity of the vertical stabilizer and rudder, the operational status of the rudder system at the time of the accident, the adequacy of pilot training, and the role of pilot actions in the accident.

Figure 1–1: American Airlines Airbus A300-600

It was determined that before the separation of the vertical stabilizer and rudder, Flight 548 encountered two wake vortices from a Boeing 747, which had departed JFK ahead of the accident aircraft. The two airplanes were separated by about 5 miles and 90 seconds at the time of the vortex encounters. During and shortly after the second encounter, the flight data recorder (FDR) on the accident aircraft recorded several large

rudder movements and corresponding pedal movements to full or nearly full available rudder deflection in one direction followed by full or nearly full available rudder deflection in the opposite direction.[1] The subsequent loss of reliable rudder position data is consistent with the vertical stabilizer separating from the airplane. Among the potential causes examined for this catastrophic failure were rudder system malfunction, as well as flight crew action.

Figure 1–2: Flight 548 crashes in Queens, New York

The National Transportation Safety Board and Airbus engineers believe that large side loads were likely present on the vertical stabilizer and rudder at the time they separated from the airplane. Calculations and simulations show that, at the time of the separation, the airplane was in an 8° to 10° airplane nose-left sideslip while the rudder was deflected 9.5° to the right. Airbus engineers have determined that this combination of local nose-left sideslip on the vertical stabilizer and right rudder

[1] Preliminary information based on FDR data and an analysis of the manner in which rudder position data is filtered by the airplane's system indicates that within about 7 seconds, the rudder traveled 11° right for 0.5 seconds, 10.5° left for 0.3 seconds, between 11° and 10.5° right for about 2 seconds, 10° left for about 1 second, and finally, 9.5° right before the data became unreliable.

deflection produced loads on the vertical stabilizer that could exceed the airplane's design loads. The Federal Aviation Administration (FAA) concluded that it was this dangerous combination of sideslip angle and rudder position which resulted in the complete lost of aircraft, crew, and passengers.

Figure 1–3: Vertical Stabilizer of Flight 548

While the vertical stabilizer and rudder appeared to separate cleanly from the fuselage, Figure (1–3) [7], the flight controller was inadequately designed to regain control of the crippled aircraft. The resulting configuration of leading edge flaps, trailing edge flaps, ailerons, and elevators were incapable of countering the sudden roll and yawing moment generated by the absence of the vertical stabilizer. Rolling upside down and finally out of control the airplane succumbed to the increasing aerodynamic forces as a massive engine, wing, and fuselage breakup scattered remains throughout Jamaica Bay and Long Island New York. The tragedy of the Flight 587 accident endures as another motivation in the emerging field of Reconfigurable Controls, which aims to develop flight controllers which can handle failures such as this.

1.2 Overview

The purpose of this work is to propose an adaptive scheme in reconfigurable flight controls capable of recovering desired performance and stability characteristics for an aircraft experiencing a fixed-position actuator failure. Chapter 2 provides a review of the current literature in the area of reconfigurable flight controls. The mathematical

details of aircraft flight mechanics and a non-technical introduction into artificial neural networks is given in Chapter 3. Chapter 4 presents a fault detection and isolation (FDI) procedure for fixed-position actuator failures using artificial neural networks. The stabilization problem and aircraft fault modeling is reviewed in Chapter 5. Chapter 6 focuses on the theoretical development of fault-tolerant controllers using LQR-based and H_∞-based techniques. Both of these fault-tolerant control methods along with FDI and stabilization results are combined in Chapter 7 to demonstrate the success of the proposed scheme on a nonlinear F/A-18 simulation. Finally, Chapter 8 presents the conclusions of this work.

CHAPTER 2
REVIEW OF LITERATURE

In 1984 the Air Force Flight Dynamics Laboratory initiated the first research program dedicated to the investigation of reconfiguration technology for flight control systems with the Self-Repairing Flight Control Systems Program. The main objective of this program was to significantly improve the reliability, maintainability, survivability, and life cycle costs of aircraft flight control systems through aerodynamic reconfiguration and maintenance diagnostics. Special consideration was given to developing a reconfiguration strategy that uses the remaining control surfaces to substitute for the lost force and moment generating capabilities when a single control surface becomes impaired from failure or battle damage [8]. Since this original initiative many methods have been proposed to solve the reconfiguration problem for flight controls. This chapter outlines the state of the art, which remains largely a theoretical topic with most applications studies based upon aerospace systems.

2.1 Introduction

The objective of reconfigurable controls is to detect a failure using the feedback signals of the flight control system then reconfigure the control law in a fashion that restores the desired stability and performance characteristics of the aircraft. There is a substantial body of reconfigurable controls literature that includes applications in hazardous chemical plants, the control of nuclear power plant reactors, space craft, and the control of unstable fly-by-wire aircraft. Research into reconfigurable control, however, is largely motivated by the control problems encountered in aircraft system design. The goal of these researchers is to provide self-repairing capability to enable a pilot to land an aircraft safely in the event of a serious malfunction [9].

The main requirement for any fault-tolerant controller as part of a reconfiguration scheme is that, subsequent to a malfunction in the system, it should either maintain some acceptable level of performance and stability or degrade gracefully. While significant theoretical progress has been made in academia, few results have been applied to real vehicles. The view is usually taken that the application of a complex fault-tolerant controller is best applied to systems where the governing principles are easily understood and verifiable. This "simplistic" approach has consistently caused a reluctance within the field to experiment on systems which pose intolerable risks in terms of safety, cost, instability or unpredictability. The general view is that simpler controllers, with fewer components or lines of software code are intrinsically more reliable and that further complexity would unnecessarily increase the overall risk of failure during routine operation [10, 11].

The initial step in developing any reconfiguration scheme is determining the limitations of the conventional feedback controller. Strategies for reconfiguration are generally application-specific and are normally dependent on available equipment and measurements. The task then becomes the design of a controller with suitable structure to guarantee stability and satisfactory performace, not only when all components are fully operational, but also in the case when sensors, actuators and other components malfunction. Owen [12] referred to a control system with this structure as one that possesses integrity or that has control loops that possess loop integrity, while Veilltette et al. [13] and Birdwell et al. [14] prefer to use the term reliable control.

Figure (2–1) shows the general schematic of a reconfigurable control system with four main components: the model (including the plant dynamics, actuators, and sensors), the fault detection and isolation (FDI) module, the controller, and the supervision module. The solid lines represent signal flow (commands, feedback, etc.) while the dashed lines represent adaption (tuning, scheduling, reconfiguration, or restructuring). The possible faults include malfunctions in sensors, actuators, or

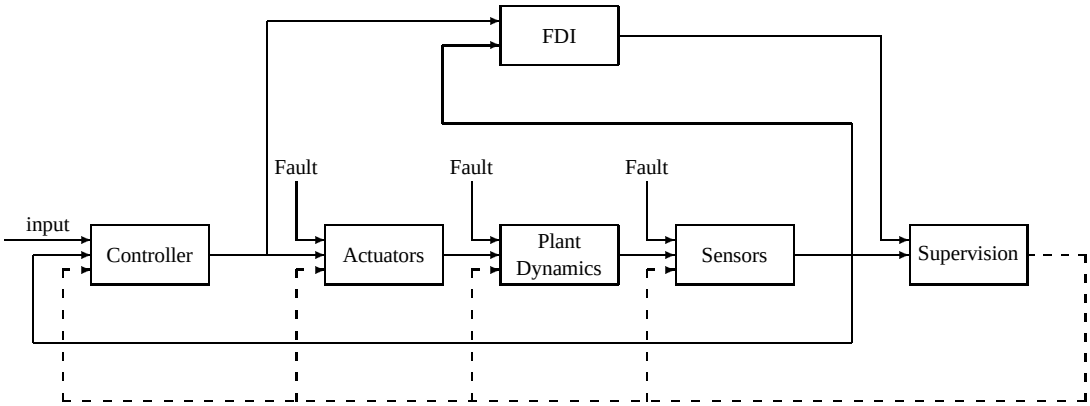

Figure 2–1: General schematic of a reconfigurable control system with supervision

other components of the plant. The FDI subsystem constantly monitors the system's performance and stability using the feedback signal of the closed-loop system and the position commands to the actuators, then provides the supervision subsystem with information about the onset, location and severity of any fault. Based on the system's measurements together with FDI information, the supervision system will reconfigure, tune, or adapt the controller to accommodate for the effects of the fault.

The relationship between the four main components of Figure (2–1) allows the reconfigurable control problem to be solved in a very systematic manner. The principles involved in the systematic design and development of a reconfigurable controller are outlined by Blanke et al. [15]. He demonstrated that the development of each subsystem affects the development of the overall system, this interdependence necessitates a comprehensive strategy for reliable and highly efficient control law redesign. Ultimately, the design procedure is a multidisciplinary task involving relevant science/technology, control theory and design, signal processing and human factors.

2.2 The State of the Art

Over the past two decades there has been an extensive investigation into many possible solutions of the reconfigurable controls problem. Figure (2–2) depicts the areas of greatest contribution and their interrelationship towards forming a complete reconfiguration strategy.

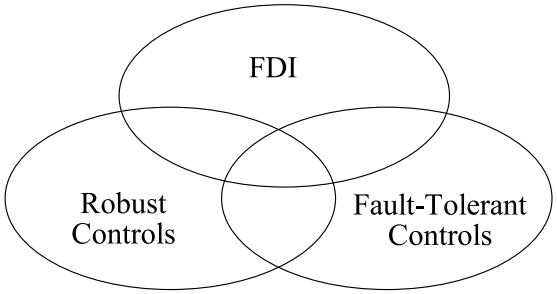

Figure 2–2: Venn Diagram of Reconfiguration Strategies

2.2.1 Fault Detection and Isolation (FDI)

With the development of powerful quantitative and/or qualitative modelling tools and artificial neural newtworks the field of FDI has become very refined [16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33]. However, much of the research has yet to be combined with fault-tolerant controllers to present a complete reconfiguration strategy. The main issue to overcome with traditional methods for FDI involves specification of all critical design conditions and/or extensive overdesign for unpredicted or uncertain conditions. To date, as noted by Barron et al. [34], the best FDI systems only consider a fraction of the operational fault conditions that might be encountered and only a small portion of those that may be encountered are actually used explicitly for design because the number of possible fault conditions is very large. This fact severely inhibits the design of a global FDI system for complex systems such as aircraft. It may also be important to identify the fault type and its severity as well as the reason for the fault development. When these functions are included along with FDI, the function performed is called fault diagnosis. Several investigators, including Legg [35], Herrin [36], Bell [37], Hunt et al. [38], and Blanke et al. [15] have discussed the use of failure mode effects analysis (FMEA) techniques to determine systematically how fault effects in components relate to fault inputs, outputs, or elements within the components. FMEA is a bottom-up analytical process which identifies potential hazards of a new system with the goal to anticipate, identify and avoid failures in the design and development stages.

2.2.2 Robust Control

Robust control design has received much attention within the controls community since the late 1970's. The few cases that have attempted to directly apply robust control theory to reconfigurable controls usually have not considered the effects of the faults on the control system [39, 40, 41, 42]. Eterno et al. [43] and Stengel [44] note that this passive approach to reconfigurable controls makes the fundamental assumption that faults can be modelled as uncertainties. Admittedly, there do exist robust control techniques that are suitable for a very specific class of failures that can indeed be modeled as uncertainty regions around a nominal model. Any failure that does not significantly degrade the system or push the system outside the stability radius given by the robust controller will not compromise satisfactory stability and performance. However, any controller with a large enough stability radius to encompass most failure situations will likely be conservative and there is still no guarantee that unanticipated failures could be handled. Despite this, several investigators, including Birdwell et al. [14] and Veilltette et al. [13], have insisted that robust control theory can be used to maintain acceptable system stability and performance when control loops malfunction in a broad sense . Most investigators, however, agree that a reconfigurable controller will require additional loops and control structure. There are too many types of common failures, such as actuator and sensor malfunctions, which cannot be adequately modelled as uncertainty. These problems motivate the need for a controller that more directly addresses the situation.

2.2.3 Fault-Tolerant Control

The area of fault-tolerant controls has attracted the attention of investigators with increased frequency over the past few years. For example, Lane and Stengel [45] and Ochi and Kanai [46] pursued the use of feedback linearization and Gao and Antsaklis [47] described the use of pseudo-inverse methods. Adaptive control approaches using artificial neural networks are consider by Calise et al. [48], Idan et

al. [49], Johnson and Calise [50], and many others. While Huang and Stengel [51], Morse and Ossman [52] and Jiang [53] have made important contributions based upon model-following control principles.

2.2.4 Robust Fault-Tolerant Control

This area of reconfiguration has received only a minimal amount of attention. Wu [54] addresses the problem of performance robustness during normal system operation versus fault recovery. A prescribed performace level is optimized under a detection criterion relating the measurements to specific faults. The specific designer is largely free to choose one of a number of suitable FDI techniques. Jiang [53] discusses how fault-tolerance can be achieved using eigenstructure assignment design.

2.2.5 Robust Fault Estimation

The joint design of robust controllers and fault estimation often leads to complex interactions between the controller and fault estimator because the design freedom is utilized to solve both problems simultaneously [55, 56, 57, 58, 59]. This fundamental problem, however, is unavoidable as most studies in this area are based upon the idea that the robust controller optimization and fault estimation designs are best combined, for example, Tyler and Morari [56] use H_∞ optimization. An alternative way of performing open-loop FDI with a separate controller design avoids the difficulties of this design and provides much of the motivation for the next area.

2.2.6 Fault-Tolerant Control w/ FDI

The functions of FDI and reconfigurable control have been combined in a few notable studies [60, 53]. It is widely accepted that the FDI function along with redundant system design can prevent the development of more serious faults. When fault detection and isolation is carried out using the open-loop approach the controller affects the FDI robustness but not vice-versa [26, 61]. The controller's robustness issue becomes de-coupled from the FDI unit design and allows for an increased freedom in controller design and structuring. The main disadvantage with this de-coupling is the

adverse affects the detection delay has upon system stability as reported by Mariton [62]. The combination of the FDI system and control reconfiguration is a complex issue and one study by Srichander and Walker [63] proposes a stochastic approach to the stability analysis of some active fault tolerant control systems employing FDI schemes. The main assumption here is that behavior resulting from randomly occurring faults can be characterized dynamically by stochastic differential equations. The stochastic differential equations vary randomly in time and the equations can be analyzed using Markov theory. These stochastic approaches to robustness analysis are an emerging theoretical field in reconfigurable control.

2.2.7 Supervision

Different forms of selection logic and system management have been introduced into fault-tolerant systems by various investigators including Rauch [64], Buckely [65], Eryurek and Upadhyaya [66], and Polycarpou and Vemuri [67]. The function of supervision is essentially the active form of fault-tolerant control in which fault decision information is used to select the most suitable control function subsequent to the declaration that a fault has occurred. Also essential to the operation of the supervision system is the ability to determine whether a fault has determental effects on the system's performance and stability serious enough to warrant controller changes. Kwong et al. [68, 69] shows that the fuzzy model reference learning controller (FMRLC) can be used to reconfigure the nominal controller in an F-16 aircraft to compensate for various actuator failures without using explicit failure information. Kwong then developed an expert supervision strategy for the FMRLC that used only information about the time at which a failure occurs and showed that it achieved higher performance control reconfiguration than an unsupervised FMRLC. Fierro and Lewis [70] discuss a hybrid system framework which considers simultaneously the control and decision-making issues. A continuous-state plant is supervised by a discrete-event system which is based on a theory of linked finite state machines.

2.3 Types of Redundancy

As mentioned in the introduction, a reconfigurable controller should ideally be developed using a systematic and integrated approach to design. Most papers only consider problems which are based on mathematical models of the plant, but there are also many non-mathematical challenges which require attention at every stage and in all aspects of system design. Blanke et al. [15] paid attention to the development of the overall concept of systematic design. His study demonstrated that the development of a complete reconfiguration strategy requires an understanding of the structure of the system, the reliability of different components, the types of redundancy available and the types of controller function that are available or might be required. It is impossible to ensure control reconfiguration without redundancy in the initial system. Often the type and level of redundancy provided determines the way in which control reconfiguration is enacted. Hence, a failed sensor or actuator in systems with varying levels of redundancy will sometimes have dramatically different reconfiguration schemes to overcome the failure.

There are two forms of redundancy associated with reconfigurable controls. Direct redundancy is achieved by the use of multiple interdependent hardware channels and analytical redundancy is achieved by backing up available measurements using a mathematical model. Sometimes a combination of the two forms of redundancy is necessary. Making the best use of both the direct redundancy and the analytical redundancy provided by the system is a major task of reconfigurable control system design.

2.4 Fault-Tolerant Control Methods

Figure (2–3) shows the taxonomy of fault-tolerant control methods.

2.4.1 Passive Approaches

Passive approaches to fault-tolerance make use of robust control techniques to ensure that a closed-loop system remains insensitive to certain faults [43, 44]. The

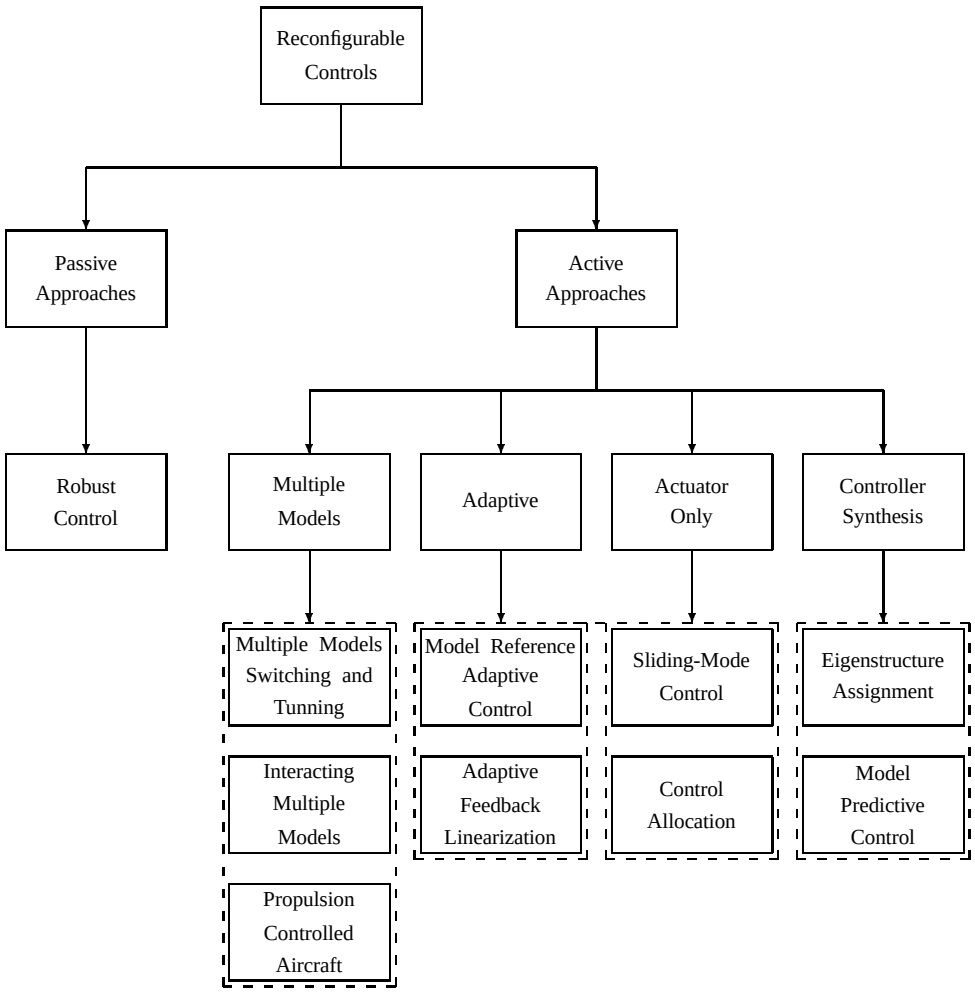

Figure 2–3: Taxonomy of Reconfigurable Control Methods

impaired system continues to operate with the same controller and system structure, i.e. the main objective it to recover the original system performance. Basically, the passive controller will reject the fault only if it can be de-sensitized to the fault's effects just as if it were a source of modelling uncertainty [43].

Among those who have extend their work on robust control to deal with passive fault-tolerance are Horowitz et al. [71] and Keating et al. [72] who used quantitative feedback theory, and McFarlane and Glover [73] and Williams and Hyde [74] who employed the frequency domain approach based on H_∞-norm optimization. Nett et al. [55], Tyler and Morari [56], and Murad et al. [57] present robust design approaches

to integrated control and fault estimation based upon the so-called four parameter controller. All of these passive fault-tolerant controllers are actually good examples of baseline controllers that can be used as a basis for further fault accommodation with active controllers. The original robustness is important during the detection and reconfiguration interval.

2.4.2 Active Approaches

In active fault-tolerance, a new control system is designed using the desirable properties of performance and robustness in the original system, but with the reduced capability of the impaired system in mind. Active fault-tolerance has this title because on-line fault accommodation is used. These methods differentiate themselves from passive approaches in that they take fault information explicitly into account and do not assume a static nominal model. In order to achieve reconfiguration or restructuring, an active fault-tolerant system requires either *a priori* knowledge of expected fault types or a mechanism for detecting and isolating unanticipated faults. This is essentially the function of a fault detection and isolation (FDI) scheme.

Active approaches are divided into two main types of methods: projection based methods and on-line automatic controller redesign methods [51]. The latter involves the calculation of new controller parameters in response to a control impairment, Gao and Antsaklis [47] referred to this method as reconfigurable control. In projection-based methods a new pre-computed control law is selected according to the type of malfunction that has been isolated [75]. Stengel [2] further classifies a reconfigurable or restructurable system whose feedback action is changed automatically as a special form of an intelligent control system. On-line restructuring or reconfiguration of control is a topic of ongoing research.

2.4.2.1 Multiple model control (MMC)

There are several areas of multiple model controls that have achieved notable success as fault-tolerant control techniques: Multiple Model Switching and Tuning

(MMST), Interacting Multiple Model (IMM), and Propulsion Controlled Aircraft (PCA). The idea of multiple model control has received increased interest in the last few years with Boskovic et al. [76, 77, 78, 79, 80, 81, 82, 83, 84], Kanev et al. [85, 86, 87], Demetriou [88], Zhang and Jiang [89], and Maybeck [90]. In MMST Boskovic et al. [76] describes the dynamics of each fault scenario by a model, then designs a controller for each fault scenario creating a massive parallel architecture. When a failure occurs, MMST switches to the pre-computed control law corresponding to the failure situation. The difficulty with this approach becomes one of choosing which model/controller pair to switch to at each time instant. In IMM Zhang and Rong Li [91] and Munir and Atherton [92] attempt to overcome this key limitation of MMST, rather than using the model which is closest to the current failure scenario, IMM computes a fault model as a convex combination of all pre-computed fault models and then uses this new model to make control decisions. Burken and Burcham [93] develops PCA which is a special case of MMST, where the only anticipated fault is total hydraulics failure and only the engines are used for control. The PCA problem was taken up by the NASA Dryden Flight Research Center [94, 95] in 1995 when they demonstrated successful landings after complete hydraulic failure using a MD-11 and a F-15 with propulsion-only control.

2.4.2.2 Control allocation (CA)

Control allocation (CA) is the technique of producing the desired set of forces and moments on an aircraft from a set of actuators. The purpose of control allocation is to allow the design of control laws which do not directly consider actuator failures. The output of the control law can be a set of desired forces and moments and the job of the allocator is to select appropriate actuator positions which will achieve the desired results. Bordignon and Durham [96] and Durham and Bordignon [97] addressed the problem of control allocation with magnitude and rate limits on the actuators, Davidson et al. [98] develops a control allocation technique for the extremely over-actuated

Innovative Control Effector (ICE) aircraft and Zhenyu et al. [99] looks at restoring as much of the performance of the original system as possible after a actuator failure.

2.4.2.3 Adaptive feedback linearization via artificial neural networks

This section examines a method primarily developed by Calise et al. [48, 49, 50, 100, 101, 102, 103] involving a model reference adaptive control scheme using adaptive feedback linearization with an artificial neural network to cancel inversion errors. The approach splits the dynamics of the plant into three single-input-single-output (SISO) subsystems for roll, pitch, and yaw. Each subsystem has a model reference adaptive controller. Brinker and Wise [104] and Wise et al. [103] have contributed by developing a control allocation technique that generates the desired roll, pitch, or yaw moment specified by the controller using the available control surfaces. Wise et al. [103] along with Calise [48, 102] have successfully demonstrated adaptive feedback linearization using artificial neural networks on the Tailless Advanced Fighter Aircraft (TAFA) and NASA's X-36.

2.4.2.4 Sliding mode control (SMC)

Shtessel et al. [105, 108, 106, 107] used Sliding Mode Control (SMC) to develop a robust controller that adaptively handles input magnitude and rate constraints. The proposed controller is set up in a two-loop configuration with the desired result of tracking a trajectory given by roll, pitch, and yaw angle. The outer-loop of the controller takes roll, pitch, and yaw and provides angular rate commands to the inner-loop, which is assumed to track the commands using the actuator inputs. There are two benefits of this controller. First, it can handle all failures which modify the dynamics of the plant less than the assumed uncertainty. Second, the on-line adaptation of the boundary layer can handle partial loss of actuator surfaces, while avoiding limits and integrator windup by reducing the tracking performance. The limitation of SMC is the assumption that the input function is square and invertible. This limitation requires that there must be one and only one control surface for every controlled variable and that

none of the control surfaces can ever be lost. Therefore, SMC is only applicable for failures which cause a loss of effectiveness of the control surface, unlike the floating or jammed surface failure scenarios.

2.4.2.5 Eigenstructure assignment (EA)

The concept of Eigenstructure Assignment (EA) was formally introduced by Andry et al. [109]. The idea behind the technique is to use state feedback to place the eigenvalues of a linear system then use the remaining degrees of freedom to align the eigenvectors as accurately as is possible. While the method for choosing appropriate eigenvectors and eigenvalues is not well-defined for aircraft, Davidson and Andrisani [110] highlighted the effects of the eigenstructure on flying qualities. Other researchers who propose EA for use in reconfigurable flight control systems are Konstantopoulos and Antsaklis [111], Belkharraz and Sobel [112], and Zhang and Jiang [113].

2.4.2.6 Model reference adaptive control (MRAC)

The goal of Adaptive Model-Following Control (MRAC) is to force the plant output to track a reference model. Although there are limitations of adaptive control for reconfiguration, Bodson and Groszkiewicz [114] and Groszkiewicz and Bodson [115] are attempting to apply it in slightly modified forms. First, a model structure must be assumed. The types of failures addressed in reconfigurable control, however, may well cause the plant structure to change drastically. Second, adaptive control requires that the system's states change slowly enough for the estimation algorithm to track them. However, faults may cause abrupt and drastic changes in the states moving the system instantaneously to a new region of the state space. As a result, adaptive control on its own is not enough to handle the general problem, but may well be an important part of the reconfigurable algorithm.

2.4.2.7 Model predictive control (MPC)

Model predictive control has been proposed as a method for reconfiguration due to its ability to handle constraints and changing model dynamics systematically. Maciejowsku [116] designed a MPC controller that has an intrinsic ability to handle jammed actuators without the need to explicitly model the failure. Failures can also be handled in a natural fashion by changing the internal model used to make prediction in either an adaptive fashion as done by Kanev and Verhaegen [86], a multi-model switching scheme as done by Boskovic and Mehra [76], or by assuming a FDI scheme that provides a fault model as done by Huzmezan and Maciejowski [117, 118]. The MPC approach to reconfiguration has achieved some notable successes but there are still fundamental issues that need to be examined. First, it is not clear how to adjust the weights in the cost function for an arbitrary fault model. Second, choosing performance targets is not a simple question. Finally, MPC requires an on-line optimization which makes it difficult to implement as an aircraft controller where the optimization must occur at high sampling rates and in fixed time. Also, there is no quarantee that there exists a solution to the optimization problem for all time.

CHAPTER 3
PRELIMINARIES

This chapter provides a review of the necessary technical background for an introductory investigation of reconfigurable flight controls. While is it assumed the reader has been exposed to these topics previously, a more in depth explanation of these concepts can be found in a number of undergraduate texts [119, 120, 121, 122].

3.1 Aircraft Flight Mechanics

The equations of motion for an aircraft in flight have changed little since their original formulation by Lanchester (1908) and Bryan (1911) [123]. The following sections identify the various reference frames used to describe an aircraft's state, provide an overview of the derivation of the general nonlinear equations of motion, and describe the small-disturbance theory linearization technique.

3.1.1 Aircraft Axis Systems

The motion of an aircraft can be described using many different axis systems. The three axis systems used here are the body-axis system fixed to the aircraft, the Earth-axis system, which we will assume to be an inertial axis system fixed to the Earth, and the stability-axis system, which is defined with respect to the relative wind. Each of these systems is useful in that they provide a convenient system for defining a particular vector such as an aerodynamic force vector, the weight vector, or the thrust vector.

3.1.1.1 Body-Axis System

The body-axis system, (b_1, b_2, b_3) in Figure (3–1), is fixed to the aircraft with its origin at the aircraft's center of gravity. The b_1 axis is defined out the nose of the aircraft, the b_2 axis is defined out the right wing of the aircraft, and the b_3 axis is

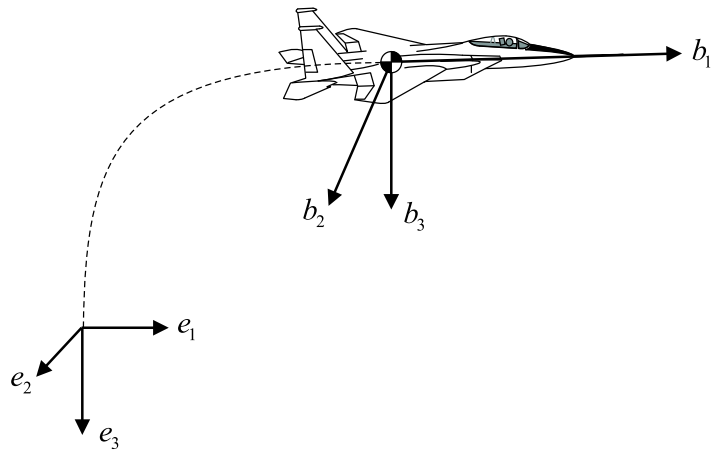

Figure 3–1: Relationship between Earth axis system and body axis system

defined down out of the bottom of the aircraft. These three axes form a traditional right-handed orthogonal reference system.

3.1.1.2 Earth-Axis System

The Earth-axis system, (e_1, e_2, e_3) in Figure (3–1), is fixed to the Earth with its e_3 axis pointing to the center of the Earth. Often, the e_1 axis is defined as North and the e_2 axis is defined as East. The Earth-axis system is assumed to be an inertial axis system for which Newton's laws of motion are valid. While this assumption is not totally accurate, it works well for most aircraft problems where the aircraft is traveling up to supersonic but not hypersonic speeds.

3.1.1.3 Stability-Axis System

The stability-axis system, (s_1, s_2, s_3), is rotated relative to the body axis system through the angle-of-attack and is used to study small deviations from a nominal flight condition. The origin of the stability-axis is also at the aircraft center of gravity. The s_1 axis points in the direction of the projection of the true airspeed onto the xz plane of the aircraft. The s_2 axis is out the right wing while the s_3 axis is orthogonal and points in accordance with the right-hand rule.

3.1.2 General Equations of Motion

The goal in this section is to develop the equations of motion which describe the position and orientation of the aircraft in appropriate reference frames. This development is essential in understanding how an aircraft behaves as well as the dynamics and relationships between the various reference frames. The velocity of the aircraft with respect to the body-axis system is given by

$$V_B = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{3.1}$$

The velocity V_B does not include the effects of wind. Any vector in the Earth-axis system can be transformed into the body-axis system using the following transformation (note the notation used for trigonometric functions, $S_\phi \equiv \sin\phi$, $C_\phi \equiv \cos\phi$, $T_\phi \equiv \tan\phi$, etc.)

$$l_{EB} = \begin{bmatrix} C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi \\ -S_\theta & S_\phi C_\theta & C_\phi C_\theta \end{bmatrix} \tag{3.2}$$

where ψ, θ, ϕ describe the orientation of the aircraft in the Earth-axis system. The angle of attach, α, and sideslip, β, can be defined in terms of the velocity components of the body axis system. The equations for α and β are defined by

$$\begin{aligned} \alpha &= \tan^{-1}\frac{w}{u} \\ \beta &= \sin^{-1}\frac{v}{\sqrt{u^2+v^2+w^2}} \end{aligned} \tag{3.3}$$

The position of the aircraft is most often used for navigation; therefore, its dynamics are given in the Earth-axis system as follows

$$r_E = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{3.4}$$

Remember that z points toward the ground and is therefore negative for positive height. As a result, the altitude or height is often used instead of z to describe the location of the aircraft while in flight. The aircraft height is given by

$$h = -z \tag{3.5}$$

The orientation of the aircraft is defined relative to the Earth-axis and is given by the Euler angles (ψ, θ, ϕ). The Euler angles define the rotations from the Earth-axis system to the body axis system. The ordering of the rotations is important and is done according to a 3-2-1 Euler angle sequence. If the sequence is performed in a different order other than ψ, θ, and ϕ, the final result will be incorrect. The accepted limits on the Euler angles are

$$\begin{aligned} 0^\circ \leq \psi \leq 360^\circ \\ -90^\circ \leq \theta \leq 90^\circ \\ -180^\circ \leq \phi \leq 180^\circ \end{aligned} \tag{3.6}$$

The angular rotation rates are defined relative to the body axis system as,

$$\omega_B = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.7}$$

where p is the roll rate, q is the pitch rate, and r is the yaw rate. The angular rates are related to the rate of change of the Euler angles by the following coordinate

transformations,

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & S_\phi C_\theta \\ 0 & -S_\phi & C_\phi C_\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$
$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi S_\theta^{-1} & C_\phi S_\theta^{-1} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.8}$$

Note that when perturbations are small, such that (ϕ, θ, ψ) may be treated as small angles, that is, $< 15°$, then Equations (3.8) can be approximated as

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} \approx \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{3.9}$$

The dynamics are derived from Newton's 2nd Law which states that the summation of the external forces acting on a body is equal to the time rate of change of the momentum of the body; and the summation of the external moments acting on the body is equal to the time rate of change of the moment of momentum (angular momentum). The force equation is given by

$$F = m\left(\frac{dV_c}{dt} + (\omega \times V_c)\right) \tag{3.10}$$

and the moment equation as

$$M = \frac{d\,(I\omega)}{dt} + (\omega \times (I\omega)) \tag{3.11}$$

where V_c is the velocity of the center of mass of the aircraft, ω is the angular velocity and I is the moment of inertia tensor. The force vector which consists of the

aerodynamic forces and thrust forces acting on the aircraft is given by

$$F = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{3.12}$$

The equations can now be written in terms of the variables defined in this section. The three force equations in the body-axis system are

$$\begin{aligned} X - mgS_\theta &= m\left(\dot{u} + qw - rv\right) \\ Y + mgC_\theta S_\theta &= m\left(\dot{v} + ru - pw\right) \\ Z + mgC_\theta C_\phi &= m\left(\dot{w} + pv - qu\right) \end{aligned} \tag{3.13}$$

where ϕ_T is the angle between the x-body direction and the thrust vector, T. Assuming that the mass distribution of the aircraft is constant, such as neglecting fuel slosh and fuel burn, the moments and products of inertia do not change with time. The three moment equations in the body-axis system are

$$\begin{aligned} L &= I_{xx}\dot{p} - I_{xz}\dot{r} + qr\left(I_{zz} - I_{yy}\right) - I_{xz}pq \\ M &= I_{yy}\dot{q} + rp\left(I_{xx} - I_{zz}\right) + I_{xz}\left(p^2 - r^2\right) \\ N &= -I_{xz}\dot{p} + I_{zz}\dot{r} + pq\left(I_{yy} - I_{xx}\right) + I_{xz}qr. \end{aligned} \tag{3.14}$$

where $[L, M, N]$ are the rolling moment, pitching moment, and yawing moment acting of the aircraft, respectively. These applied moments consist of aerodynamic and thrust moments acting on the aircraft. The forces and moments are functions of the control surfaces, thrust, and aerodynamics of the aircraft and can be written as functions of the six linear and angular velocities (u, v, w, p, q, r) and the actuator positions.

3.1.2.1 Longitudinal and Lateral-Directional Equations of Motion

The six aircraft equations of motion, (3.13)-(3.14), can be decoupled into two sets of three equations. These are the three longitudinal equations of motion and the three

lateral-directional equations of motion. This is convenient in that for many flight conditions only three equations need to be solved simultaneously. The three longitudinal equations of motion consist of the x force, y moment, and z force equations

$$\begin{aligned} X &= m\left(\dot{u} + qw - rv\right) + mgS_\theta \\ M &= I_{yy}\dot{q} + rp\left(I_{xx} - I_{zz}\right) + I_{xz}\left(p^2 - r^2\right) \\ Z &= m\left(\dot{w} + pv - qu\right) - mgC_\theta C_\phi \end{aligned} \tag{3.15}$$

The lateral-directional equations of motion consist of the x moment, y force, and z moment equations

$$\begin{aligned} L &= I_{xx}\dot{p} - I_{xz}\dot{r} + qr\left(I_{zz} - I_{yy}\right) - I_{xz}pq \\ Y &= m\left(\dot{v} + ru - pw\right) - mgC_\theta S_\theta \\ N &= -I_{xz}\dot{p} + I_{zz}\dot{r} + pq\left(I_{yy} - I_{xx}\right) + I_{xz}qr \end{aligned} \tag{3.16}$$

In addition to the six force and moment equations of motion, Equation (3.8) is required to completely solve the aircraft problem because there are more than six unknowns due to the presence of the Euler angles in the force equations. Recall, the three kinematic equations

$$\begin{aligned} p &= -S_\theta\dot{\psi} + \dot{\phi} \\ q &= S_\phi C_\theta\dot{\psi} + C_\phi\dot{\theta} \\ r &= C_\phi C_\theta\dot{\psi} - S_\phi\dot{\theta} \end{aligned} \tag{3.17}$$

3.1.3 Linearized Equations of Motion

The nine aircraft equations of motion, (3.15)-(3.17), are nonlinear differential equations. They can be solved with various numerical integration techniques to obtain time histories of motion variables, but it is nearly impossible to obtain closed form solutions. It is assumed that the motion of the aircraft consists of small deviations from a reference condition of steady flight; therefore, the small perturbation approach can be

used to linearize the equations of motion and develop the closed form solutions around trim conditions. Steady flight can be defined, for example, as one of the following:

steady wings-level flight	$\phi = \dot{\phi} = \dot{\theta} = \dot{\psi} = 0$
steady turning flight	$\dot{\phi} = \dot{\theta} = 0$, $\dot{\psi} =$ turn rate
steady pull-up	$\phi = \dot{\phi} = \dot{\psi} = 0$, $\dot{\theta} =$ pull-up rate
steady roll	$\dot{\theta} = \dot{\psi} = 0$, $\dot{\phi} =$ roll rate,

where $\dot{p} = \dot{q} = \dot{r} = \dot{V} = \dot{\alpha} = \dot{\beta} = 0$ and all control surface inputs are zero. There are three possible methods for computing a linear model for small perturbations around the trim or steady-state condition. The first is to replace any nonlinearities in the general dynamics equation with their first order Taylor series approximations. The second is to run an identification algorithm using data collected from either a nonlinear model or the physical system. The final method, and one used throughout this work, is to numerically compute the effects of small changes in state variables and inputs on the state derivatives. This can be done, for example, by using a Matlab linearization routine such as *linmod* on a nonlinear Simulink model. The benefits of the linearization is that we can write the physical system in a convenient matrix form

$$\dot{x} = Ax + Bu \tag{3.18}$$

where $x \in R^n$ is the state variables and $u \in R^p$ is the control inputs. The system output $y \in R^q$ is given by

$$y = Cx + Du \tag{3.19}$$

The state, control, and output vectors are defined as follows

$$x = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} \qquad \text{state vector } (n \times 1) \tag{3.20}$$

$$u = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_p(t) \end{bmatrix} \qquad \text{input vector } (p \times 1) \tag{3.21}$$

$$y = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_q(t) \end{bmatrix} \qquad \text{output vector } (q \times 1) \tag{3.22}$$

The matrices A, B, C are constant matrices and defined as follows

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \qquad \text{plant matrix } (n \times n) \tag{3.23}$$

$$B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} \qquad \text{control matrix } (n \times p) \tag{3.24}$$

$$C = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qn} \end{bmatrix} \qquad \text{output matrix } (q \times n) \tag{3.25}$$

where A is the plant matrix, B is the control matrix, and C is the output matrix. For the aircraft considered throughout this work the D matrix is the null matrix.

3.2 Neural Networks

This secion is intended to review and help the reader understand what artificial neural networks are, how they work, and where they are currently being used. The intent is to give a non-technical introduction; therefore, it does not go into depth with mathematical formulas. A more detailed explanation is provided in [121, 122].

3.2.1 Artificial Neural Networks

An Artificial Neural Network is a system loosely modeled on the human brain. While neural networks do not approach the complexity of the brain, they are an an attempt to simulate within specialized hardware or sophisticated software the multiple layers of simple processing elements called neurons. Each neuron is linked to a certain number of its neighbors with varying coefficients of connectivity that represent the strengths of these connections. Learning is accomplished by adjusting these strengths to cause the overall network to output appropriate results.

3.2.1.1 The Biological Neuron

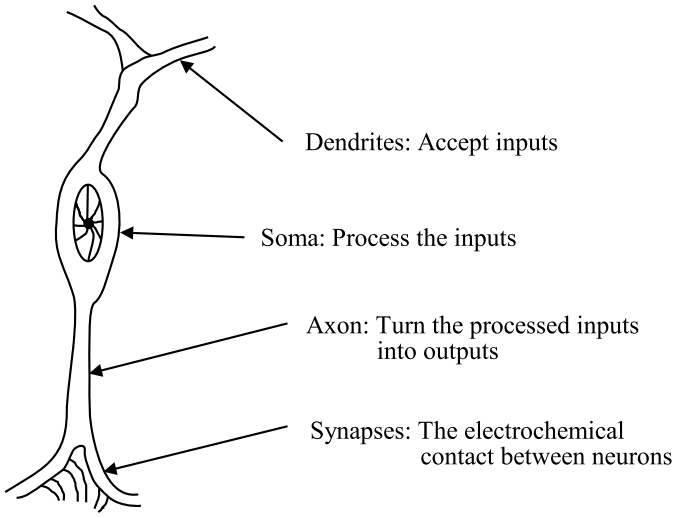

Figure 3–2: A biological neuron

The most basic components of neural networks are modeled after the structure of the brain; therefore, a great deal of the terminology is borrowed from neuroscience. The neuron is the most basic element of the human brain and provides us with the

abilities to remember, think, and apply previous experiences to our every action. The power of the brain comes from the large number of neurons (approximately 10^{11}) and the multiple connections between them (up to 200000). All natural neurons have four basic components, Figure (3–2), which are dendrites, soma, axon, and synapses. Basically, a biological neuron receives inputs from other sources, combines them in some way, performs a generally nonlinear operation on the result, and then outputs the final result.

3.2.1.2 The Artificial Neuron

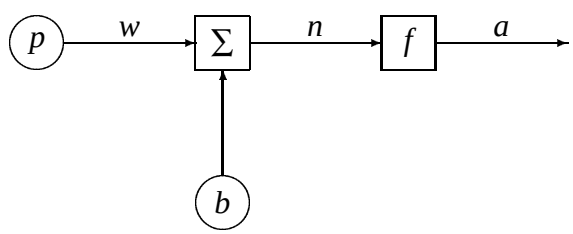

Figure 3–3: An artificial neuron

The basic unit of artificial neural networks, the artificial neuron, simulates the four basic functions of natural neurons. That various inputs to the network, p, are multiplied by a connection weight, w, these products are simply summed with a bias, b, then fed through a transfer function, f, to generate a result, a. Referring to Figure (3–3), an artificial neuron output is given by

$$a = f(wp + b) \tag{3.26}$$

Even though all artificial neural networks are constructed from this basic building block their architectures and applications are extremely diverse.

3.2.2 Design

Designing a neural network consists of four important steps: arranging neurons in various layers, deciding the type of connections among neurons within a layer as well as among those in different layers, deciding the way a neuron receives input and

produces output, and determining the strength of connection within the network by allowing the network to learn the appropriate values. The process of designing a neural network is an iterative one.

3.2.2.1 Layers

Biological neural networks are constructed in a three dimensional way from microscopic components. Artificial neural networks are the simple layering of artificial neurons, which are then connected to one another. All artificial neural networks have a similar structure of topology, Figure (3–4). Some of the neurons receive input, the input layer, and other neurons provide the network's outputs, the output layer. All the rest of the neurons are hidden from view, the hidden layer.

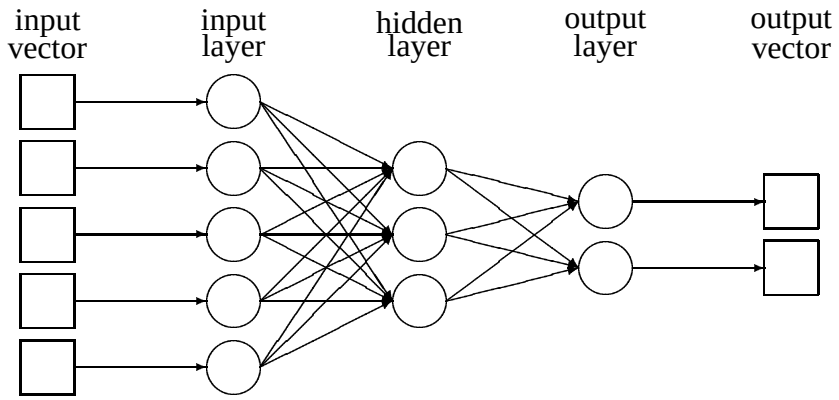

Figure 3–4: An artificial neural network

When the input layer receives the input, its neurons produce output, which becomes input to the other layers of the system, the process continues until the output layer is reached and information is passed to the output vector. The determination of the number of hidden neurons the network should have in order to perform its best is often a process of trial and error. If the number of hidden neurons is increased too much, the network will memorize the training set and will have problems in generalization.

3.2.2.2 Learning

The brain basically learns from experience, this is also true for artificial neutral networks. Learning typically occurs through training or exposure to a truthed set of input/output data where the training algorithm iteratively adjusts the connection weights. For this reason, artificial neural networks are sometimes called machine learning algorithms. The learning ability of a neural network is determined by its architecture and by the algorithm chosen for training. The training method usually consists of one of three schemes:

- Unsupervised learning: The hidden neurons must find a way to organize themselves without help from the external environment. In this approach, there are no target outputs available for the network to measure its predictive performance for a given vector of inputs.
- Reinforcement learning: Reinforced learning is also called supervised learning. The connections among the neurons in the hidden layer are randomly arranged, then reshuffled as the network is told how close it is to solving the problem. Instead of begin provided with the correct output for each network input, reinforced learning only gives a grade. The grade is given by a teacher. The teacher may be a training set of data or an observer who grades the performance of the network results.
- Backpropagation: This method has proved highly successful in training of multilayered neural nets. The network is not just given reinforcement for how it is doing on a task. Information about errors is also filtered back through the system and is used to adjust the connections between the layers, thus improving performance. A form of backpropagation is used in this work.

One can categorize the learning methods into yet another group: off-line or on-line.

- Off-line: In the off-line learning methods, once the systems enters into the operation mode, its weights are fixed and do not change any more. Most of the

current networks are of the off-line learning type. Off-line learning is used in this work.

- On-line: In on-line or real time learning, when the system is in operating mode, it continues to learn while being used as a decision tool. This type of learning has a more complex design structure.

3.2.3 Areas of Applications

The first practical application of artificial neural networks came in the late 1950s, with the invention of the perceptron network and associated learning rule by Frank Rosenblatt [121]. Rosenblatt and his colleagues built a network and demonstrated its ability to perform pattern recognition. Today, neural networks are performing successfully in a wide variety of problems including interpretation, prediction, diagnosis, planing, monitoring, debugging, repair, instruction, and control. Basically, most applications of neural networks fall into the following five categories: prediction, classification, data association, data conceptualization, and data filtering.

CHAPTER 4
FAULT DETECTION AND ISOLATION

The purpose of this chapter is to develop an artificial neural network that can be used in fault detection and isolation (FDI) of fixed-position actuator failures. The failure class is briefly reviewed and defined mathematically then the procedure to develop an artificial neural network is outlined.

4.1 Failure Parameterization

A flight control system is working properly if all the control effectors, i.e., leading edge flaps, trailing edge flaps, ailerons, stabilators, and rudders maintain the state variables in the neighborhood of their desired values. A fault occurs when a certain level of deterioration takes place in one or more state variable because of permanent physical change, i.e., jammed or hard-over control surfaces. System failure occurs when a fault or combination of faults lead to complete system deterioration and a sudden termination of flight control. Faults may produce only poor or reduced performance, but may also lead to catastrophic failure including loss of aircraft and crew.

This work focuses on the case when a control surfaces freezes in a fixed-position and does not respond to subsequent commands. This is an especially difficult type of failure to overcome since the remaining actuators should be reconfigured not only to achieve the control objective, but also to compensate for a disturbance due to the failure. We assume that the failure is unknown but can be determined from the feature history of state variables and position commands to the actuators. The failure introduces a constant disturbances into the overall closed-loop system so that the solution to the new control problem is far from trivial.

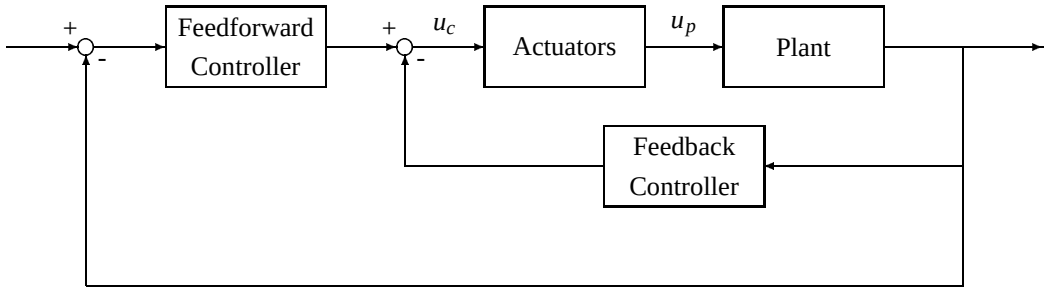

Figure 4–1: The structure of a full-state feedback control system

The hope is that a positive identification of the failed control surface and fixed-position will facilitate the development of a revised control law to stabilize the failed aircraft and reinstate optimal maneuvering performance. Referring to Figure (4–1), let u_c describes the signal generated by the controller and u_p the signal that enters the actual plant through the control surfaces. The reason for making a distinction between u_c and u_p is that control surface jamming is manifested by u_p assuming a constant value even while u_c varies with time. For simplicity, assume that in the case with no failures $u_c(t) = u_p(t)$. A fixed-position actuator failure is define as

$$u_p(t) = \begin{cases} u_c(t) & \text{if } t < t_f \\ w & \text{if } t \geq t_f \end{cases}$$

where t_f denotes the failure instant of the control surface and w is the value at which the control surface has frozen. The proposed neural network will monitor a signal consisting of state measurements and control surface position commands, u_c, to determine if the aircraft is operating properly, $u_c(t) = u_p(t)$, or has suffered a control surface failure, $u_c(t) \neq u_p(t)$. When a control surface has failed the neural network will hopefully identify which control surface has failed and at what position the failure has occurred, w.

4.2 FDI via Artificial Neural Networks

4.2.1 Artificial Neural Network FDI Formulation

Traditional methods for FDI tend to employ mathematical state-space models of the monitored system such as state observers and Kalman filters, which continually estimate predicted state measurements. Fault detection is achieved in these methods by comparing predicted to actual state measurements. The pivotal assumption with these techniques is that the state-space model of the system is known positively. In reality, state-space models are good assumptions at best. Being based on a mathematical model, they can be very sensitive to modeling errors, parameter variations, noise, disturbances, etc. For example, modeling errors can be interpreted as a fault, thus producing false alarms, or hinder actual system degradation from being detected in the first place. A mathematical model is simply a description of system behavior and accurate modeling for a complex nonlinear system is very difficult to achieve in practice even when the analytical equations of motion are known. For this reason, fault detection and isolation by these methods is an imprecise science and has shown to be quite difficult over the past 20 years. Hence, the development of a robust and less model dependent method for fault detection and diagnosis for complex nonlinear system is warranted. Artificial neural networks are an ideal solution to this problem since they demonstrate several desired advantages: powerful nonlinear mapping properties, noise tolerance, self-learning and parallel processing capabilities.

Artificial neutral networks have been proposed as solutions for a wide variety of tasks. Among the most promising applications is that of pattern classification. Pattern classification implies observing input data with the intend of recognizing specific traits. This classification process facilitates the initiation of certain actions based on the input data. The inputs representing a pattern are called the measurement or feature vector. In fault diagnosis, the different types of faults occurring in the system may be viewed as decision classes. The function preformed by a pattern recognizing neural network is

the mapping of the input feature vector into one of the various decision classes. Fault detection and diagnosis can, therefore, be considered a pattern classification activity, and, thus, the potential exists for fault detection and diagnosis using an artificial neural network.

4.2.2 Artificial Neural Network Development

Neural networks are excellent mathematical tools for dealing with nonlinear problems, as they are designed to learn patterns of activities. A nonlinear system can be approximated by a neural network given suitable weighting factors and an architecture consisting of at least one hidden layer. The system model can be extracted from historical training data using a learning algorithm that often requires little or no *a priori* knowledge about the system. Learning is just determining the proper connection strengths to allow the outputs nodes to achieve the correct target output for a given feature vector. This adaptive nature of neural networks provides great flexibility for modeling nonlinear systems by allowing the weights to be learned by experience, thus producing a self-learning system. The default performance function for many feedforward neural networks is the mean squared error, which is the average squared error between the network outputs and the target outputs. A backpropagation training algorithm is used to train the network. Learning proceeds by updating the network weights in the direction in which the performance function decreases most rapidly, the negative of the gradient. One iteration of the backpropagation algorithm can be written

$$x_{k+1} = x_k + \alpha_k g_k$$

where x_k is a vector of current weights, g_k is the current gradient, and α_k is the learning rate. Learning begins by initializing all the connection strengths to small randomly selected values. Then a training pattern of feature vectors composed of simulated flight data is introduced into the input nodes of the network.

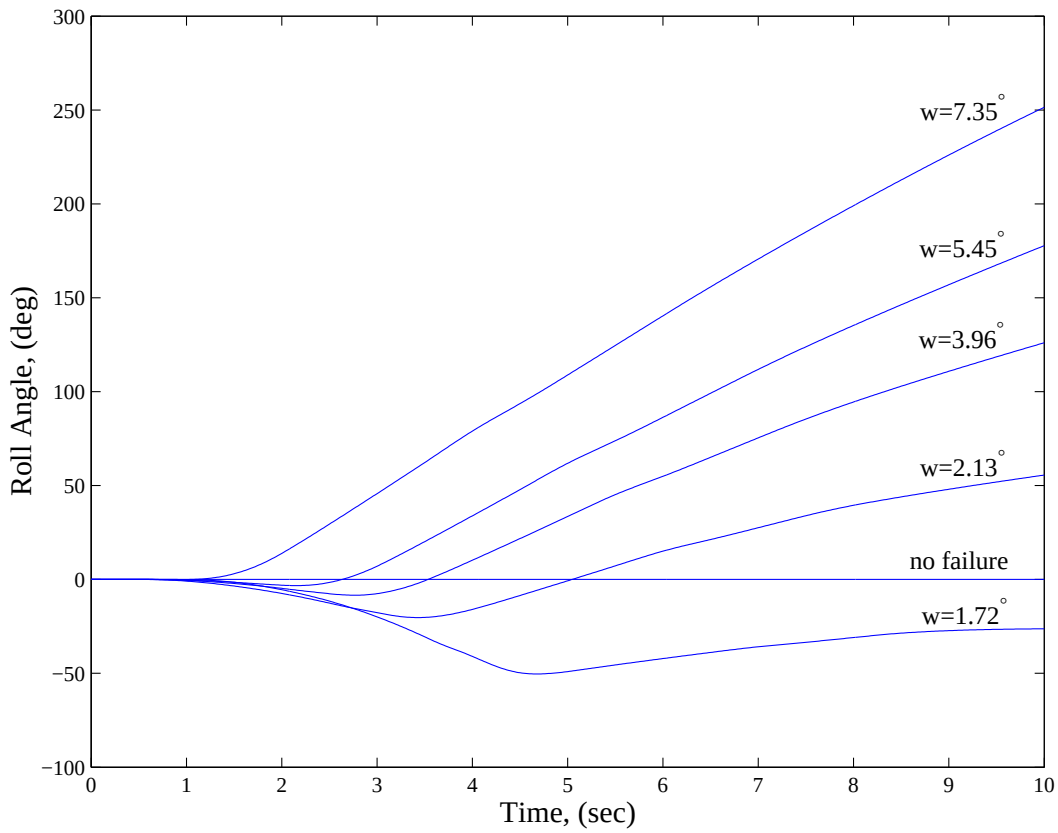

Figure 4–2: Roll feature vectors for different failure position *w*

The training data consists of roll angle, pitch angle, yaw angle, roll rate, pitch rate, yaw rate, angle of attack, sideslip, and position commands to the control surfaces for the unfailed case and a characteristic set of simulated failures for a given pilot command. Figure (4–2) shows the roll angle feature history for five different failures during a pitch doublet, the roll angle is zero for the unfailed case. The failures occurred on the left leading-edge flap at postions denoted by *w*. As can be easily seen by Figure (4–2) small changes in the failure position produce drastic changes in the evolution of the roll angle of the aircraft. These results constitute historically rich flight data to design a neural network for FDI. Once all these feature vectors are gathered the input vectors are then propagated in a feed-forward fashion through the network to produce output values. The outputs were compared to the desired target outputs to produced a mean squared error signal. The connection strengths are then systematically

adjusted by the learning algorithm to reduced the mean squared error to a desired value. After each training cycle, the neural network will know more about the system dynamic behavior. Once the connection strength is properly determined to undershoot the desired mean squared error, training is stopped and the neural network is ready for fault detection and isolation. The output vector consists of two variables, namely a surface identifier and a position indicator, i.e., the output vector $[0;0]$ corresponds to the absence of any particular fault and $[n;w]$ corresponds to the nth control surface jammed at w.

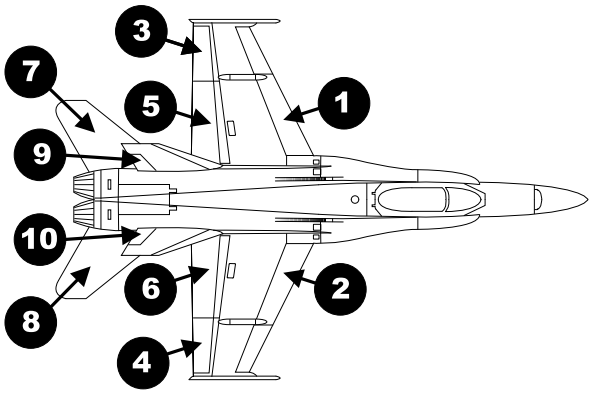

Figure 4–3: F/A-18 flight control surface numbering scheme

The proposed neural network has the ability to detect a specific fault, control surface and fixed-position, using pattern recognition techniques that activate an alarm in the form of the output vector. It therefore acts as a pattern recognizer for the detection of specific faults and classifies the faults accordingly. Figure (4–3) shows the numbering scheme for the control surfaces of an F/A-18 aircraft and Table (4–1) lists the position limits for each control surface, the maximum and minimum values are the hardover position.

After the network is trained, fault detection and diagnosis is simply a matter of presenting a new historically rich feature vector to the input nodes and reading the output vector from the output nodes. The neural network can be tested by simulating new flight data to produce a new input vector not in the original training pattern.

Table 4–1: F/A-18 Control Surface Position Limits

#	Effector	Position Limits
1,2	Leading Edge Flaps (lef)	$-3 \leq \delta_{lef} \leq 33$
3,4	Trailing Edge Flaps (tef)	$-8 \leq \delta_{tef} \leq 45$
5,6	Ailerons (ail)	$-25 \leq \delta_{ail} \leq 45$
7,8	Stabilators (stb)	$-24 \leq \delta_{hzt} \leq 10.5$
9,10	Rudders (rud)	$-30 \leq \delta_{rud} \leq 30$

On-line fault detection and diagnosis can be achieved by using a system of delays to produce the historical data necessary for the neural network to perform its calculation properly.

CHAPTER 5
THE STABILIZATION PROBLEM

This chapter presents the formulation of the stabilization solution which is one of the most time-critical components of a reconfiguration scheme. A fixed-position actuator failure can create a constant force and moment disturbance on the aircraft. This constant force and moment can lead to a significant deviation from the desired trim condition and leave the remaining control surfaces incapable of regaining control of the aircraft. There are several suitable methods for returning an aircraft to a trim condition following a failure which include the use of a regulator system with integral control or, if the disturbance can be measured, in a linear trim subsystem through the use of feedforward control. This work will focus on the development of the latter approach which has the distinct advantage of rapid response to disturbances while not adversely affecting the stability of the system. Its disadvantage is that any errors between the approximated disturbance received through fault detection and isolation and true disturbance will directly appear in the output. The following sections present a formal description of the stabilization problem and describe a decomposition of the problem which allows the use of a fast and efficient Matlab algorithm in the solution [124, 125].

5.1 The Nonlinear Trim Problem

During normal flight, the motion of an aircraft with respect to the Earth-axis system can, in general, be described by the nonlinear autonomous differential equations

$$\begin{aligned} \dot{x}(t) &= f_o\left(x(t), u(t)\right) + \zeta \\ y(t) &= h_o\left(x(t), u(t)\right) \end{aligned} \tag{5.1}$$

where $f_o : R^{n\times p} \mapsto R^n$ and $h_o : R^{n\times p} \mapsto R^q$ are nonlinear mappings, $x(t) \in R^n$ is the state variables vector, $u(t) \in R^p$ is the input vector, $\zeta(t)$ is a vector of unmeasurable disturbances, and $y(t) \in R^q$ is the output vector. The orientation of the aircraft is trimmed at the nominal values (x_n, u_n) when

$$\begin{aligned} f_o\left(x_n, u_n\right) &= 0 \\ h_o\left(x_n, u_n\right) &= 0 \end{aligned} \tag{5.2}$$

During straight and level flight the nominal control settings u_n are established which maintain steady state flight $(\dot{x} = 0)$ with wings level at constant altitude, airspeed, and heading. Following a fixed-position actuator failure, the aircraft dynamics are assumed to satisfy

$$\begin{aligned} \dot{x}(t) &= f_o\left(x(t), u(t)\right) + \zeta(t) + d \\ y(t) &= h_o\left(x(t), u(t)\right) \end{aligned} \tag{5.3}$$

where d is a constant or slowly varying measurable disturbance vector. For our failure class, d respresents the constant disturbance that results from the nonzero deflection of the failed control surface. Following a fixed-position actuator failure, a trim condition results when

$$\begin{aligned} f_o\left(x_n, u_n\right) + d &= 0 \\ h_o\left(x_n, u_n\right) &= 0 \end{aligned} \tag{5.4}$$

The problem then becomes how to determine a solution (x_n, u_n) which satisfies Equation (5.4). The solution can be determined by using the Matlab trimming routine *trim* on a nonlinear model of the aircraft's equations of motion at a desired flight condition. It is only necessary to develop the proper constraints on the magnitudes of the control surfaces and states to produce a feasible solution for implementation.

5.2 The Linear Trim Problem

Let the open-loop linearized dynamics of the healthy aircraft be described as

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{5.5}$$

where $x(t) \in R^n$ is the state variables vector, $u(t) \in R^p$ is the control vector, $A \in R^{n\times n}$ is the plant matrix, and $B \in R^{n\times p}$ is the control matrix. Let the measurements be given by

$$y(t) = Cx(t) + Du(t) \tag{5.6}$$

where $y(t) \in R^q$ is the output vector, $C \in R^{q\times n}$ is the output matrix, and $D \in R^{q\times p}$ is the null matrix. Assume that a postfailure model of an aircraft at a chosen flight condition is given by

$$\dot{x}(t) = Ax(t) + B_r u_r(t) + d \tag{5.7}$$

where $x(t)$ is the state vector of the linear aircraft dynamics and $u_r(t)$ is the vector of available (i.e., failed surface is deleted) control surfaces deflections, and d is a vector of constant disturbances that can be used to represent forces and moments generated by a failed surface. For the general problem, the disturbance vector d may be measured (e.g., by the use of an FDI algorithm) or unmeasurable. Let the key quantities that are to be regulated in denoted by

$$y(t) = Cx(t) + Du_r(t) \tag{5.8}$$

Elements of $y(t)$ might represent quantities such as altitude, bank angle, flight path angle, and rotational rate perturbation. The objective of our problem will be to automatically select $u_r(t)$ to guarantee that $y(t)$ achieves some desired value in steady state, $y_d(t)$. More precisely, the linear trim objective can be expressed as finding the solution (x_n, u_n) that guarantees

$$y(t) = y_d(t) \tag{5.9}$$

and

$$0 = Ax_n + Bu_n + d \tag{5.10}$$

For this work, we will assume that the disturbance d is caused by a fixed-position actuator failure which can be measured through FDI. We will further assume that the disturbance takes the form

$$d = b_w w \tag{5.11}$$

where w is the difference between the jammed position of the failed control surface and its nominal value, and b_w is the column removed from the B matrix corresponding to the failed control surface. Now define the model of an aircraft with a fixed-position actuator failure as

$$\dot{x}(t) = Ax(t) + B_r u_r(t) + b_w w \tag{5.12}$$

where $x(t) \in R^n$ is the state vector, $u_r(t) \in R^{p-1}$ is the vector of remaining control surfaces (i.e., failed surface is deleted), $b_w w$ is the input to the aircraft caused by the jammed surface w, and b_w is the column in B corresponding to the jammed surface. As with the nonlinear trim formulation, it is necessary to impose some constraints on the allowable magnitudes of the states and control surfaces, (x,u), for which a solution should be feasible. These constraints can be described as upper and lower limits on the allowable perturbation,

$$\begin{aligned} x_L \leq x \leq x_U \\ u_L \leq u \leq u_U \end{aligned} \tag{5.13}$$

Equation (5.9) through (5.13) describe the main objectives of the linear trim problem. That is, produce a control system that achieves stable flight at constant altitude with certain specified states set to zero, while the deviation of all other states is minimized from their desired values. Various norms and weighting matrices can be used to determine the solution to this problem, the Matlab routine *trim* is demonstrated in the final chapter.

CHAPTER 6
FAULT-TOLERANT CONTROL DESIGN METHODS

The objective of this work is to develop a reconfiguration scheme that is reliable and offers a degree of assured success for the targeted types of failures. The reconfiguration scheme is expected to stabilized the aircraft in the event of a control surface failure and provide reasonable command-tracking performance.

To accomplish these goals, two approaches are investigated and evaluated in this chapter. One is based on linear-quadratic regulator (LQR) methodology. In this approach the effect of the jammed surface is treated as a measurable constant disturbance to the system. An LQR controller is designed to stabilize the aircraft (stabilization), balance the jammed surface (disturbance rejection), and provide command tracking. The second method is developed by the author of this work, which is based on an H_∞ approach. Here the effect of the jammed surface is treated as a constant disturbance which is bounded by a low-pass filter. A reference nominal controller is designed for the healthy aircraft with all control surfaces operable. This nominal controller is used as a target model in H_∞ synthesis to design a robust controller which is capable of canceling the influence of the jammed surfaces and reproduce as closely as possible the desired outputs of the healthy aircraft.

The problem formulation for the targeted type of failure is presented before each fault-tolerant control method is developed in the following sections. Let the open-loop linearized dynamics of the healthy aircraft be described in state variable form as

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{6.1}$$

where $x(t) \in R^n$ is the vector of aircraft states and $u(t) \in R^p$ is the vector of control surfaces. Let the measurements be given by

$$y(t) = Cx(t) + Du(t) \tag{6.2}$$

where $y(t) \in R^q$ is the output variables available for feedback control. It is assumed that a baseline control law has been designed based on (6.1) that provides satisfactory stabilization and command-tracking performance of the aircraft. Suppose now that one of the control surface actuators fails suddenly and jams at a position w. Let us rewrite the entire postfailure system in state space form as

$$\dot{x}(t) = Ax(t) + B_r u_r(t) + b_w w \tag{6.3}$$

where $x(t) \in R^n$ is the state vector, $u_r(t) \in R^{p-1}$ is the vector of remaining control surfaces (i.e., failed surface is deleted), $b_w w$ is the input to the aircraft caused by the jammed surface w, and b_w is the column in B corresponding to the jammed surface.

6.1 Fault-Tolerant Control Design Using LQR Theory

LQR design methodology can be applied directly to Equation (6.3) assuming that $b_w w$ is a constant disturbance that can be eliminated by using integral control. While that assumption is completely accurate and can produce desired results it is not the approach taken here. The approach taken here is based upon a systematic procedure in which the failure is identified through FDI, Chapter (4), and directly canceled by finding a new trim condition, Chapter (5). The result is a new linear system which directly considers the effects of the constant disturbance. This new linear system is achieved by considering the results of our automatic trim algorithm

$$0 = Ax_n + B_r u_n + b_w w \tag{6.4}$$

where $x_n \in R^n$ is the vector of nominal aircraft states and $u_n \in R^{p-1}$ is the vector of nominal control surface deflections such that the state derivatives are identically zero.

Note that the trim condition (x_n, u_n) is only available for calculation when w is known through some FDI procedure. By simply rearranging Equation (6.4) we can achieve an expression for the constant disturbance in terms of the state matrices and trim condition

$$b_w w = -\left(Ax_n + B_r u_n\right) \tag{6.5}$$

By substituting this result into Equation (6.3) we find our new linear system is given by

$$\dot{x}(t) = A\left(x(t) - x_n\right) + B_r\left(u_r(t) - u_n\right) \tag{6.6}$$

The fault-tolerant control problem can now be stated as follows using LQR methodology. Find the control $u_r(t) - u_n$ that minimizes

$$J = \int_0^\infty \left[\left(x - x_n\right)^T Q\left(x - x_n\right) + \left(u_r - u_n\right)^T R\left(u_r - u_n\right)\right] dt \tag{6.7}$$

The optimal control that minimizes (6.7) is given by

$$u_r(t) - u_n = -R^{-1}B^T P\left(x(t) - x_n\right) \triangleq -K\left(x(t) - x_n\right) \tag{6.8}$$

where P solves the algebraic Riccati equation

$$0 = A^T P + PA - QPBR^{-1}B^T P \tag{6.9}$$

Assuming that the linearized model is valid the feedback law (6.7) quarantees that the linearized closed loop system will be stable, and that the important states (6.2) will approach their desired trajectories regardless of the constant disturbance. Integral control can be added to the design process to minimize errors and improve tracking performance. Thus the primary goals of stabilizing the aircraft (stabilization), balancing the jammed surface (disturbance rejection), and providing command tracking will be met by this LQR fault-tolerant design.

6.2 Fault-Tolerant Control Design Using H_∞ Theory

This section develops the theory for aircraft tracking control for a class of aircraft failures using H_∞ control design methodology. The author uses a two-step process of first designing the feedforward part of the controller to achieve perfect trajectory following and then designing the feedback part of the controller using H_∞ regulator theory. The objective of the tracking problem is to get a plant output to track a desired model signal. The design procedure will attempt to exploit the H_∞-optimality criterion for judging tracking performance while minimizing the worst case tracking error norm over an admissible ball of disturbances. The resulting controller design is an innovative technique for fault-tolerant controls in which H_∞ methodology is applied directly to the targeted failure class.

The desired model signal to be tracked is given by a reference LQR controller design for the healthy aircraft (see Appendix B for controller design) shown in Figure (6–1)

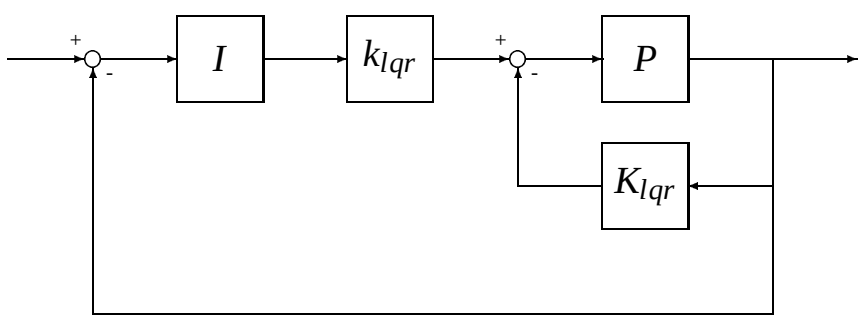

Figure 6–1: Reference closed-loop system

where P is the linearized F/A-18 model (see Appendix A), K_{lqr} is the feedback gains, k_{lqr} is the feedforward gains, and I is an integrator. The reference controller as shown in Figure (6–1) presents some difficulties in our design process, since, in general, H_∞ design frameworks do not consider integral control. The problem is that H_∞ control theory cannot be applied directly to a system that is neutrally-stable. H_∞ synthesis will attempt to stabilize the pole at $s = 0$ and such a pole in the reference closed-loop

system is not stabilizable. However, this obstacle can be overcome by implementing a two-step design approach. First, the feedforward and feedback gains are designed using LQR methodology to achieve desired trajectory following and disturbance rejection criteria for the healthy aircraft. Then the plant model and feedback gains are removed from the reference controller and used as a target model for H_∞ synthesis. The target model, T, is given by Figure (6–2).

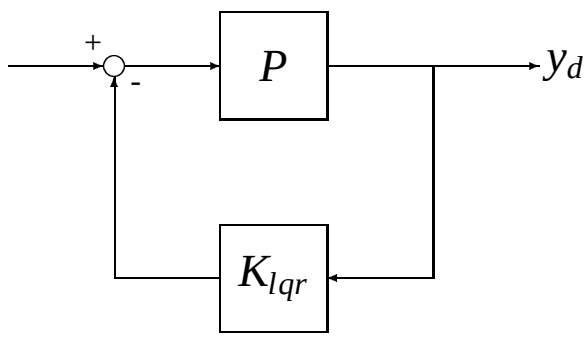

Figure 6–2: Target Model

The goal is to design an H_∞ controller that not only achieves the same tracking performance of the baseline controller but one that is capable of rejecting a disturbance caused by a jammed control surface. The problem can be set up as follows, let the postfailure state-space form be given as

$$\dot{x}(t) = Ax(t) + B_r u_r(t) + d \tag{6.10}$$

where u_r is the remaining control surfaces (i.e., failed surface is deleted), and d is a disturbance force. The measurements $y(t)$ are corrupted by noise such that

$$y_n(t) = y(t) + w_n n \tag{6.11}$$

Our objective is to design a control law so that the effect of the disturbance force d on the state measurements of the aircraft is reduced over an extremely small frequency range, $0 \leq \omega \leq 0.01$, such that the resulting disturbance is modeled as a constant force

upon the system. A low-pass filter given by

$$w_w = \frac{0.01w}{s+0.01} \tag{6.12}$$

is used to limit the disturbance force and achieve this goal. The result is a constant disturbance upon the system which is magnitude bounded by the position value of the jammed surface w. The synthesis model for H_∞ design is shown in Figure (6–3)

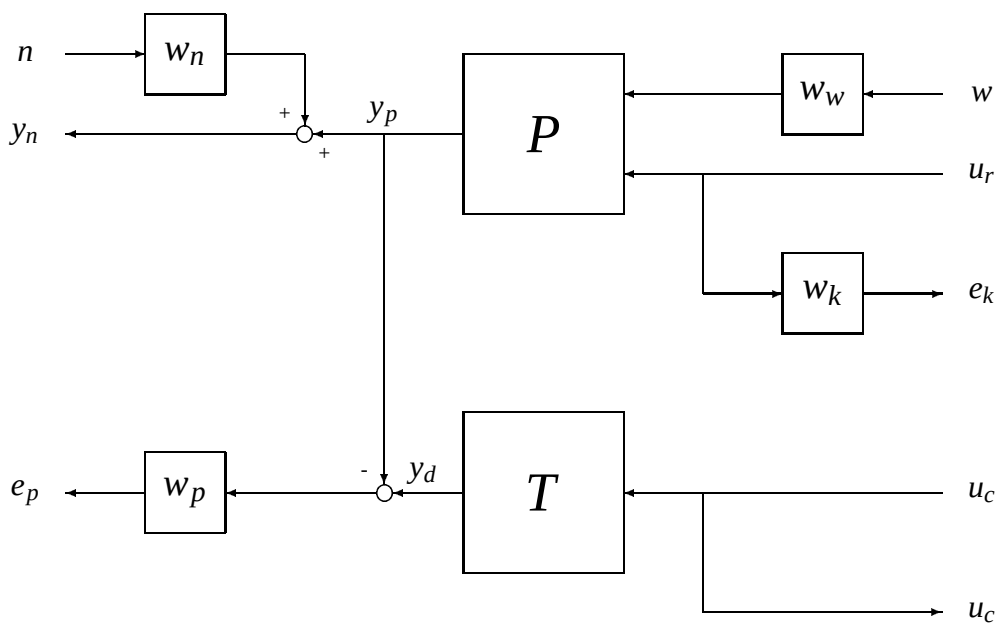

Figure 6–3: H_∞ synthesis model

where w_k is the control weight and w_p is the performance weight on the error signal of desired measurements to actual measurements

$$e = y_d - y \tag{6.13}$$

The synthesis model shown in Figure (6–3) along with weighting functions w_d, w_k, w_n, and w_p can be used to determine the sub-optimal H_∞ controller, K_{H_∞}, that minimizes the worst case tracking error norm over a magnitude bounded disturbance force. Thus, the primary goal of designing a controller that not only achieves the same tracking performance of the baseline controller but one that is capable of rejecting a disturbance

caused by a jammed control surface will be met by this H_∞ fault-tolerant design. The implementation of the H_∞ controller is shown in the analysis model in Figure (6–4).

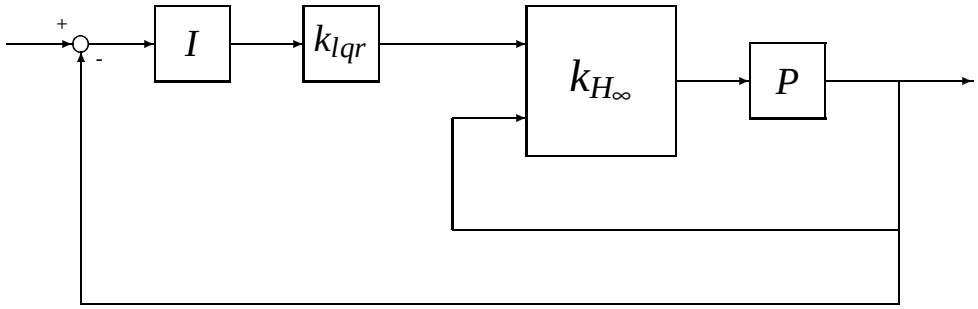

Figure 6–4: H_∞ analysis model

CHAPTER 7
APPLICATION TO AN F/A-18

In this final chapter the reconfiguration scheme proposed throughout this work is demonstrated on a high-fidelity nonlinear F/A-18 simulation. The simulation is based on the 6 degree-of-freedom equations of motion for a rigid body driven by aerodynamic, propulsive, and gravitational forces. The aerodynamic model is nonlinear and full independent control authority is available with realistic actuator models including rate and position limits. A nonlinear controller is included with this simulation that provides excellent performance and stability characteristics for a wide variety of high-performance maneuvers over the entire F/A-18 flight envelope. This controller is appropriate for the healthy aircraft and should not be altered; therefore, fault-tolerance will be achieved by switching to a predetermined controller for recovery and subsequent command following.

The failure scenario handled throughout this chapter involves a 2-inch longitudinal stick motion that commands a pitch doublet during which the left trailing-edge flap becomes stuck from a fixed-position actuator failure. It is assumed the failure is unknown but can be overcome by combining the fault detection and isolation procedure, Chapter (4), the stabilization procedure, Chapter (5), and either of the two formulated fault-tolerant control procedures, Chapter (6).

7.1 Healthy F/A-18

In this section the normal response of the F/A-18 to a 2-inch longitudinal stick doublet is reviewed for the unfailed case so that the reader can properly appreciate the evolution from the unfailed baseline controller to the fault-tolerant controllers. The maneuver is conducted in a 20 second simulation in which the pilot holds the stick at its neural point from 0-3 seconds, pulls and holds the stick at a positive 2 inches from

3-6 seconds, pushes and holds the stick at a negative 2 inches from 6-9 seconds, then returns and holds the stick to its neural point from 9-20 seconds.

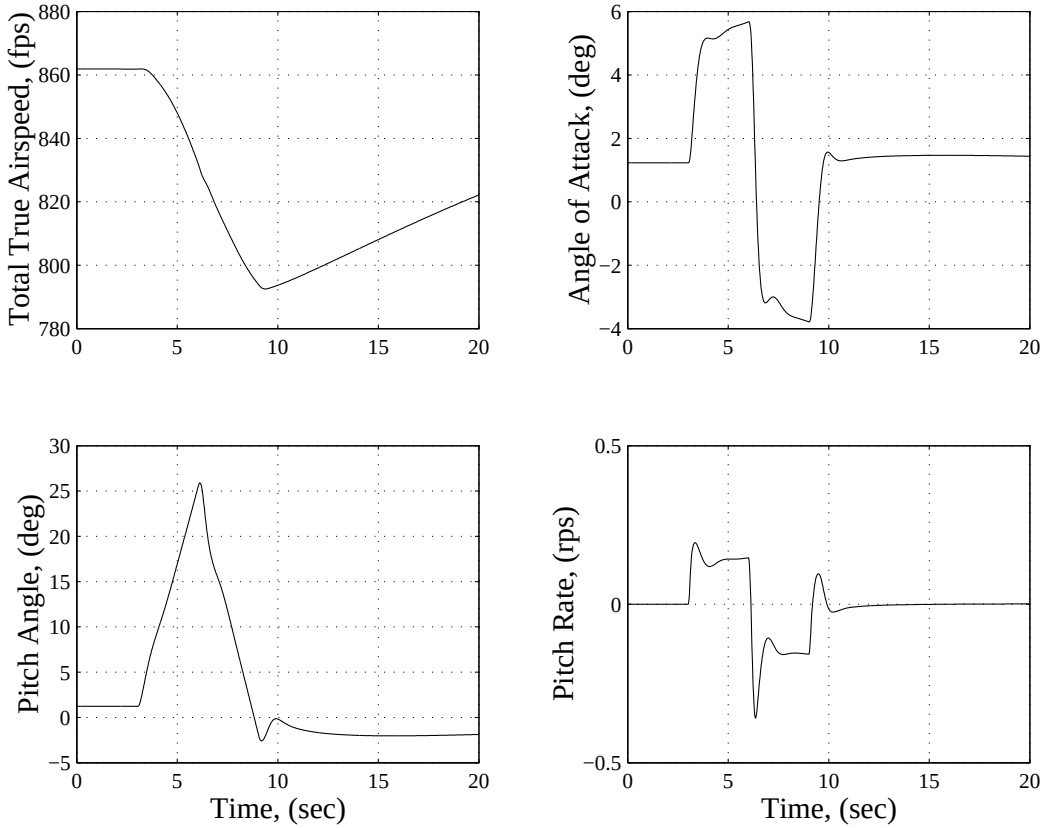

Figure 7–1: Longitudinal responses: healthy aircraft

Figure (7–1) shows the longitudinal responses to the 2-inch longitudinal stick doublet in the unfailed case. The maneuver causes the aircraft to pitch upwards at 0.15 radians/second from 3-6 seconds, then pitch down-wards at -0.15 radians/seconds from 6-9 seconds. The aircraft pitches to a maximum 25 degrees then returns back to wings level.

Figure (7–2) shows the lateral responses to the 2-inch longitudinal stick doublet in the unfailed case. The maneuver is essentially decoupled causing no response in the lateral states. This is characteristic of the F/A-18 which is known to have excellent maneuverability.

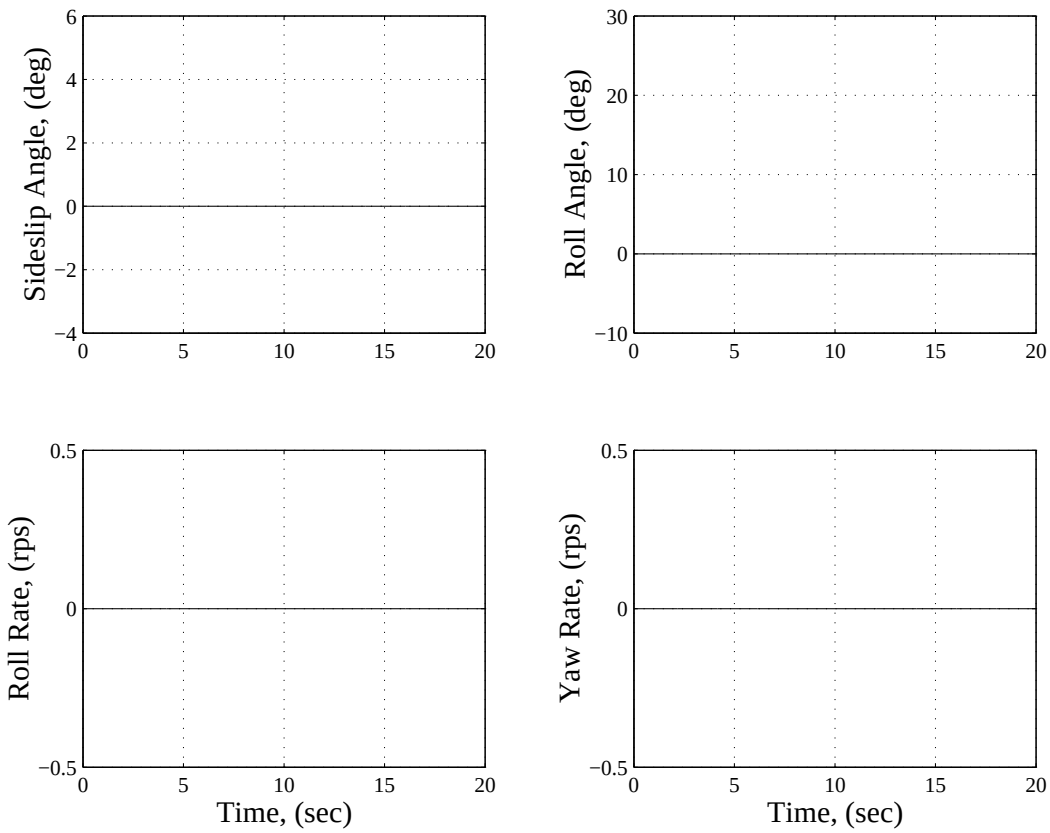

Figure 7–2: Lateral-directional responses: healthy aircraft

Figure (7–3) shows the control surface deflections commanded by the 2-inch longitudinal stick doublet in the unfailed case. The leading-edge flaps, trailing-edge flaps, and stabilators are used collectively to pitch the aircraft with the primary pitching moment being generated by the stabilators. It is important to note that neither the ailerons nor rudders are required to pitch the aircraft using the baseline controller in the unfailed case.

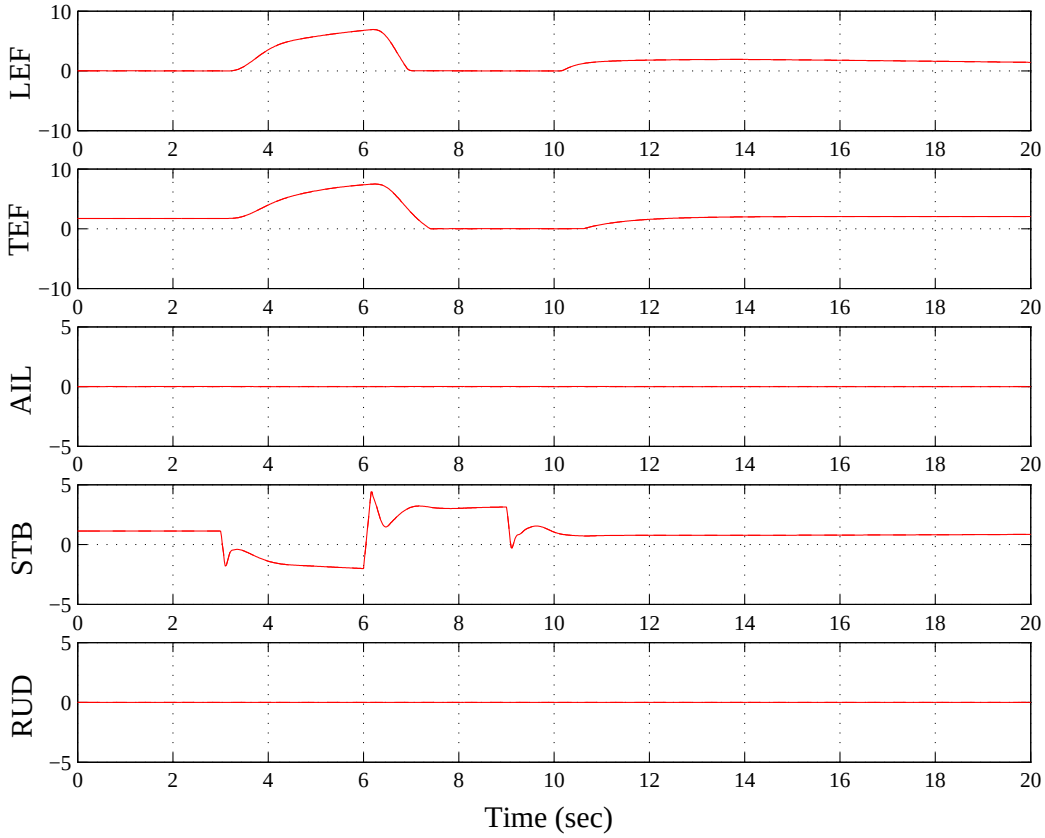

Figure 7–3: Control surface deflections: healthy aircraft

7.2 Failed F/A-18

In this section we show the effects of the left trailing-edge flap failure on the baseline controller. The maneuver is identical to the previous section with no attempt by the pilot to correct for the severe deviations from the desired trajectory. While this is not an ideal assumption, it is made in this case to simplify the fault detection and isolation process and to guarantee full control authority is passed to the fault-tolerant controller. The failure occurs at 4 seconds with the left trailing-edge flap becoming stuck at approximately 3.96 degrees. While this failure may not sound severe the effects upon performance and stability are devastating.

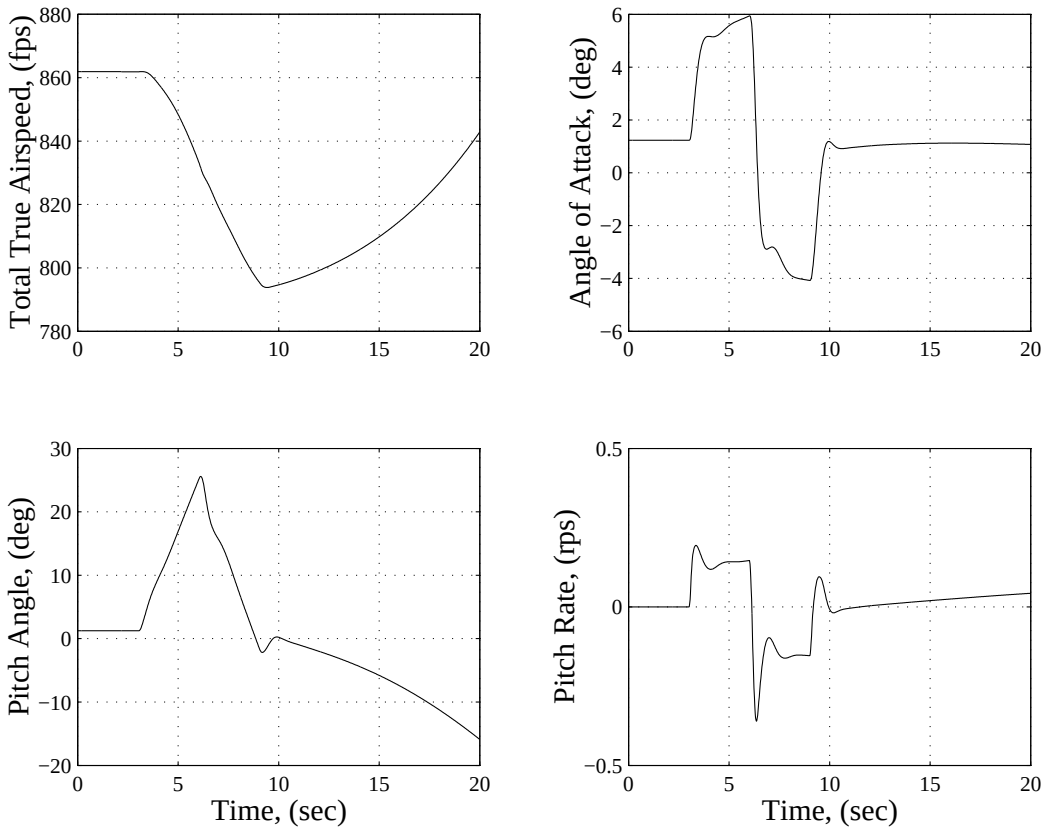

Figure 7–4: Longitudinal responses: failed aircraft

Figure (7–4) shows the longitudinal responses to the 2-inch longitudinal stick doublet with a left trailing-edge flap failure at 4 seconds. The aircraft retains reasonable responses for pitch angle and pitch rate during the maneuver. Once the aircraft is

commanded back to wings level it begins to pitch downward violently. At 20 seconds the aircraft is pitched downward at a negative 16 degrees with increasing pitch rate and total true airspeed.

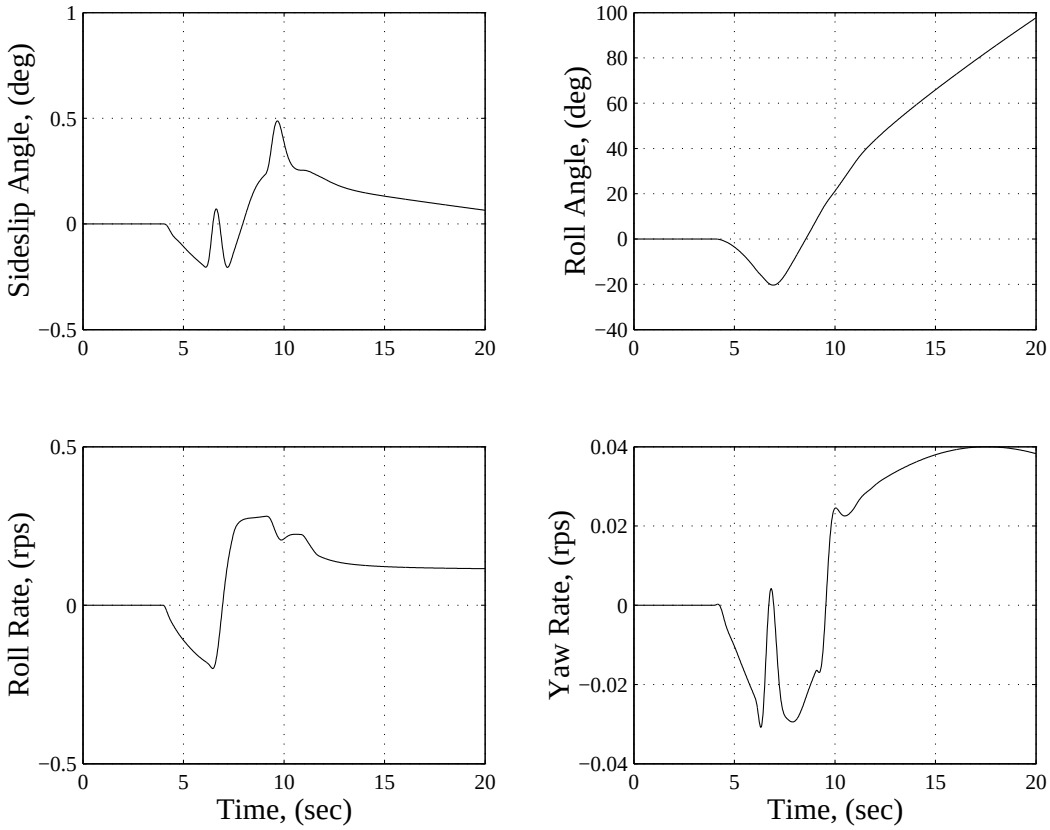

Figure 7–5: Lateral-directional responses: failed aircraft

Figure (7–5) shows the lateral responses to the 2-inch longitudinal stick doublet with a left trailing-edge flap failure at 4 seconds. As previously shown, for the unfailed aircraft's the lateral states are not excited by a pitch doublet. This decoupling is not the case for a pitch doublet in which the leading-edge flap has failed. At 20 seconds the aircraft has rolled completely on its side with a constant roll rate and a minimal side-slip and yaw rate. The aircraft continues to roll and pitch until it is nose down. Finally the aircraft impacts the ground in approximately 37 seconds (time of failure plus 34 seconds) traveling just over Mach 1.

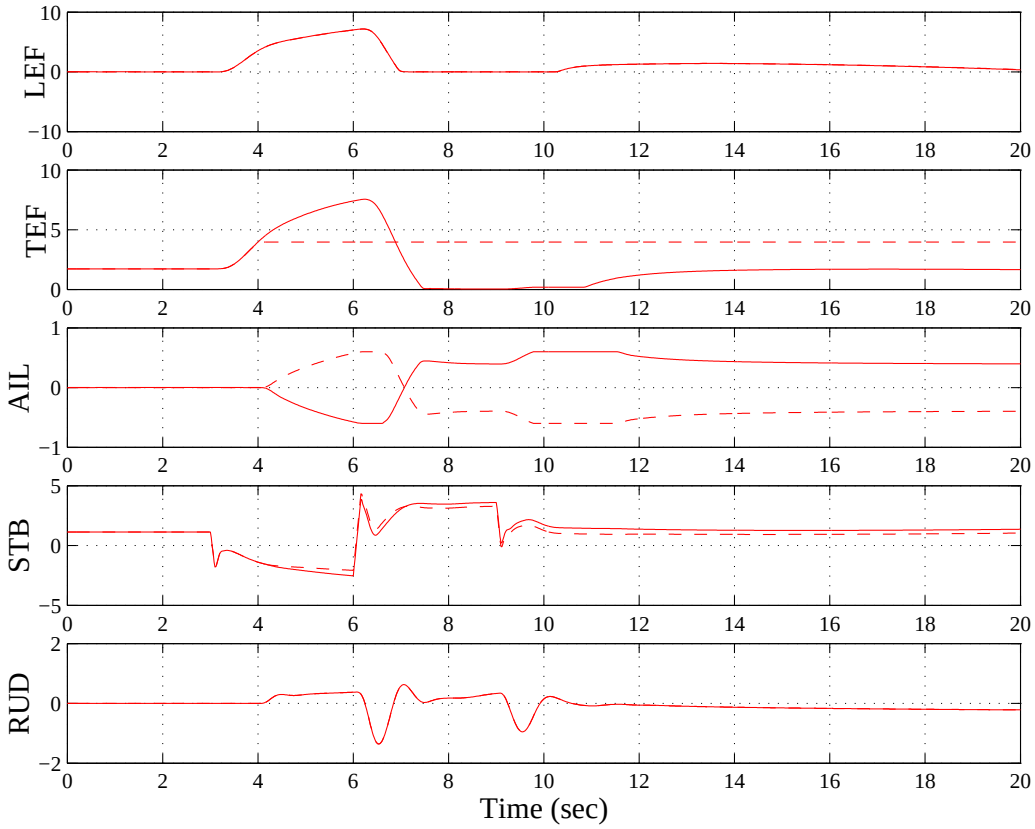

Figure 7–6: Control surface deflections: failed aircraft

Figure (7–6) shows the control surface deflections for a 2-inch longitudinal stick doublet with a left trailing-edge flap failure at 4 seconds. First, it is essential to notice the effects of the failure on the left trailing-edge flap. This is shown by the red dashed line in the second response. The position of the control surface remains constant after the failure instant. Also notable is the excitation of the ailerons and rudders by the feedback elements of the baseline controller attempting to counter the roll and yaw moments generated by the failure. This is also apparent in the differential stabilator deflection. Functioning with only nine operational control surfaces, the baseline controller proves ill-equipped to handle the failure.

7.3 Artificial Neural Network FDI

The artificial neural network (ANN) proposed in Chapter (4) for fault detection and isolation (FDI) of a fixed-position actuator failure is developed and evaluated in this section for the proposed scenario. The network was designed in Matlab and accepts as inputs five seconds of flight data sampled at 5 Hz. The measurements used for creating the input feature vector are Euler angles, Euler rates, angle-of-attack, side-slip, and position commands to the trailing-edge flaps, ailerons, stabilators, and rudders. It was determined that the position commands to the leading-edge flaps, which are primarily used for trimming the aircraft, were not producing a "rich" feature history; therefore, they were not used for training the network or performing the FDI operation. The removal of the leading-edge flaps position commands from the network design and operation is without incident, since there exists ample measurements with "rich" feature histories to generate desired results. As previously stated, the network is designed to accept as input a vector composed of position commands and aircraft measurements generated by the baseline controller; and output a vector identifying the failed control surface and failed postion in the event a failure occurs.

The network used for the scenario presented throughout this chapter was designed to monitor flight maneuvers including pitch and roll doublets and wind-up turns. The specifications of the network include: four layers with 312 neurons in the input layer; 156 neuron in the first hidden layer; 78 neurons in the second hidden layer; and 2 neurons in the output layer. The activation functions are constant throughout each layer and are *logsig*, *logsig*, *logsig*, and *pureline*, respectively. The learning algorithm selected was *trainscg*, which is a backpropagation technique where the network training function updates weight and bias values according to the scaled conjugate gradient method. The algorithm is capable of training any network as long as its weights, net inputs, and activation functions have derivatives, which are satisfied by the design. The network was designed and trained off-line with a fixed architecture. Once the desired

mean-squared error was achieved with the training algorithm the network weights and bias were not changed during FDI operation. For the proposed scenario, five seconds of flight data starting from the failure instant were input into the finalized network to produce FDI results. The actual failure occurs on the left trailing-edge flap (control surface #3) at a fixed-position of 3.96 degrees. The network results are given by

$$\text{ANN output} = \begin{cases} 3.01 & \text{failed surface} \\ 3.99 & \text{failed position} \end{cases} \tag{7.1}$$

These results demonstrate the capability of an artificial neural network to perform fault detection and isolation of fixed-position actuator failures. The first result, 3.01, represents the identifier for the failed control surface and correctly identifies the left-trailing edge flap (control surface #3) as the failed surface. The second result, 3.99, represents the identifier for the control surface position in degrees. This result achieves the desired accuracy and positively identifies a left trailing-edge flap failure with precision suitable to continue with trimming and fault-tolerant control. Acceptable results for this simulation would have included a failed surface identifier of ± 0.25 the actual integer value or a failed position identifier of ± 0.5 degrees the actual position value. While these ranges were reached through the process of trial and error, they have been determined to consistently facilitate desired results throughout the entire reconfiguration scheme.

7.4 Stabilization

The stabilization solution from Chapter (5) was implemented on the scenario presented throughout this chapter using the fault detection and isolation (FDI) results, Equation (7.1), from Section (7.3). The constraints placed upon the Matlab function *trim* included returning the aircraft with the failed control surface back to wings level flight at constant altitude, airspeed, and heading. This orientation can be expressed as

finding the nominal control surface position u_n such that

$$0 = B_r u_n + b_r w \tag{7.2}$$

while the nominal state vector x_n is given by

$$0 = x_n \tag{7.3}$$

Also, the allowable deflection of each control surface was constrained by the position limits defined by the nonlinear simulation, which can be found in Table (4–1). The results for a left trailing-edge flap failure at 3.99 degrees are given as

Table 7–1: Stabilization Results

Surface	Results
δ_{lef_l}	1.74°
δ_{lef_r}	1.74°
δ_{tef_l}	3.99°
δ_{tef_r}	-2.47°
δ_{ail_l}	-9.25°
δ_{ail_r}	3.47°
δ_{stb_l}	1.18°
δ_{stb_r}	-0.85°
δ_{rud_l}	0.07°
δ_{rud_r}	0.07°

These results can be easily verified by solving the equation representing the aircraft with a control surface failure, Equation (5.12), using the appropriate x_n, u_n, and w such that

$$0 = Ax_n + B_r u_n + b_w w \tag{7.4}$$

7.5 Fault-Tolerant Control Nonlinear Simulations

The results included in this section show the implementation of the developed reconfiguration techniques on a nonlinear F/A-18 simulation. The only alteration to the nonlinear simulation, inaddition to the new fault-tolerant controllers, involved permitting each control surface independent deflection rather than the traditional

collective or differential deflection of the baseline controller. The failure scenario continues from the previous sections. While performing a pitch doublet the left trailing-edge flap fails at 3.96 degrees at 4 second, resulting in an undesired roll, pitch, and yaw motion. The task of the reconfiguration scheme developed throughout this work is to positively identify that failure has occurred, determine the severity of the failure, and then switch control from the baseline controller to a fault-tolerant controller to regain stability and restore performance. The FDI procedure was performed off-line using five seconds of flight data from 4-9 seconds with acceptable results to proceed with control authority switching from the baseline controller to the fault-tolerant controller at 9 seconds. Then, a 2-inch longitudinal stick doublet is initiated at 23 seconds to demonstrate the command-tracking capabilities of each fault-tolerant controller on the nonlinear equations of motion. The goal here was to pitch the aircraft without exciting any lateral states of the aircraft, just as in the unfailed case.

The following figures show the results for each control methodology. Figures (7–7)-(7–9) show the control surface deflection and state responses using the LQR fault-tolerant controller and Figures (7–10)-(7–12) show the control surface deflection and state responses using the H_∞ fault-tolerant controller. For the state responses, the red dashed line is the desired performance of a healthy F/A-18 performing two consecutive pitch doublets while the black solid line is the results achieved with each fault-tolerant controller. In each case, the pitch moment is primarily generated by the deflection of the stabilators about a new trim point. Similar results are achieved by each fault-tolerant controller. The rise time during the commanded maneuver is slightly slower than the desired healthy F/A-18. The maneuver is performed with zero steady-state error and without producing any measurable roll angle or roll rate during the pitch maneuver even though the left trailing-edge flap has failed. The desired results are achieved, stability is restored, and command-tracking is performed.

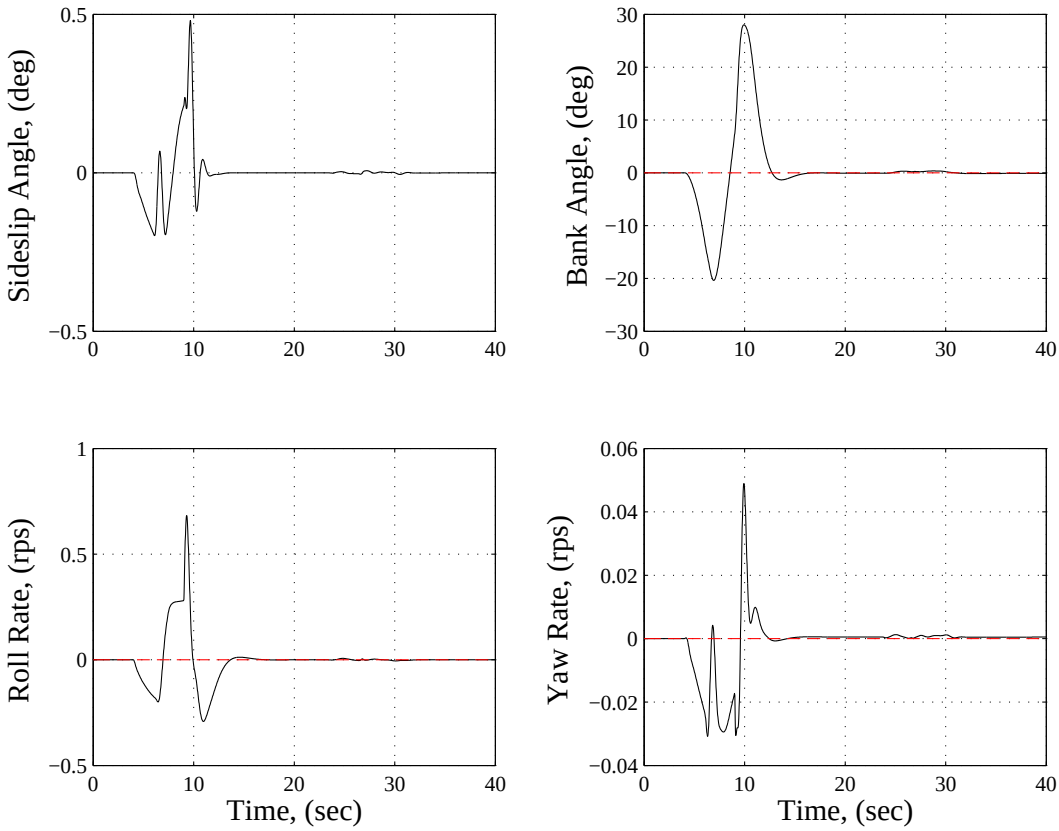

Figure 7–7: Lateral responses : LQR FTC

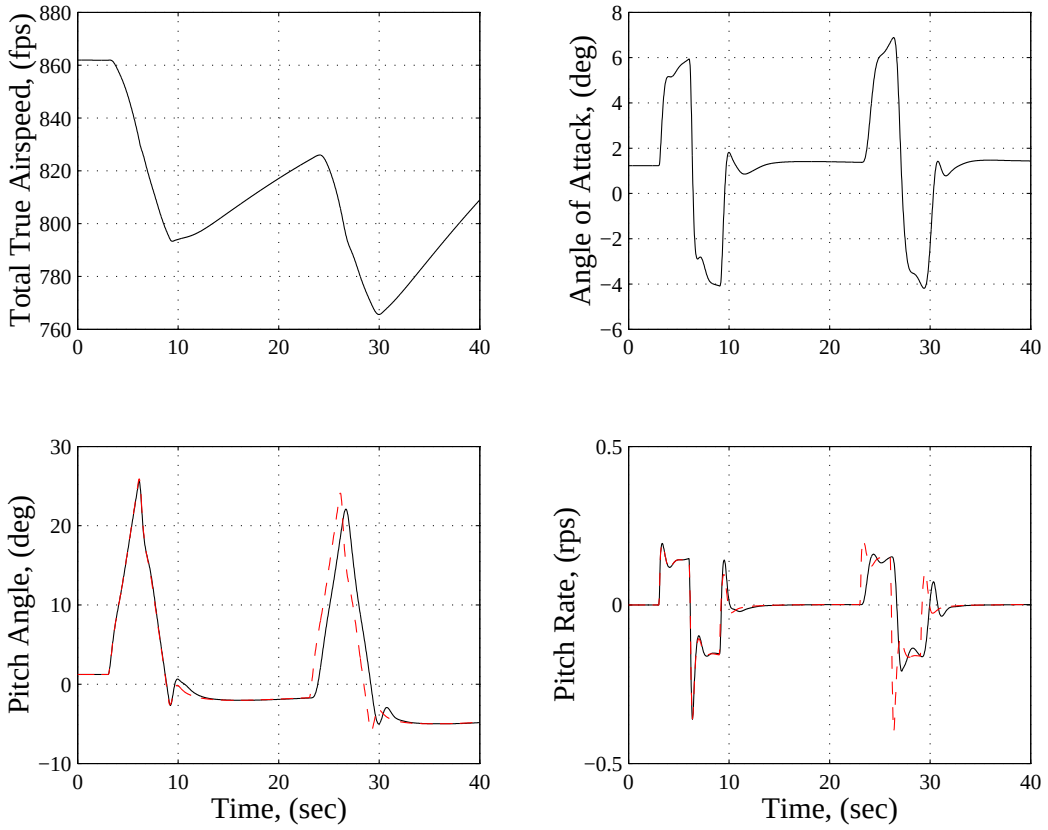

Figure 7–8: Longitudinal responses : LQR FTC

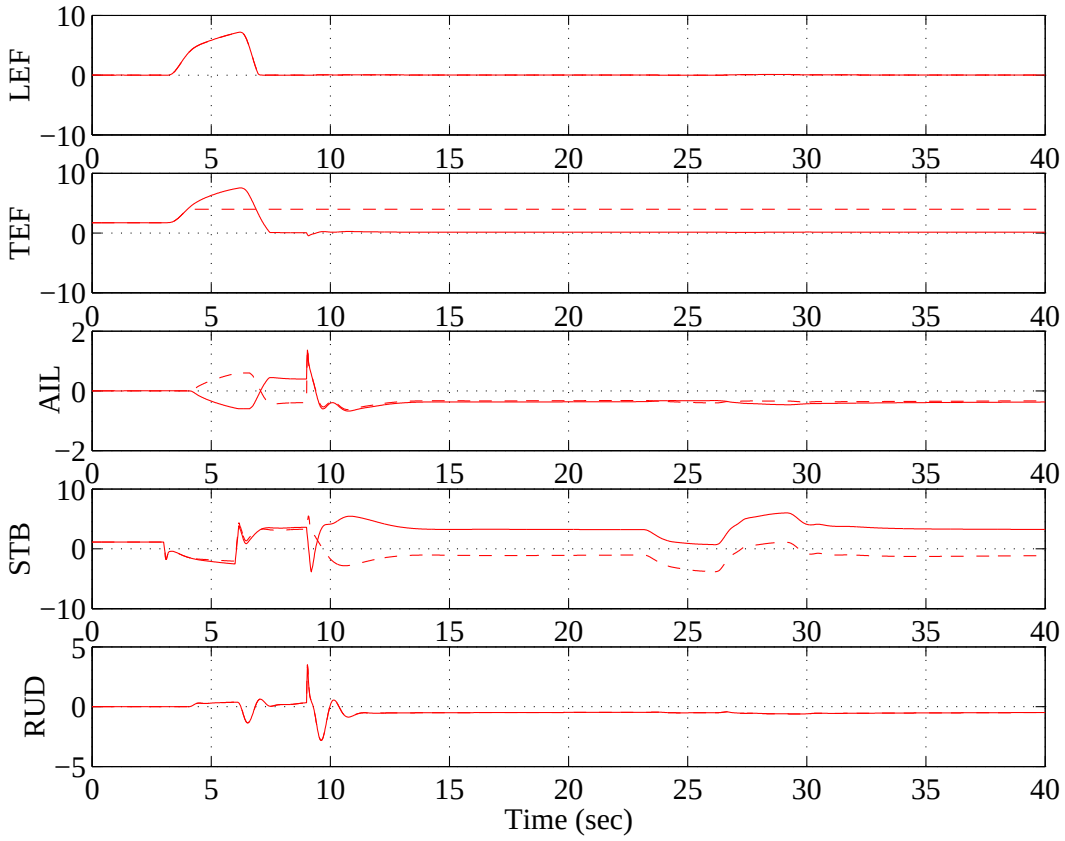

Figure 7–9: Control surface deflections : LQR FTC

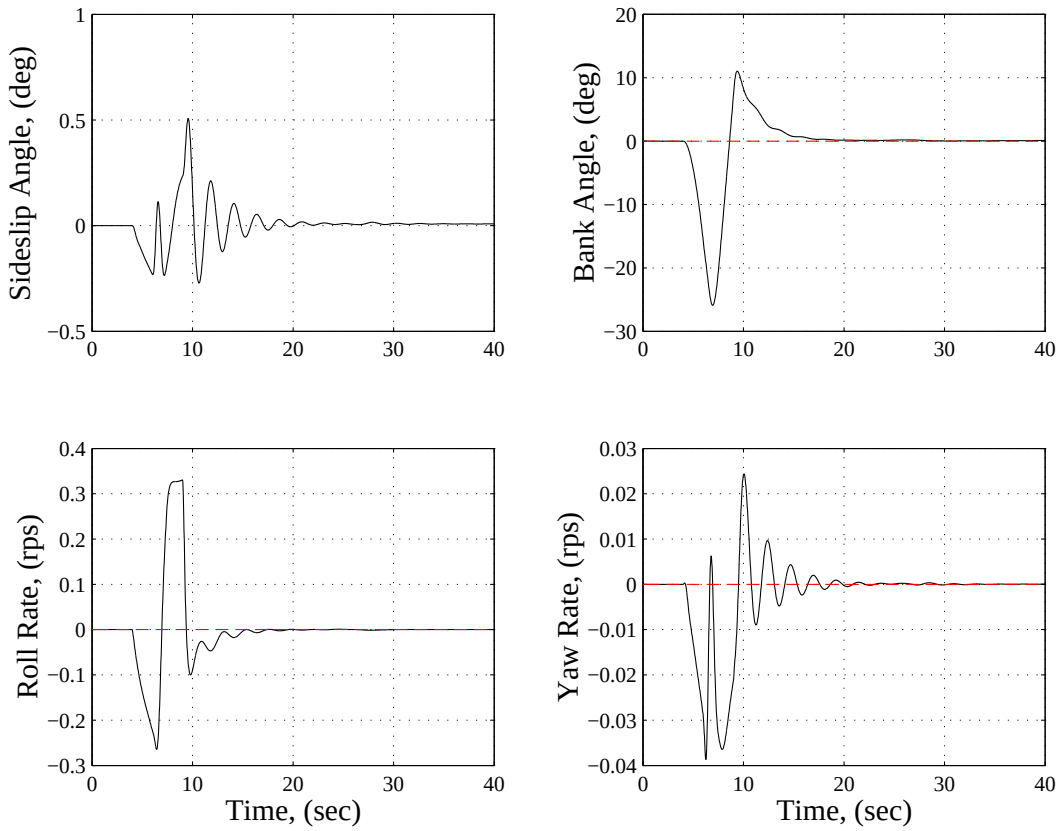

Figure 7–10: Lateral responses : H_∞ FTC

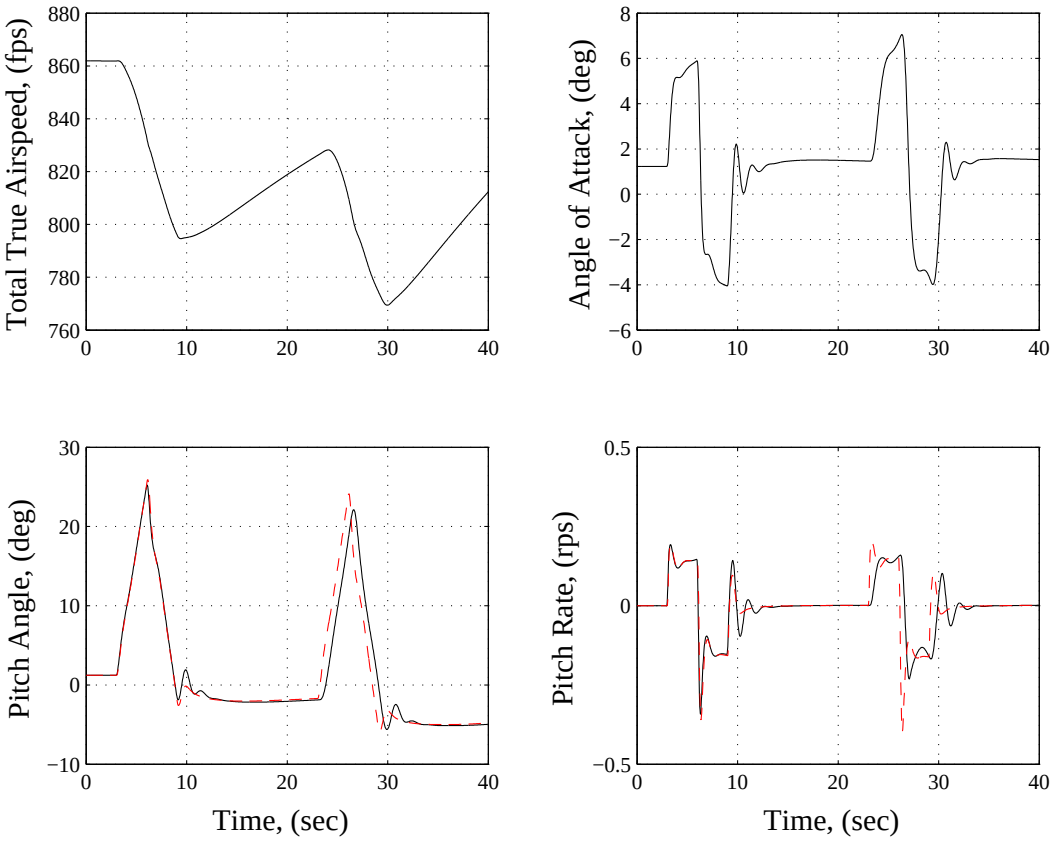

Figure 7–11: Longitudinal responses : H_∞ FTC

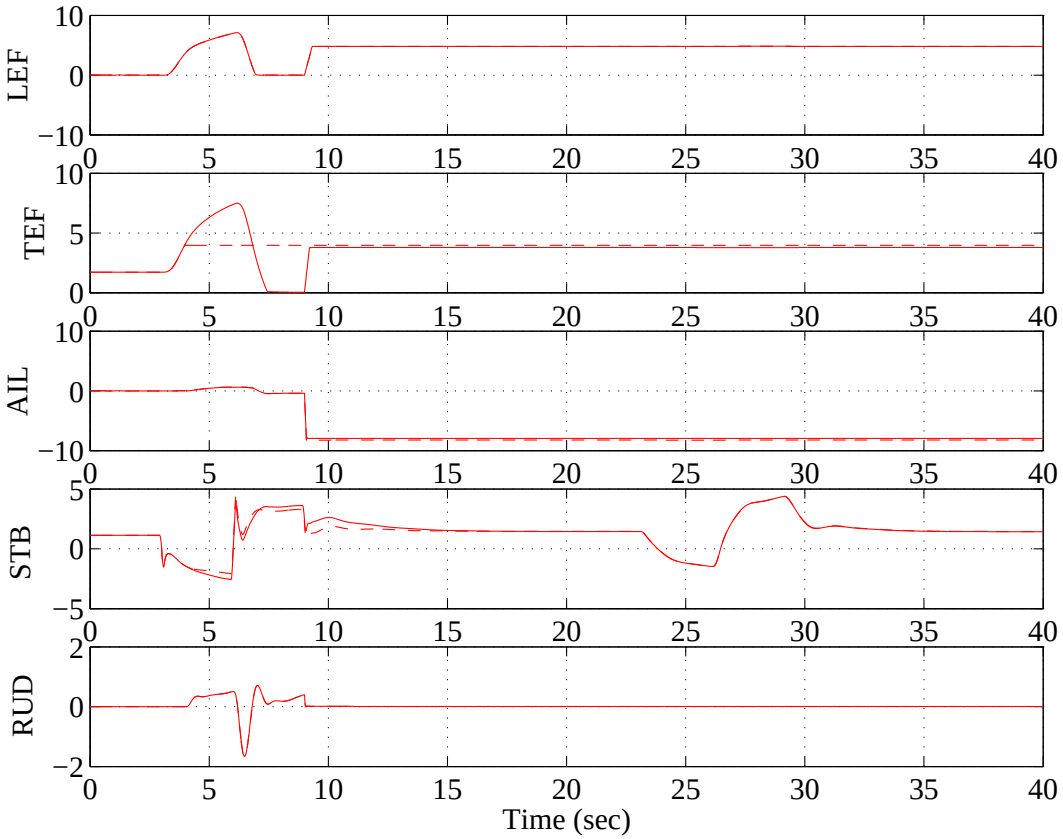

Figure 7–12: Control surface deflections : H_∞ FTC

CHAPTER 8
CONCLUSIONS

A reconfiguration scheme for flight control adaptation to fixed-position actuator failures is expected to accomplish three tasks. First, the scheme must have a fast and efficient method for identifying that a failure has occurred and the resulting effects upon stability and performance. Second, the scheme must adjust the trim values for command input so that level flight can be achieved. Third, the closed-loop system must ensure command-tracking, despite the detrimental effects of the failure and reduction in control effectiveness. The failure class analyzed throughout this work is a fixed-position or jammed actuator failure, which results in a flight control surface becoming inoperable. This work has introduced, developed, and demonstrated the necessary concepts to satisfactorily achieve all three goals for the targeted failure class.

The reconfiguration scheme developed through this work is a systematic procedure that attempts to maximize the tracking performance of the failed aircraft while satisfying the stability requirements. As a result, the proposed scheme relied on three interdependent processes: 1) fault detection and isolation, 2) stabilization, and 3) command-tracking. The use of artificial neural networks proved to be an excellent tool for identifying fixed-position actuator failures. These highly organized and versatile architectures were readily suited to perform the fault detection and isolation (FDI) task which maps state measurements into various failure classes. The results from the FDI procedure facilitated the development of a feedforward trim solution to recover system stability and two fault-tolerant control strategies to restore system performance. The two fault-tolerant methodologies explored, LQR and H_∞, assumed that the effects of the failed surfaces would introduce a constant disturbance into the dynamical equations governing the motion of the aircraft. The resulting theoretical development relied

on exploiting the robustness of each technique to directly address and overcome the effects of the failure by as nearly as possible reconstructing the forces and moments of the unfailed aircraft. The complete reconfiguration scheme was demonstrated on a nonlinear simulation of an F/A-18 to show the potential of the two methods in reconfigurable controls. The LQR and H_∞ methods achieved virtually the same results for the targeted failure class with both regaining stability and restoring performance in all instances.

The author of this work recognizes that several assumptions made throughout this work limit its application into flight systems. For example, the solutions presented throughout this work were developed using a single flight condition for a very specific failure class. No consideration was given to expanding the results over the entire flight envelope or into other failure scenarios. Furthermore, the standards used for judging the fault-tolerant controllers design were exceptionally high. Success was only defined by the the complete restoration of prefailure performance. In some situations in which an aircraft has suffered a significant system failure, it may not be necessary or even desirable to restore the performance to that of the healthy aircraft. Therefore, a method for determing how much performance is desired after a specific failure must be developed in conjunction with the pilots who fly the aircraft. Additionally, the incorporation of a reconfiguration scheme into an aircraft will most likely not be accomplished successfully post-production; rather, the tools for reconfiguration must be integrated into the initial design concepts of the aircraft. The initial design integration of reconfiguration technology may lead to design conflicts between normal operation and the rare occurrence of many of the failures currently under investigation in reconfigurable flight controls. Finally, a reconfiguration scheme will have to be flight tested to prove its usefullness in real-world situations. Such testing on full-size piloted aircraft is the essential step in demonstrating the promised benefits of the reconfiguration technology.

In summary, the results achieved in this work demonstrate the ability of artificial neural networks with linear control techniques to accommodate a very specific failure class while restoring stability and command-tracking to an aircraft which has experienced a significant control system failure. The methods developed here appear to be very effective in achieving the major objective to develop a reconfiguration scheme to accommodate fixed-position actuator failures.

APPENDIX A
LINEARIZED MODEL OF THE F/A-18

The following is a linearized model of an F/A-18 generated from a high fidelity six degree-of-freedom nonlinear simulator. Since the aircraft potentially has ten independent control surfaces, it is an ideal candidate for control restructuring and is used throughout this work for control synthesis. The flight conditions for the linearized model are Mach $= 0.8$, height $= 10,000$ ft, $\alpha_{\text{trim}} = 1.23°$, $\theta_{\text{trim}} = 1.23°$, $\phi_{\text{trim}} = \beta_{\text{trim}} = 0°$, and weight $= 30,777$ lbm. Let $u = \left(\delta_{lef_l}\ \delta_{lef_r}\ \delta_{tef_l}\ \delta_{tef_r}\ \delta_{ail_l}\ \delta_{ail_r}\ \delta_{stb_l}\ \delta_{stb_r}\ \delta_{rud_l}\ \delta_{rud_r}\right)^T$ be the input vector of control surfaces perturbations from the trim values, where δ_{lef_l}, δ_{lef_r} are the left and right leading-edge flaps, δ_{tef_l}, δ_{tef_r} the left and right trailing-edge flaps, δ_{ail_l}, δ_{ail_r} the left and right ailerons, δ_{stb_l}, δ_{stb_r} the left and right stabilators, and δ_{rud_l}, δ_{rud_r} the left and right rudders.

All of the control surface surface deflections are in degrees. The sign convention is positive leading-edge flap deflection is up, positive trailing-edge flap deflection is down, positive aileron deflection is down, positive stabilator deflection is down, and positive rudder deflection is left looking forward. The surface trim values are $1.72°$ for the trailing-edge flaps, $1.12°$ for the stabilators, and $0°$ for the trailing-edge flaps, ailerons, and rudders. Let the state vector for perturbations from the trim conditions be $x = (u\ w\ q\ \phi\ v\ p\ r\ \phi)^T$, where the components are, in order of appearance in x, forward velocity, vertical velocity, pitch rate, pitch angle, side velocity, roll rate, yaw rate, and roll angle. The units are in radians/second for angular rates, radians for angles, and feet/seconds for velocities. The linearized dynamics of the F/A-18 at the preceding

flight conditions are given by

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1}$$

where $x(t)$ is the state vector and $u(t)$ is the vector of available control surfaces. The dynamics of the A matrix are decoupled in the longitudinal and lateral directors, while the B matrix is not. The first four states represent the longitudinal dynamics and the second four represent the lateral dynamics such that

$$A = \begin{bmatrix} A_{lon} & 0 \\ 0 & A_{lat} \end{bmatrix} \tag{2}$$

and

$$B = \begin{bmatrix} B_1 & B_2 \end{bmatrix} \tag{3}$$

with

$$A_{lon} = \begin{bmatrix} -0.0209 & 0.0482 & -18.3387 & -32.1361 \\ -0.0377 & -1.8386 & 853.1909 & -0.6901 \\ 0.0002 & -0.0206 & -0.9431 & 0 \\ 0 & 0 & 1.000 & 0 \end{bmatrix} \tag{4}$$

$$A_{lat} = \begin{bmatrix} -0.3196 & 18.3106 & -860.7181 & 32.1361 \\ -0.0346 & -6.9243 & 0.7349 & 0 \\ 0.0098 & 0.0044 & -0.3233 & 0 \\ 0 & 1.0000 & 0.0215 & 0 \end{bmatrix} \tag{5}$$

$$B_1 = \begin{bmatrix} 0.0259 & 0.0259 & 0.0097 & 0.0097 & 0.0111 \\ 0.3174 & 0.3174 & -2.5367 & -2.5367 & -0.5699 \\ -0.0302 & -0.0302 & 0.0262 & 0.0262 & -0.0252 \\ 0 & 0 & 0 & 0 & 0 \\ -0.0000 & -0.0000 & 0 & 0 & -0.0381 \\ 0.0000 & 0.0000 & 0.5317 & -0.5317 & 0.3497 \\ 0.0000 & 0.0000 & -0.0066 & 0.0066 & -0.0028 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6}$$

$$B_2 = \begin{bmatrix} 0.0111 & -0.1304 & -0.1304 & 0 & 0 \\ -0.5699 & -2.0319 & -2.0319 & 0 & 0 \\ -0.0252 & -0.2560 & -0.2560 & -0.0005 & -0.0005 \\ 0 & 0 & 0 & 0 & 0 \\ 0.0381 & -0.2425 & 0.2425 & 0.5106 & 0.5106 \\ -0.3497 & 0.5134 & -0.5134 & 0.0765 & 0.0765 \\ 0.0028 & 0.0034 & -0.0034 & -0.0465 & -0.0465 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{7}$$

The open-loop eigenvalues of the aircraft are

$$\text{Longitudinal} \begin{cases} \text{Short Period:} & -1.3921 \pm 4.0273j \\ \text{Phugoid:} & -0.0076 \pm 0.0799j \end{cases}$$

$$\text{Lateral} \begin{cases} \text{Dutch Roll:} & -0.3535 \pm 2.9485j \\ \text{Spiral:} & -0.0014 \\ \text{Roll:} & -6.8587 \end{cases}$$

APPENDIX B
NOMINAL CONTROLLER DESIGN

The design process followed to arrive at the reference state-feedback controller used for the target model in the H_∞ design uses LQR methodology. Since the longitudinal and lateral dynamics are decoupled for the unfailed aircraft, we can design controllers for them separately. See Appendix A for the linearized aircraft model for the form

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{8}$$

To decouple the inputs, we need to mix them to obtain differential and collective inputs. We do this as follows. Let

$$B_{mix} \triangleq \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \tag{9}$$

so

$$u_{new} = B_{mix} \cdot u$$

where $u_{new} = \left(\delta_{lef_c}\ \delta_{tef_c}\ \delta_{stb_c}\ \delta_{lef_d}\ \delta_{tef_d}\ \delta_{ail_d}\ \delta_{stb_d}\ \delta_{rud_c}\right)^T$, where the components are, in order of appearance in u_{new}, collective leading-edge flaps, collective trailing-edge flaps, collective stabilators, differential leading-edge flaps, differential trailing-edge flaps, differential ailerons, differential stabilators, and collective rudders. The first three

mixed inputs represent the longitudinal control while the the last five mixed inputs represent the lateral control. We can now scale the inputs to ease our controller design such that 1 unit in each input is approximately equivalent in terms of importance. We pick the scaling for the new inputs as follows

$$u_p = S_1^{-1} u_{new} \tag{10}$$

where

$$S_1 \triangleq \mathrm{diag}\left[33, 45, 10.5, 33, 45, 45, 10.5, 30\right] \tag{11}$$

We then let

$$B_1 = B\left(B_{mix}\right)^{-1} S_1 \tag{12}$$

be our new B matrix, which is mixed and scaled. We can follow the same reasoning for the state variables such that

$$x_p = T_1 \cdot x \tag{13}$$

where

$$T_1 = \mathrm{diag}\left[0.01, 0.01, 1, 1, 0.01, 1, 1, 1\right] \tag{14}$$

then our new system matrices are

$$\begin{aligned} A_p &= T_1 \cdot A_a \cdot T_1^{-1} \\ B_p &= T_1 \cdot B_1 \end{aligned} \tag{15}$$

and the linear aircraft model becomes

$$\dot{x}_p = A_p x_p + B_p u_p \tag{16}$$

We can now split the aircraft into longitudinal and lateral models and design controllers for each individually.

Longitudinal Design

The longitudinal model is given by

$$A_{lon} = \begin{bmatrix} -0.0209 & 0.0482 & -0.1834 & -0.3214 \\ -0.0377 & -1.8386 & 8.5319 & -0.0069 \\ 0.0222 & -2.0562 & -0.9431 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \end{bmatrix}$$
$$B_{lon} = \begin{bmatrix} 0.0086 & 0.0044 & -0.0137 \\ 0.1047 & -1.1415 & -0.2134 \\ -0.9978 & 1.1789 & -2.6884 \\ 0 & 0 & 0 \end{bmatrix} \tag{17}$$

The states are the scaled versions of the longitudinal states, $x = (u\ w\ q\ \theta)^T$, and the scaled inputs are $u = \left(\delta_{lef_c}\ \delta_{tef_c}\ \delta_{stb_c}\right)^T$. We would like to control the pitch angle in the longitudinal axis. This is done by augmenting the system with an integrator on pitch angle. Let

$$C_{lon} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \tag{18}$$

so that

$$y = \theta = C_{lon} x \tag{19}$$

Then our new longitudinal system becomes

$$\begin{bmatrix} \dot{x} \\ \dot{x}_I \end{bmatrix} = \dot{z} = \begin{bmatrix} A_{lon} & 0 \\ C_{lon} & 0 \end{bmatrix} \begin{bmatrix} x \\ x_I \end{bmatrix} + \begin{bmatrix} B_{lon} \\ 0 \end{bmatrix} u \tag{20}$$

or

$$\dot{z} = \tilde{A}z + \tilde{B}z \tag{21}$$

We then proceed to design an LQR controller for this augmented model. We used

$$
\begin{aligned}
Q_{lon} &= \mathrm{diag}\,[0.0, 0.3, 1.2, 5.0, 36.0] \\
R_{lon} &= \mathrm{diag}\,[1.0, 1.0, 0.6]
\end{aligned}
\tag{22}
$$

The result is

$$
K_{lon} = \begin{bmatrix} -0.0009 & 0.0293 & -13.2190 & -54.4364 & -53.1336 \\ 0.0033 & -0.1085 & 24.7311 & 133.9899 & 116.7205 \\ -0.0010 & 0.0303 & -18.2382 & -69.0431 & -70.0172 \end{bmatrix} \tag{23}
$$

Figure (B-1) shows the state responses to a pitch doublet, the black line represents the nominal controller designed here while the red dashed line represents the nonlinear baseline controller.

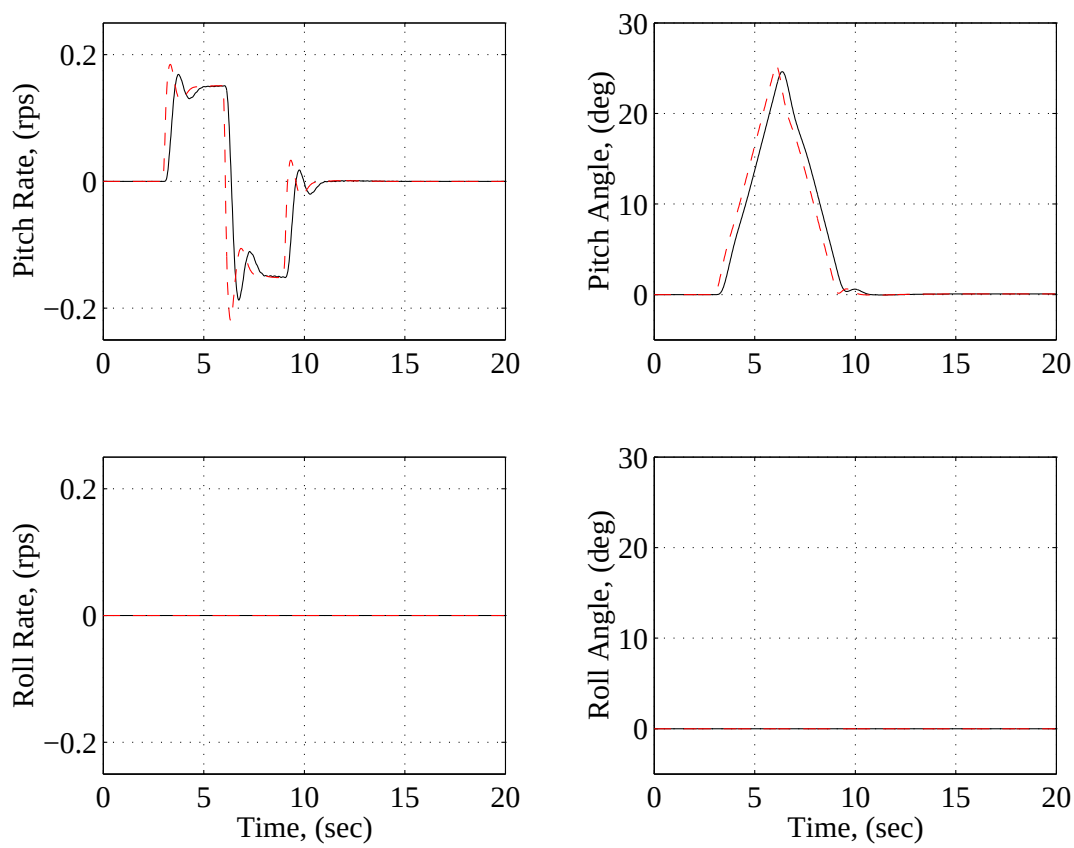

Figure B-1: Nominal Longitudinal Responses

Lateral Design

The lateral model is given by

$$A_{lat} = \begin{bmatrix} -0.3196 & 0.1831 & -8.6072 & 0.3214 \\ -3.4559 & -6.9243 & 0.7349 & 0.0 \\ 0.9842 & 0.0044 & -0.3233 & 0.0 \\ 0.0 & 1.0 & 0.0215 & 0.0 \end{bmatrix}$$
$$B_{lat} = \begin{bmatrix} 0.000 & 0.0 & -0.0172 & -0.0255 & 0.1532 \\ 0.000 & 23.9279 & 15.7358 & 5.3908 & 2.2956 \\ -0.0000 & -0.2959 & -0.1255 & 0.0360 & -1.3954 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{24}$$

The states are the scaled versions of the lateral states, $x = (v\ p\ r\ \phi)^T$, and the scaled inputs are $u = \left(\delta_{lef_d}\ \delta_{tef_d}\ \delta_{ail_d}\ \delta_{stb_d}\ \delta_{rud_c}\right)^T$. The goal we would like to achieve with lateral design is automatically coordinated flight. One way to achieve this is by controlling side velocity and roll angle so that a nonzero commanded roll angle with zero-commanded side velocity will produce a steady turn. This is done by augmenting the system with an integrator on side-velocity and roll angle. Let

$$C_{lat} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{25}$$

so that

$$y = \begin{bmatrix} v \\ \phi \end{bmatrix} = C_{lat} x \tag{26}$$

Then our new lateral system becomes

$$\begin{bmatrix} \dot{x} \\ \dot{x}_I \end{bmatrix} = \dot{z} = \begin{bmatrix} A_{lat} & 0 \\ C_{lat} & 0 \end{bmatrix} \begin{bmatrix} x \\ x_I \end{bmatrix} + \begin{bmatrix} B_{lat} \\ 0 \end{bmatrix} u \tag{27}$$

or

$$\dot{z} = \tilde{A}z + \tilde{B}z \tag{28}$$

We then proceed to design an LQR controller for this augmented model. We used

$$\begin{aligned} Q_{lat} &= \text{diag}\,[0.01, 0.001, 50.0, 0.01, 10.0, 1.0] \\ R_{lat} &= \text{diag}\,[1.0, 5.0, 1.0, 0.25, 1.0] \end{aligned} \tag{29}$$

The result is

$$K_{lat} = \begin{bmatrix} 0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0819 & 1.2723 & -14.2166 & 8.9759 & 0.1520 & 8.5733 \\ 0.2299 & 4.1653 & -29.2495 & 30.0738 & 0.4254 & 29.3576 \\ 0.0315 & 1.3124 & 9.3172 & 10.2124 & 0.0568 & 10.6313 \\ 0.4417 & 0.4901 & -230.8703 & -4.1777 & 0.8727 & -11.0881 \end{bmatrix} \tag{30}$$

Figure (B-2) shows the state responses to a roll doublet, the black line represents the nominal controller designed here while the red dashed line represents the nonlinear baseline controller.

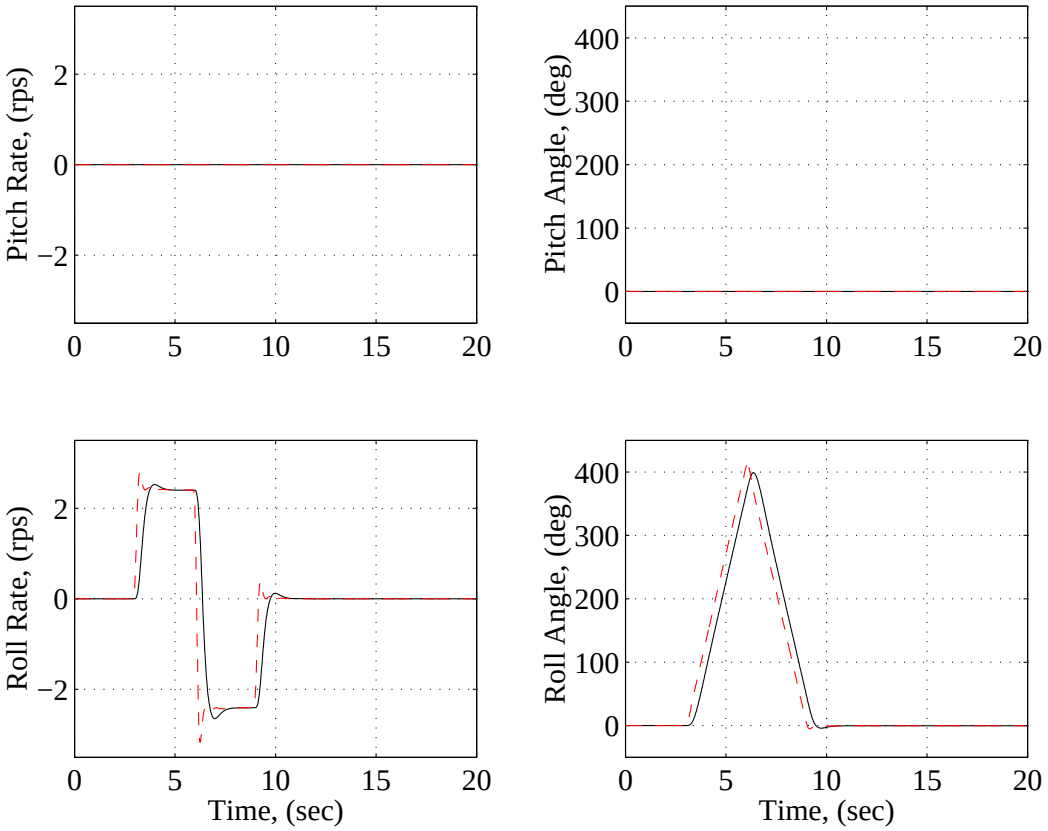

Figure B-2: Nominal Lateral Responses

REFERENCES

[1] Burken, J. J., Lu, P., Wu, Z., and Bahm, C., "Two Reconfigurable Flight-Control Design Methods: Robust Servomechanism and Control Allocation" *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 3, 2001, pp. 482-493.

[2] Stengel, R. F., "Intelligent Failure-Tolerant Control," *Proceedings of the 5th IEEE International Symposium on Intelligent Control*, Vol. 1, Philadelphia, PA, 1990, pp. 548-557.

[3] American Airlines A300-600, "An Airbus A300 of American Airlines on final approach to London Heathrow Airport," http://home.wanadoo.nl/airruud/a300aa.htm (visited July 10, 2003).

[4] American Airlines Flight 587 Belle Harbor, New York, "Investigation Information," March 2003, http://www.ntsb.gov/events/2001/AA587/default.htm (visited July 10, 2003).

[5] American Airlines Flight 587 Belle Harbor, New York, "American Airlines Flight 587 crashes in NYC," November 2002, http://www.cnn.com/2001/US/11/12/AA587.facts/ (visited July 10, 2003).

[6] American Airlines Flight 587 Belle Harbor, New York, "Flight 587 Crashes," October 2002, www.september11news.com/ Nov12Flight587AerialFire.jpg (visited July 10, 2003).

[7] American Airlines Flight 587 Belle Harbor, New York, "Feds rule out sabotage in N.Y. crash," October 2002, www.cnn.com/2002/US/South/10/29/flight.587/ (visited July 10, 2003).

[8] Eslinger, R. A., and Chandler, P. R., "Self-Repairing Control System Program Overview," *Proceedings of the IEEE National Aerospace and Electronics Conference*, Vol. 2, Dayton, OH, 1988, pp. 504-511.

[9] Patton, R. J., "Fault-Tolerant Control: The 1997 Situation," *Proceedings of IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes : SAFEPROCESS'97*, University of Hull, UK, 1998, pp. 1033-1055.

[10] Bajpai, G., and Chang, B. C., "Decoupling of Failed Actuators in Flight Control Systems," *Proceedings of American Control Conference*, Arlington, VA, 2001, pp. 1836-1840.

[11] Bajpai, G., Chang, B. C., and Lau, A., "Reconfiguration of Flight Control Systems for Actuator failures," *IEEE Aerospace and Electronics Systems Magazine*, Vol. 16, No. 9, 2001, pp. 29-33.

[12] Owen, D. H., *Feedback and Multivariable Systems*, Peter Peregrinus, Stevenage, England, 1978.

[13] Veilltette, R. J., Medanic, J. B., and Perkins, W. R., "Design of Reliable Control Systems," *IEEE Transactions on Automatic Control*, Vol. 37, No. 3, 1992, pp. 290-304.

[14] Birdwell, J. D., Castanon, D. A., and Athans, M., "On Reliable Control System Designs," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-16, No. 5, 1986, pp. 703-711.

[15] Blanke, M., Izadi-Zamanabadi, R., Bogh, S. A., and Lunau, C. P., "Fault Tolerant Control Systems : A Holistic View," *Department of Control Engineering*, R-1997-4175, Aalborg University, Denmark, March 1997.

[16] Willsky, A. S., "A Survey of Design Methods for Failure Detection in Dynamic Systems," *Automatica*, Vol. 12, No. 6, 1976, pp. 601-611.

[17] Mironovski, L. A., "Functional Diagnosis of Dynamic Systems : A Survey," *Automation and Remote Control*, Vol. 41, 1980, pp. 1122-1143.

[18] Isermann, R., "Process Fault Detection Based on Modeling and Estimation Methods : A Survey," *Automatica*, Vol. 20, No.4, 1984, pp. 387-304.

[19] Milne, R., "Strategies for Diagnosis," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-17, No. 3, 1987, pp. 333-339.

[20] Gertler, J., "Survey of Model-Based Failure Detection and Isolation in Complex Plants," *IEEE Control Systems Magazine*, December 1998, pp. 3-11.

[21] Frank, P. M., "Fault Diagnosis in Dynamic System Using Analytical and Knowledge Based Redundancy," *Automatica*, Vol. 26, No. 3, 1990, pp. 459-474.

[22] Patton, R. J. and Chen, J., "Robust Fault Detection Using Eigenstructure Assignment : A Tutorial Consideration and Some New Results," *Proceedings of the 30th IEEE Conference on Decision and Control*, Brighton, UK, 1991, pp. 2242-2247.

[23] Leitch, R. R., and Quek, H. C., "An Architecture for Integrated Process Supervision," *IEEE Proceedings Control Theory and Applications*, Vol. 139, No. 3, 1992, pp. 317-327.

[24] Patton, R. J., "Fault Detection and Diagnosis in Aerospace Systems Using Analytical Redundancy," *Computing and Control Engineering Journal*, Vol. 2, No. 3, 1991, pp. 127-136.

[25] De Persis, C., and Isidori, A., "A Geometric Approach to Nonlinear Fault Detection and Isolation," *IEEE Transactions on Automatic Control*, Vol. 46, No. 6, 2001, pp. 853-865.

[26] Frank, P. M., "On-line Fault Detection in Uncertain Nonlinear Systems Using Diagnostic Observers : A Survey," *International Journal of Systems Science*, Vol. 25, No. 12, 1994, pp. 2129-2154.

[27] Patton, R. J., Frank, P. M., and Clark, R. N., *Fault Diagnosis in Dynamic Systems, Theory and Applications*, Prentice Hall, New York, NY, 1989.

[28] Frank, P. M., "Application of Fuzzy Logic to Process Supervision and Fault Diagnosis," *Proceedings of the IFAC/IMACS Symposium on Fault Detection, Supervision and Safety for Technical Processes : SAFEPROCESS'94*, Helsinki, Finland, June 13-16, 1994, pp. 531-538.

[29] Isermann, R., "Integration of Fault Detection and Diagnosis Methods," *Proceedings of the IFAC/IMACS Symposium on Fault Detection, Supervision and Safety for Technical Processes : SAFEPROCESS'94*, Helsinki, Finland, June 13-16, 1994, pp. 597-612.

[30] Krishnasaami, V., and Rizzoni, G., "Nonlinear Parity Equation Residual Generation for Fault Detection and Isolation," *Proceedings of the IFAC/IMACS Symposium on Fault Detection, Supervision and Safety for Technical Processes : SAFEPROCESS'94*, Helsinki, Finland, June 13-16, 1994, pp. 305-310.

[31] Patton, R. J., Chen, J., and Nielsen, S. B., "Model-Based Methods for Fault Diagnosis : Some Guide-Lines," *Transactions of The Institute of Measurement and Control*, Vol. 17, No. 2, 1995, pp. 73-83.

[32] Isermann, R., and Balle, P., "Trends in the Application of Model Based Fault Detection and Diagnosis of Technical Processes," *Proceedings of the 13th IFAC World Congress*, Vol. N, San Francisco, USA, June 30 - July 5, 1996, pp. 1-12.

[33] Basseville, M., and Benveniste, A., "Detection of Abrupt Changes in Signals and Dynamical Systems," *Lecture Notes in Control and Information Sciences*, Vol. 77, Springer-Verlag, New York, NY, December 1985.

[34] Barron, R. L., Cellucci, R. L., Jordan, P. R., and Beam, N. E., "Application of Polynominal Neural Networks to FDIE and Reconfigurable Flight Control," *Proceedings of the IEEE National Aerospace and Electronics Conference*, Vol. 2, Dayton, OH, May 1990, pp. 507-519.

[35] Legg, J. M., "Computerized Approach for Matrix-Form FMEA," *IEEE Transactions on Reliability*, Vol. R-27, No. 1, October 1978, pp. 154-157.

[36] Herrin, S. A., "Maintainability Applications Using the Matrix FMEA Technique," *IEEE Transactions on Reliability*, Vol. 30, No. 3, 1981, pp. 212-217.

[37] Bell, A. R., "Managing Murphy's Law: Engineering a Minimum-Risk System," *IEEE Spectrum*, June 1989, pp. 24-26.

[38] Hunt, J. E., Pugh, D. R., and Price, C. J., "Failure Mode Effects Analysis: A Practical Application of Functional Modeling," *Applied Artificial Intelligence*, Vol. 9, No. 1, 1995, pp. 33-44.

[39] Savonov, M. G., *Stability and Robustness of Multivariable Feedback Systems*, MIT Press, Cambridge, MA, 1980.

[40] Morari M., and Zafirou, E., *Robust Process Control*, Prentice Hall, Englewood Cliffs, NJ, 1989.

[41] Maciejowski, J. M., *Multivariable Feedback Design*, Addison Wesley, Reading, MA, 1989.

[42] Zhou, K., Doyle, J., and Glover, K., *Robust and Optimal Control*, Prentice Hall, Upper Saddle River, NJ, 1996.

[43] Eterno, J. S., Weiss, J. L., Looze, D. P., and Willsky, A., "Design Issues for Fault Tolerant-Restructurable Aircraft Control," *Proceedings of the IEEE 24th Conference on Decision & Control*, Vol. 2, Fort Lauderdale, FL, 1985, pp. 900-905.

[44] Stengel, R. F., "Intelligent Failure-Tolerant Control," *IEEE Control System Magazine*, Vol. 11, No. 4, 1991, pp. 14-23.

[45] Lane, S. H., and Stengel, R. F., "Flight Control Design Using Non-Linear Inverse Dynamics," *Automatica*, Vol. 24, No. 4, 1988, pp. 471-483.

[46] Ochi, Y., and Kanai, K., "Design of Restructurable Flight Control Systems Using Feedback Linearization," *Jounrnal of Guidance, Control, and Dynamics*, Vol. 14, No. 5, 1991, pp. 903-911.

[47] Gao, Z., and Antsaklis, P. J., "Stability of the Pseudo-Inverse Method for Reconfigurable Control Systems," *International Journal of Control*, Vol. 53, No. 3, 1991, pp. 717-729.

[48] Calise, A. J., Lee, S., and Sharma, M., "Direct Adaptive Reconfigurable Control of a Tailless Fighter Aircraft," *Guidance, Navigation and Control Conference*, Boston, MA, August 1998, pp. 88-97.

[49] Idan, M., Johnson, M., and Calise, A. J., "A Hierarchical Approach to Adaptive Control for Improved Flight Saftey," *Journal on Guidance, Control and Dynamics*, July 2001, pp. 1012-1023.

[50] Johnson, E. N., and Calise, A. J., "Neural Network Adaptive Control of Systems with Input Saturation," *Proceedings of the American Control Conference*, Vol. 5, Arlington, VA, June 2001, pp. 3527-3532.

[51] Huang, C. Y., and Stengel, R. F, "Restructurable Control Using Proportional-Integral Implicit Model Following," *Jorunal of Guidance, Control, and Dynamics*, Vol. 13, No. 2, 1990, pp. 303-309.

[52] Morse, W. E., and Ossman, K. A., "Model Following Reconfigurable Flight Control System for the AFTI/F-16," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 6, 1990, pp. 969-976

[53] Jiang, J., "Design of Reconfigurable Control Systems Using Eigenstructure Assignments," *Internation Journal of Control*, Vol. 59, No. 2, 1994, pp. 395-410.

[54] Wu, N. E., "Robust Feedback Design with Optimized Diagnostic Performance," *IEEE Transactions on Automatic Control*, Vol. 42, No. 9, 1997, pp. 1264-1268.

[55] Nett, C. N., Jacobson, C. A., and Miller, A. T., "An Integrated Approach to Controls and Diagnostics : The 4-Parameter Controller," *Proceedings of the 1988 American Control Conference*, Atlanta, USA, 1988, pp. 824-835.

[56] Tyler, M. L., and Morari, M., "Optimal and Robust Design of Integrated Control and Diagnostic Modules," *Proceedings of the 1994 American Control Conference*, Baltimore, MD, June 1994, pp. 2060-2064.

[57] Murad, G. A., Postlethwaite, I., and Gu, D. W., "A Robust Design Approach to Integrated Controls and Diagnostics," *Proceedings of the 13th IFAC World Congress*, Vol. N, San Francisco, CA, USA, 1996, pp. 199-204.

[58] Akesson, M., "Integrated Control and Fault-Detection for a Mechanical Servo Process," *Proceedings of the IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes - SAFEPROCESS'97*, Hull, England, August 1997, pp. 1252-1257.

[59] Eich, J., and Sattler, B., "Fault Tolerant Control System Design Using Robust Control Techniques," *IFAC Safeprocess*, University of Hull, UK, 1997, pp. 1246-1251.

[60] Chiang, C. Y., and Youssef, H. M., "Neural Network and Fuzzy Logic Approach to Aircraft Reconfigurable Control Design," *Proceedings of the American Control Conference*, Vol. 5, Seattle, WA, June 1995, pp. 3505-3509.

[61] Patton, R. J., and Chen, J., "Robust Fault Detection and Isolation (FDI) Systems," *Control and Dynamic Systems*, Vol. 74, 1996, pp. 171-224.

[62] Mariton, M., "Detection Delays, False Alarm Rates and Reconfiguration of Control Systems," *International Journal of Control*, Vol. 49, 1989, pp. 981-992.

[63] Srichander, R., and Walker, B. K., "Stochastic Stability Analysis for Continuous Time Fault Tolerant Control Systems," *International Journal of Control*, Vol. 57, No. 2, 1993, pp. 433-452.

[64] Rauch, H., "Autonomous Control Reconfiguration," *IEEE Control Systems Magazine*, Vol. 15, No. 6, 1995, pp.37-48.

[65] Buckley, A. P., "Hubble Space Telescope Pointing Control System Design Improvement Study Results," *IEEE Control Systems Magazine*, Vol. 15, No. 2, 1995, pp. 34-42.

[66] Eryurek, E., and Upadhyaya, B. R., "Fault-Tolerant Control and Diagnostic for Large-Scale Systems," *IEEE Control Systems Magzaine*, Vol. 15, No. 5, 1995, pp. 34-42.

[67] Polycarpou, M. M., and Vemuri, A. T., "Learning Methodology for Failure Detection and Accomodation," *Control Systems Magazine, Special Issue on Intelligent Learning Control*, Vol. 15, 1995, pp. 16-24.

[68] Kwong, W. A., Passino, K. M., Laukonen, E .G., Yurkovich, S., "Expert Supervision of Fuzzy Learning Systems with Applications to Reconfigurable Control for Aircraft," *Proceedings of the 33rd IEEE Conference on Decision and Control*, Vol. 4, Lake Buena Vista, FL, December 1994, pp. 4116-4121.

[69] Kwong, W. A., Passino, K. M., Laukonen, E. G., Yurkovich, S., "Expert Supervision of Fuzzy Learning Systems for Fault-Tolerant Aircraft Control," *Proceedings of the IEEE*, Vol. 83, No. 3, March 1995, pp. 466-483.

[70] Fierro, R., Lewis, F. L., "A framework for hybrid control design," *Transactions on Systems, Man and Cybernetics*, Vol. 27, No. 6, November 1997, pp. 765-773.

[71] Horowitz, I., Arnold, P. B., and Houpis, C. H., "Flight Control System Reconfiguration Design Using Quantitative Feedback Theory," *Proceedings of the National Aerospace & Electronics Conference*, Dayton, OH, May 1985, pp. 578-585.

[72] Keating, M. S., Pachter, M., and Houpis, C. H., "QFT Applied to Fault-Tolerant Flight Control System Design," *Proceedings of the American Controls Conference*, Seattle, WA, June 1995, pp. 184-188.

[73] McFarlane, D. C., and Glover, K., "Robust Controller Design using Normalised Coprime Factor Plant Descriptions," *Lecture Notes in Control and Information Sciences*, Vol. 138, Springer Verlag, Berlin, Germany, 1989.

[74] Williams, S., and Hyde, R. A., "A Comparision of Characteristic Locus and H_∞ Design Methods for VSTOL Flight Control System Design," *Proceedings of American Controls Conderence*, San Diego, CA, May 1990, pp. 2508-2513.

[75] Passino, K. M., and Antsaklis, P. J., "On Inverse Stable Sampled Low Pass Systems," *International Journal of Control*, Vol. 47, No. 6, June 1988, pp. 1905-1913.

[76] Boskovic, J. D., and Mehra, R. K., "A Multiple Model-Based Reconfigurable Flight Control System Design," *Proceedings on the 37th IEEE Conference on Decision & Control*, Vol. 4, Tampa, FL, December 1998, pp. 4503-4508.

[77] Gopinathan, M., Boskovic, J. D., Mehra, R. K., and Rago, C., "A Multiple Model Predictive Scheme for Fault-Tolerant Flight Control Design," *Proceedings of the 37th IEEE Conference on Decision & Control*, Vol. 2, Tampa, FL, December 1998, pp. 1376-1381.

[78] Boskovic, J. D., and Mehra, R. K., "Stable Multiple Model Adaptive Flight Control for Accommodation of a Large Class of Control Effector Failures," *Proceedings of the American Control Conference*, Vol. 3, San Diego, CA, June 1999, pp. 1920-1924.

[79] Boskovic, J. D., Li, S. M., and Mehra, R. K., "Reconfigurable Flight Control Design Using Multiple Switching Controllers and On-Line Estimation of Damage Related Parameters," *Proceedings of the 2000 IEEE International Conference on Control Applications*, Anchorage, AK, September 2000, pp. 479-484.

[80] Boskovic, J.D., Li, S. M., and Mehra, R. K., "Study of an Adaptive Reconfigurable Control Scheme for Tailless Advanced Fighter Aircraft (TAFA) in the Presence of Wing Damage," *Proceedings of the Position, Location, and Navigation Symposium*, San Diego, CA, March 2000, pp. 341-348.

[81] Boskovic, J.D., Li, S. M., and Mehra, R. K., "Robust Supervisory Fault-Tolerant Flight Control System," *Proceedings of the American Control Conference*, Vol. 3, Arlington, VA, June 2001, pp. 1815-1820.

[82] Boskovic, J. D., and Mehra, R. K., "A decentralized scheme for accommodation of multiple simultaneous actuator failures," *Proceedings of the American Control Conference*, Vol. 6, May 2002, pp. 5098-5103.

[83] Boskovic, J. D., and Mehra, R. K., "Multiple model-based adaptive reconfigurable formation flight control design," *Proceedings of the 41st IEEE Conference on Decision and Control*, Vol. 2, December 2002, pp. 1263-1268.

[84] Boskovic, J. D., and Mehra, R. K., "Fault accommodation using model predictive methods," *Proceedings of the American Control Conference*, Vol. 6, May 2002, pp. 5104-5109.

[85] Kanev, S., and Verhaegen, M., "A Bank of Reconfigurable LQG Controllers for Linear Systems Subjected to Failures," *Proceedings of the 39th IEEE Conference on Decision and Control*, Vol. 4, Sydney, NSW, Australia, December 2000, pp. 3684-3689.

[86] Kanev, S., and Verhaegen, M., "Controller Reconfiguration for Non-Linear Systems," *Control Engineering Practice*, Vol. 8, October 2000, pp. 1223-1235.

[87] Kanev, S., Verhaegen, M., and Nijsse, G., "A Method for the Design of Fault-Tolerant Systems in Case of Sensor and Actuator Faults," *Submitted to the European Control Conference*, Houffalize, Belgium, September 2001.

[88] Demetriou, M. A., "Adaptive Reorganization of Switched Systems with Faulty Actuators," *Proceedings of the 40th IEEE Conference on Decision and Control*, Vol. 2, Orlando, FL, December 2001, pp. 1874-1884.

[89] Zhang, Y., and Jiang, J., "An Interacting Multiple-Model Based Fault Detection, Diagnosis and Fault-Tolerant Control Approach," *Proceedings of the 38th Conference on Decision & Control*, Vol. 4, Phoenix, AZ, December 1999, pp. 3593-3598.

[90] Maybeck, P. S., "Multiple Model Adaptive Algorithms for Detecting and Compensating Sensor and Actuator/Surface Failures in Aircraft Flight Control Systems," *International Journal of Robust and Nonlinear Control*, Vol. 9, 1999, pp. 1051-1070.

[91] Zhang Y., Rong Li, X., "Detection and Diagnosis of Sensor and Actuator Failures Using Interacting Multiple-Model Estimator," *Proceedings of the 36th IEEE Conference on Decision and Control*, Vol. 5, San Diego, CA, December 1997, pp. 4475-4480.

[92] Munir, A., Atherton, D. P., "Maneuvring Target Tracking Using Different Turn Rate Models in the Interacting Multiple Model Algorithm," *Proceedings of the 34th IEEE Conference on Decision and Control*, Vol. 3, New Orleans, LA, December 1995, pp. 2747-2751.

[93] Burken, J. J, and Burcham, F. W., "Flight-Test Results of Propulsion-Only Emergency Control System on MD-11 Airplane," *Journal of Guidance, Control and Dynamics*, Vol. 20, No. 5, October 1997, pp. 980-987.

[94] Burcham, F. W., Burken, J. J., Maine, T. A., and Bull, J., "Emergency Flight Control Using Only Engine Thrust and Lateral Center-Of-Gravity Offset: A First Look," *Technical report*, NASA, 1997.

[95] Burcham, F. W., Burken, J. J., Maine, T. A., and Fullerton, C. G., "Development and Flight Test of an Emergency Flight Control System Using Only Engine Thrust on an MD-11 Transport Airplane," *Technical report*, NASA, October 1997.

[96] Bordignon, K. A, and Durham, W. C., "Closed-Form Solutions to Constrained Control Allocation Problem," *Journal of Guidance, Control and Dynamics*, Vol. 18, No. 5, September 1995, pp. 717-725.

[97] Durham, W. C., and Bordignon, K. A., "Multiple Control Effector Rate Limiting," *Journal of Guidance, Control and Dynamics*, Vol. 19, No. 1, February 1996, pp. 30-37.

[98] Davidson, J. B., Lallman, F. J., and Bundick, W. T., "Integrated Reconfigurable Control Allocation," *Submitted to the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Montreal, Canada, August 6-9 2001.

[99] Zhenyu, Y., Huazhang, S., and Zongji, C., "The Frequency-Domain Heterogeneous Control Mixer Module for Control Reconfiguration," *Proceedings of the 1999 IEEE International Conference on Control Applications*, Vol. 2, Kohala Coast, HI, August 1999, pp. 1223-1228.

[100] Idan, M., Johnson, M., Calise, A. J., and Kaneshige, J., "Intelligent Aerodynamic/Propulsion Flight Control for Flight Safety: A Nonlinear Adaptive Approach," *American Control Conference*, Vol. 4, Arlington, VA, June 2001, pp. 2918-2923.

[101] Calise, A. J., Hovakimyan, N., and Idan, M., "Adaptive Output Feedback Control of Nonlinear Systems Using Neural Networks," *Automatica*, Vol. 37, No. 8, March 2001, pp. 1201-1211.

[102] Calise, A. J., Lee, S., and Sharma, M., "Development of a Reconfigurable Flight Control Law for the X-36 Tailless Fighter Aircraft," *Guidance, Navigation, and Control Conference*, Denver, CO, August 2000, pp. 896-902.

[103] Wise, K. A.,Brinker, J. S., Calise, A. J, Enns, D. F., Elgersma, M. R., and Voulgaris, P., "Direct Adaptive Reconfigurable Flight Control for a Tailless Advanced Fighter Aircraft," *International Journal of Robust and Nonlinear Control*, Vol. 9, No. 14, December 1999, pp. 999-1010.

[104] Brinker, J. S., and Wise, K. A., "Flight Testing of Reconfigurable Control Law on the X-36 Tailless Aircraft," *Journal of Guidance, Control and Dynamics*, Vol. 24, No. 5, September 2001, pp. 903-909.

[105] Shtessel, Y. B., and Buffington, J., "Multiple Time Scale Flight Control Using Reconfigurable Sliding Modes," *Journal on Guidance, Control and Dynamics*, Vol. 22, No. 6, 1999, pp. 873-883.

[106] Shtessel, Y., Buffington, J., Banda, S., "Tailless Aircraft Flight Control Using Multiple Time Scale Reconfigurable Sliding Modes," *IEEE Transactions on Control Systems Technology*, Vol. 10, No. 2, March 2002, pp. 288-296.

[107] Shtessel, Y. B., Tournes, C. C., "Integrated Flight Control Problem on Decentralized Sliding Modes," *Proceedings of the Twenty-Seventh Southeastern Symposium on System Theory*, Starkville, MS, March 1995, pp. 487-491.

[108] Shkolnikov, I. A., and Shtessel, Y. B., "Tracking MIMO Nonlinear Nonminimum Phase Systems Using Dynamic Sliding Manifolds," *Journal of Guidance, Control and Dynamics*, Vol. 24, No. 3, June 2001, pp. 826-836.

[109] Andry, A. N., Shapiro, E. Y., and Chung, J. C., "Eigenstructure Assignment for Linear Systems," *IEEE Transactions on Aerospace Electronic Systems*, Vol. 19, No. 5, September 1983, pp. 711-728.

[110] Davidson, J. B., and Andrisani, D., "Lateral-Directional Eigenvector Flying Qualities Guidelines for High Performance Aircraft," *Technical report*, NASA, December 1996.

[111] Konstantopoulos, I. K., and Antsaklis, P. J., "Eigenstructure Assignment in Reconfigurable Control Systems," *Technical report : Interdisciplinary Studies of Intelligent Systems*, January 1996.

[112] Belkharraz, A. I., and Sobel, K., "Fault-Tolerant Flight Control for a Class of Control Surface Failures," *Proceedings of the American Control Conference*, Vol. 6, Chicago, IL, June 2000, pp. 4209-4213.

[113] Zhang, Y., and Jiang, J., "Integrated Design of Reconfigurable Fault-Tolerant Control Systems," *Journal of Guidance*, Vol. 24, No. 1, July 2000, pp. 133-136.

[114] Bodson, M., and Groszkiewicz, J. E., "Multivariable Adaptive Algorithms for Reconfigurable Flight Control," *IEEE Transactions on Control Systems Technology*, Vol. 5, No. 2, 1997, pp. 217-229.

[115] Groszkiewicz, J. E., and Bodson, M., "Flight Control Reconfiguration Using Adaptive Methods," *Proceedings of the 34th Conference on Decision & Control*, Vol. 2, New Orleans, LA, December 1995, pp. 1159-1164.

[116] Maciejowski, J. M., "The Implicit Daisy-Chaining Property of Constrained Predictive Control," *Applied Math and Computer Science*, Vol. 8, No. 4, 1998, pp. 695-711.

[117] Huzmezan, M., and Maciejowski, J. M., "Reconfigurable Flight Control of a High Incidence Research Model Using Predictive Control," *UKACC International Confrence on Control*, Vol. 2, Swansea, UK, September 1998, pp. 1169-1174.

[118] Huzmezan, M., and Maciejowski, J. M., "Reconfiguration and Scheduling in Flight Using Quasi-LPV High-Fidelity Models and MBPC Control," *Proceedings of the American Control Conference*, Vol. 6, Philadelphia, PA, June 1998, pp. 3649-3653.

[119] Nelson, R. C., *Flight Stability and Automatic Control*, McGraw Hill, Boston, MA, 1998.

[120] Yechout, T. R., Morris, S. L., Bossert, D. E., and Hallgren, W. F., *Introduction to Aircraft Flight Mechanics*, American Institute of Aeronautics and Astronautics, Herndon, VA 2003.

[121] Hagan, M. T., Demuth, H. B., and Beale, M., *Neural Network Design*, PWS Publishing, Boston, MA, 1996.

[122] Haykin, S., *Neural Networks: A Comprehensive Foundation*, Prentice Hall, New York, NY, 1999.

[123] Cook, M. V., *Flight Dynamics Principles*, John Wiley & Sons Inc., New York, NY, 1997.

[124] Looze, D. P., Krolewski, S., Weiss, J. L., Barrett, N. M., and Eterno, J. S., "Automatic Control Design Procedures for Restructurable Aircraft Control" *NASA Contractor Report*, January 1985.

[125] Weiss, J. L.,Looze, D. P., Eterno, J. S., and Grunberg, D. B., "Initial Design and Evaluation of Automatic Restructurable Flight Control System Concepts" *NASA Contractor Report*, June 1986.

BIOGRAPHICAL SKETCH

Robert Eick was born in Long Beach, California. He attended Stanton College Preparatory High School in Jacksonville, Florida, where he received an International Baccalaureate diploma with a concentration in Renaissance art in May of 1998. Starting college at the University of Florida the following fall he received a Bachelor of Science in Aerospace Engineering with high honors in May of 2002. Following graduation he interned with John Burken at NASA Dryden Flight Research Center over the summer of 2002. He then started graduate school at the University of Florida under the supervision of Dr. Rick Lind joining a dynamics, systems, and controls laboratory. He is the recipient of the Brightfutures Scholarship (1998), the NASA Undergraduates Student Research Scholarship (2002), the Aerospace Engineering, Mechanics, and Engineering Science Best Paper Award (2002), the University of Florida Graduate Alumni Fellowship (2002), and the Mechanical and Aerospace Engineering Best Presentation Award (2003). He is a member of the AIAA.