Retrofit Systems for Reconfiguration in Civil Aviation

by

Jerry M. Wohletz

B.S. Aerospace Engineering
University of Kansas, 1994
M.Eng. Aeronautics and Astronautics
Massachusetts Institute of Technology, 1997

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2000

Author .....
Department of Aeronautics and Astronautics
January 07, 2000

Certified by .....
James D. Paduano
Principal Research Engineer of Aeronautics and Astronautics
Thesis Supervisor

Certified by .....
John J. Deyst
Professor of Aeronautics and Astronautics

Certified by .....
Eric Feron
Associate Professor of Aeronautics and Astronautics

Certified by .....
Anuradha M. Annaswamy
Principal Research Scientist of Mechanical Engineering

Accepted by .....
Nesbitt W. Hagood, IV
Associate Professor of Aeronautics and Astronautics
Chairman of Department Graduate Office

Retrofit Systems for Reconfiguration in Civil Aviation

by

Jerry M. Wohletz

Submitted to the Department of Aeronautics and Astronautics
on January 07, 2000, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

A new concept for retrofitting a reconfiguration module to an existing control law is reported in this thesis. The concept is motivated by the need for low cost, add-on modules that improve air safety in the existing fleet of civil air transport vehicles. A direct adaptive approach that accommodates control surface nonlinearities is adopted, which uses a slowly adapting model of the closed-loop aircraft as the reference model. The motivation, benefits, and components of the architecture are presented. In addition, the issues of control surface magnitude and rate saturation are addressed. A proof of stability is outlined for input-error adaptation when position and rate saturation are present. The reconfiguration architecture is demonstrated using an F/A-18 and a generic transport nonlinear simulator. General issues associated with commercial transport reconfiguration are highlighted. In both the longitudinal and directional axes, the control surfaces are not well balanced from a reconfiguration viewpoint. As a result, a novel reconfiguration control allocation scheme was devised that blends in all the control effectors in a given axis to perform the reconfiguration task. The simulation results revealed that the reconfiguration architecture does provide reconfiguration functionality for a wide variety of control surface failures. The reconfiguration potential is illustrated through comparisons of post-failure performance with and without reconfiguration via non-linear simulations. Additionally, comparisons between post-failure performance and nominal performance are made through non-linear simulations, closed-loop frequency responses, and aircraft handling qualities. For all of the failure scenarios illustrated, the simulation results showed that the aircraft without reconfiguration departs; with reconfiguration, nominal performance is achieve provided that adequate control authority exists post-failure.

Thesis Supervisor: James D. Paduano
Title: Principal Research Engineer of Aeronautics and Astronautics

Acknowledgments

I would like to thank all of those people who made this thesis possible. First and for most, I would like to thank my wife Tina, and my family for all of the support they provided me.

Next I would like to thank the faculty for their support and advice. I acknowledge Professor James D. Paduano for his continual support, both technically and financially, for both my masters and doctoral degree. Special thanks go to the other members of my Doctoral Committee: Professor John J. Deyst, Professor Eric Feron, Professor Anuradha M. Annaswamy, and Professor David W. Miller.

My sincerest appreciation goes to John J. Burken, Trindel Maine, and to all the members of the NASA Dryden Flight Research Center. I would also like to acknowledge Siva Banda and all the other members of the AFRL/VAAD Multivariable and Reconfigurable Control Center of Excellence. Would you believe that there are three golf courses on that WPAFB? Next, I would like to thank Dr. Kevin Wise, Darrin Groll, and Dave Evans from Boeing, formerly McDonnell Douglas, for sharing my vision and securing my educational leave-of-absence for both degrees.

All of my colleagues at MIT deserve my thanks and gratitude. Not only did I receive a technical degree, I refined by viewpoints on a wide assortment of geopolitical issues, and I acquired the fundamental skills for transferring momentum to pucks and other bodies. I have grown in more dimensions that can be measured. Special thanks go to labmates Piero Miotto, Marc Shewchun, Laurent G. Duchense, Vladislav Gavrilets, Emilio Frazzoli, Alex M. Budge, Arkadiy Turevskiy, Eric Durand, Kazutaka Takahashi, Nusawardhana, and all the other labmates that I interacted with through my four and half years at MIT.

THIS PAGE LEFT BLANK INTENTIONALLY

Contents

List of Figures

THIS PAGE LEFT BLANK INTENTIONALLY

List of Tables

THIS PAGE LEFT BLANK INTENTIONALLY

Symbols and Notation

Variables

Symbol	Description	Units
A	Linear State Dyanmics Matrix	
α	Angle of Attack	rad
B	Linear State Input Dynamics Matrix	
C	Linear Output State Dynamics Matrix	
D	Linear Output Input Dynamics Matrix	
d	Atmospheric Disturbance Vector	
Δu	Control Surface Saturation Level	^{o}
e_i	Input Error Vector	^{o}
e_o	Output Performance Error Vector	rad/s^2
f	Linear Disturbance Vector	
g	Higher-Order State Nonlinear Dynamics Vector	
Γ	Input Error Adaptation Gain	
h	Higher-Order Output Nonlinear Dynamics Vector	
j	Nonlinear State Dynamics Vector	
K	Reconfiguration Control Gain Matrix	
l	Nonlinear Output Dynamics Vector	
Φ	Input Error Adaptation Parameter Errors	

Symbol	Description	Units
r	Reference/Pilot Input	in
τ	Time Delay	sec
u	Control Input Vector	^{o}
V	Regressor Noise Vector	
v	Sensor Noise Vector	
W	Adaptation Parameter Error Lyapunov Equation	
w	Linear Atmosheric Disturbance Vector	
x	State Vector	
Z	Input Error Adaptive Regressor Matrix	
	Identification Model Structure Measurement	
z	Performace Vector	

Generic Subscripts

Subscript	Description
a	Actuator
c	Commanded
f	Linear Disturbance Vector
k	Nominal Controller
m	Desired Model
o	Upper Bound
p	Plant
p_m	Measured Plant
r	Reference/Pilot Input
rcm	Reconfiguration Control Module
x	State Vector

Generic Superscripts

Superscript	Description
$*$	Optimal Value
$'$	Modified
$\hat{}$	Parameter Estimate
$\tilde{}$	Parameter Error
T	Transpose

Acronyms

Acronym	Description
ARMAX	AutoRegressive Moving Average with eXternal input (Identification Model Structure)
LOES	Low-Order Equivalent System
MRAC	Model Reference Adaptive Control
MSLS	Modified Sequential Least-Squares
OE	Output-Error (Identification Model Structure)
PIM	Pseudo-Inverse Methods
RCM	Reconfiguration Control Module
RFCS	Reconfigurable Flight Control System

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 1

Introduction

"We will achieve a national goal of reducing the aircraft accident rate by 80% within 10 years" President William J. Clinton – February 12, 1997.

In response to this challenge, NASA established the Aviation Safety Program. The objective of this program is to reduce the fatal commercial accident rate by 80% in 10 years, and 90% in 20 years. If the accident rate was to remain at today's already low-level, by 2015, the mean time between major accidents would be from 7 to 10 days simply because of the large volumes of passengers and flights [1].

Traditionally, aviation safety improvements have accompanied significant technology milestones: introduction of jet engines, advances in materials and manufacturing, global navigation and communications, integrated air traffic control systems, etc. Considering the maturity of the commercial transport technology, it is unlikely that within the next two decades a single new technology will be able to improve the fatal accident rate to the extent desired. Thus, many incremental improvements must be made to achieve the safety objective.

As a result, NASA outlined several key technology areas to incrementally improve aviation safety, one of which is damage tolerant aircraft and control systems. According to Eslinger and Chandler [3], 20% of aircraft losses are due to faulty or damaged control systems. Zemlaykov [2]

notes that today, control surfaces are the primary cause of control system unreliability. Traditionally, hardware redundancy has been the primary approach of fault-tolerant flight control systems; however this approach has limitations in cost and weight. An alternative approach is to develop algorithms to support the fault-tolerance of a control system, and this approach has led to the field of Reconfigurable Flight Control Systems (RFCS).

Currently, pilot training is the sole method for commercial transport reconfiguration. Pilots can be trained to react to some anticipated failures; however, they cannot be expected to respond correctly and quickly to a wide range of conceivable failures. Furthermore, the pilot must know what type of failure has occurred in order to take corrective action. Failure classification can be a daunting task in an emergency situation. Eslinger and Chandler [3] also noted that about 70% of the accidents caused by control surfaces could have been prevented in principle. There are three well-publicized cases of commercial pilot reconfiguration during emergencies [4, 5, 6]. In all of these cases, the aircraft remained controllable despite the damage. The first case is a L-1011 — Delta Flight 1080, April 12, 1977 — where the left elevator became stuck at 19^o during takeoff; the pilot was able to reconfigure the vehicle. The next case is a DC10 — Chicago, May 25, 1979 — where the left engine flew off the airplane and struck the left outboard leading edge slat; the pilot was not able to reconfigure the vehicle and the aircraft was lost. Post-accident simulations demonstrated that proper reconfiguration by the pilot could have resulted in a successful landing. Finally, a DC10 — United Airlines Flight 232, July 1989 — had an uncontained tail engine failure that resulted in a complete loss of hydraulics; pilots were able to reconfigure the vehicle, but were unable to land safely. Post-accident flight tests demonstrated that a RFCS using only engines can provide adequate control to land a commercial transport [6].

From the above discussion, a RFCS that automatically tailors the control system for a certain class of failures has the potential to improve commercial aviation safety. With this motivation, the research questions are:

- How should the control system be altered in an emergency?
- What type of control surface failures can be accommodated?
- What is the resulting complexity of the RFCS?

These are the questions that this research will attempt to address.

1.1 Research Problem Statement

The overall performance objective of any RFCS is to implement a control strategies that will provide adequate control, whenever possible, despite large changes to the dynamics and despite the presence of sometimes large force and moment disturbances. For commercial transports, the control strategy to implement needs to be determined automatically and quickly so that a pilot may maintain adequate control during an emergency.

There are many obstacles to making commercial transports *reconfigurable.* One of the primary obstacles is that the commercial fleet in 10-20 years will contain a significant percentage of today's fleet; thus, complex, all-encompassing changes to the flight control laws of these vehicles is a costly proposition. Furthermore, any new technology requiring significant changes to the flight control design methodology is unlikely to be accepted by industry.

Considering the primary obstacles and the performance objectives, the focus of this research is to develop reconfiguration strategies that can be modularly added to existing flight control systems while keeping cost and complexity low.

The type of emergency situations that this research will address are failures to the control surfaces. The failure set includes unanticipated, single or multiple, simultaneous or sequential failures that affect the control power of the aircraft, that may affect the baseline aerodynamics, and that may produce large disturbances. The failure set excludes those unsolvable areas where the plane cannot be saved, i.e, a static equilibrium must exist and there must be enough control power available to stabilize the aircraft.

Figure 1-1 illustrates the concept of a modular reconfiguration strategy. An algorithm that is completely separate from the existing control law takes information about the state of the vehicle, command inputs, and control law outputs; it then generates an augmentation signal that compensates for the deficiencies in the basic control law. A switch locks out these signals when

there is no failure. To ensure smooth transition from the primary control laws to the augmented system, the reconfiguration module must naturally exhibit zero or very small outputs when there is no failure.

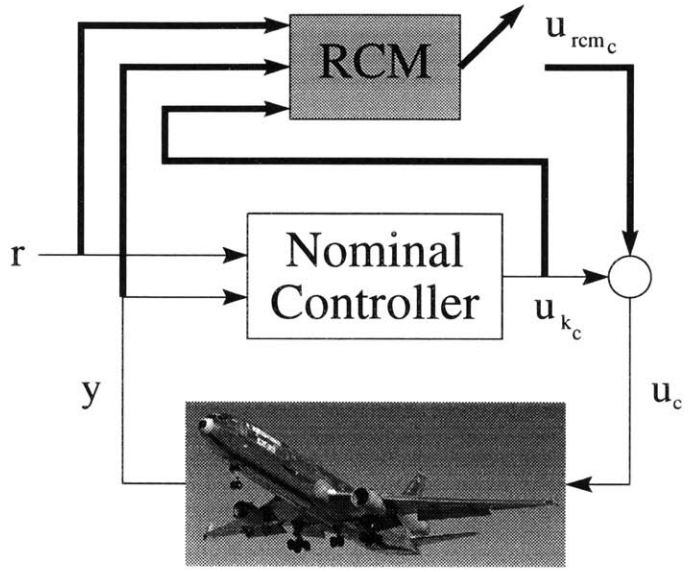

Figure 1-1: Retrofit Reconfiguration Architecture for Civil Aviation Aircraft

1.2 Literature Survey

The performance of control systems under off-nominal conditions has always been a central theme in control design. In recent years, attention has focused on control system robustness, i.e., satisfactory performance for a nominal controller in the presence of model uncertainty. Clearly, control surface failures can be viewed as model uncertainty. The are two approaches to the control system robustness problem: passive and active. Passive robustness has a rich developed theory for linear systems [7, 8]. Weiss et al [9] demonstrated that a flight control system designed using passive robustness techniques could posses fault-tolerance for a certain class of control system failures. However, passive robustness approaches require complete redesign of the flight control system, which is not compatible with our problem statement.

The active approach to the robustness problem has evolved into the field of adaptive control.

There are three broad categories of adaptive control schemes in the literature: diagnostic, indirect, and direct. Some have been investigated for aircraft reconfiguration, and a few have actually been tested in flight. The objective of this section is to review some of these general approaches and to identify possible schemes that fit the research problem statement. The focus of this discussion will be on general techniques that are prominent in the literature; the survey presented is by no means exhaustive.

A note to the reader, the following three sections contain a brief introductory discussion of the general advantages and disadvantages of the category as it pertains to the problem statement in Section 1.1. The reader may scan these brief discussions and proceed directly to Section 1.3.

1.2.1 Diagnostic Adaptive Control

Adaptive control systems in this class posses the ability to accommodate system failures automatically based upon *a priori* assumed conditions. On-line failure detection and isolation methods are used to diagnose the failure, and based on this result, appropriate control action is implemented based on stored information. Advantages of this approach are summarized below:

- Control solutions can reflect current industry design methodologies.
- Implementation is straight forward.
- Nominal performance is not affected.
- Rapid adaptability is possible.

The primary disadvantages of this approach are:

- Design and analysis process at each flight point must be duplicated for each failure mode.
- Stability is not guaranteed for failure modes that were not modeled *a priori*.
- Failure mode data must be stored in the flight control computer.
- Suffers from the 'Curse of Dimensionality'.

Adaptive control techniques that belong to this class include gain scheduling and controller scheduling.

Gain Scheduling

Gain scheduling was the first form of controller reconfiguration for the purpose of achieving good performance throughout the flight envelope. This approach has been extended for the purpose of reconfiguration by adding the failure status to the parameterization of the controller gains. Many of the early reconfiguration attempts via gain scheduling were based on Pseudo-Inverse Methods (PIM) [2, 10, 3]. The objective of this approach was to minimize the differences between the nominal and failed linear time-invariant closed loop dynamics in the sense of a Frobenius norm [10]. Minimizing the Frobenius norm of this objective function minimizes the upper bound of the differences between the nominal and failed closed-loop eigenvalues. Thus, if the minimized objective function is zero, the nominal and failed closed-loop poles are identical. The primary limitation of this approach is that minimization of the Frobenius norm does not guarantee stability. Furthermore, the minimization may result in compensator gains that will saturate the control surfaces. Gao [10] suggested an extension to this general concept by using linear robust synthesis combined with the PIM to guarantee stability. It is worth noting that the PIM gain scheduling approach formed the basis of an Air Force program – The Self-Repairing Flight Control System Program – in the 1980s, which lead to a flight demonstration [3].

Another gain scheduling approach is based on the Constrained Least-Squares Method (CLSM) [11]. In this approach, reconfiguration is performed via control allocation. The procedure relies on minimizing the differences between the nominal and failed forces/moments produced by generalized forces/moment commands. The minimization is performed in a least squares sense, and constraints are added to limit high gains and coupling. Like the PIM, this approach does not guarantee stability. This technique is currently being implemented on the X33 [11].

Wise and Sedwick [12] address the issue of ensuring stability of failure dependent gain schedules for failures that lie between the discrete set of *a priori* modeled failures. Stability analysis is presented using linear matrix inequalities for failure-dependent models that are characterized by their scheduling parameters. This technique determines a conservative bound of the closed loop system stability and relative stability given an *a priori* designed controller for a discrete number of failures.

More recently, Chen et al [13] propose a failure-dependent gain scheduling controller based on a linear matrix inequality design. Assuming that the failures are additive in nature, a robust fault-tolerant controller was obtained for a short period mode which exhibited satisfactory performance and guaranteed stability robustness.

Controller Scheduling

The other broad category of diagnostic adaptive control is controller scheduling. In this approach, the entire controller is switched or blended based upon the diagnostic results. A well known controller scheduling approach is called multiple model, which has its roots in LQ control theory. As a hypothesized alternative to gain scheduling, Athans et al [14] investigated the potential of a multiple model, full envelope controller for the NASA F-8C. The architecture consisted of a bank of Kalman filters and LQ controllers all producing control outputs simultaneously. The implemented control action is a weighted sum of all the controller outputs; the weighting is determined online and reflects the conditional probability that the aircraft is at a certain point in the flight envelope. Maybeck [15, 16] has since extended this method for RFCS.

An additional example of a controller scheduling approach is the Propulsion-Controlled Aircraft (PCA) [6]. For this flight test demonstration, the PCA controller was used as an emergency control system in the event of complete loss of hydraulic power. Flight tests demonstrated that a commercial transport could land using engines as the only control device.

1.2.2 Indirect Adaptive Control

Indirect adaptive control involves on-line parameter estimation of the plant, and this information is then used to implement a suitable control law. The general area of indirect adaptive control is primarily based on classical techniques used for designing feedback systems that have been extended for the purpose of adaptation. Advantages of this approach are summarized below:

- Control architecture is generally modular: feedback control, system identification, and control allocation.

- Independent technology advances in any of these areas benefit the overall system.
- Primarily based on classical design techniques.
- Flexible in incorporating *a priori* knowledge of the plant.

The primary disadvantages of the indirect adaptive control are:

- Performance is generally linked to good on-line parameter estimation; thus, unbiased convergence is required.
- Persistent excitation is required for parameter convergence, and this can be difficult for closed-loop control with multiple effectors.
- Inherent trade-off exists between speed and accuracy of parameter estimation.
- Proof of stability is generally not available.

Conforming to the modular nature of indirect adaptive control, the literature survey will be presented separately in the following categories: system identification, controller redesign and control allocation.

Parameter Estimation

The general theory of parameter estimation is well documented in Ljung [17] and Mendel [18]. The objective of parameter estimation is to provide unbiased estimates of the aircraft's stability and control derivatives with and without impairment. A parameter estimation methodology that has been successfully applied to several flight simulations was developed at the Air Force Research Lab. The foundation of this technique is an equation error, minimum-variance parameter estimation approach [18]. This is a static system identification approach, which is well suited for aircraft reconfiguration if the aircraft states and *state rates* are measurable. The theory was expanded to online estimation of stability and control derivatives in [19]. The fundamental requirement to obtain real-time estimates of aircraft parameters was addressed by exploiting the flight mechanics. The estimation problem is formulated as a constrained linear regression that is computed either by batch or recursive processing. The constraints to the linear regression can include relationships between stability and control derivatives, *a priori* estimates of derivatives,

or restrictions on the rate of change of the estimate. The methodology was simulated for the pitch plane dynamics of the F-16 VISTA [19].

The next advancement to the methodology was the use of regularization techniques for real-time identification of aircraft parameters [20]. The fundamental limitations of closed-loop identification for a moderate failure with tightly regulated control were addressed. Tight control attenuates the additional excitation caused by the failure; thus, the signal-to-noise ratio might not be adequate for identification. Also, the control is often a linear combinations of the states, which results in biased estimates. To address these problems, singular value decomposition is used to re-parameterize the model, and *a priori* data is used to generate estimates of the original parameter set. This stage of the methodology was demonstrated on F-16 VISTA simulations, and included both longitudinal and lateral-directional dynamics [20]. Additionally, a version of this methodology was flight tested on the F-16 VISTA [21, 22].

The method has now been extended to aircraft with distributed control effectors [23]. This advancement addresses the relationship between the system identification and control allocation modules; on-line control allocation requires knowledge of the aerodynamic control derivatives. For aircraft with redundant control effectors, traditional control allocation algorithms gang a subset of 'favorite' control surfaces while not exciting the remaining control surfaces. This control allocation approach often reduces the rank of the regressor matrix and results in poor estimates. This problem is circumvented by superimposing additional excitation into the control surfaces while attempting to minimize the associated motion of the aircraft.

An additional advancement to the system identification algorithm is the separation of estimation tasks into two separate stages. The first stage estimates the stability derivatives and generalized loop gain for the purpose of rapid stabilization after the onset of failure. The second stage estimates the control power derivatives associated with the individual control effectors; this information is used in the control allocation to optimize a performance metric. This version of the estimation methodology was demonstrated on simulations of the Innovative Control Effectors (ICE) tailless aircraft [23]. This aircraft is a delta wing configuration with eleven individual control effectors. The two-stage parameter estimation methodology has also being extended to

aircraft that have highly nonlinear aerodynamics such as the Tailless Advanced Fighter Aircraft (TAFA) [24, 25, 26]. This aircraft is an agile, stealth configuration that is similar to the X-36.

As in all parameter estimation techniques, there is an inherent trade-off between the parameter convergence rate and parameter sensitivity to noise. For the above methodology, a hybrid batch/sequential approach is used that utilizes a short data window for fast convergence and *a priori* knowledge as constraints to dampen parameter variations. Bodson [27, 28] has suggested a modified recursive weighted least-squares algorithm which has similar performance to the previous algorithm with the addition of a second-order convergence response. The two algorithms were compared off-line using flight test data from the F-16 VISTA [29].

Controller Redesign

Given that the plant has been successfully identified, there are a number of on-line control design methods that can be used. In general, the controller redesign methods are extensions of non-adaptive control techniques.

Probably the most extensively applied technique for on-line flight control redesign is the receding horizon, model predictive control approach. For the general theory of receding horizon control, see Michalska and Mayne [30], and for a general overview of model predictive control see Mehra [31]. Pachter, Chandler, and Mears [32, 33] extended this approach to the case of aircraft reconfiguration. The basic structure involves the on-line solution of an LQ cost function — penalizing tracking error and control — over a prediction horizon. Because of the LQ formulation, a closed form optimal control solution exists and is a function of the current state, previous pilot input, and the predicted pilot input over the prediction horizon. Using this functional dependence, an outer loop linear program was employed to prevent control surface rate saturation. Again, a closed form solution was obtained. In [32, 33], simulation results were presented for an F-16 with a severe failure, and good tracking performance was demonstrated. Based on this success, Barron Associates incorporated this controller redesign approach into their Self-Designing Controller Design Program and demonstrated the approach through flight test of the F-16 VISTA with a simulated missing stabilator [21, 22].

The literature contains several examples of LQ design techniques that were extended for the purpose of reconfiguration. Looze, Weiss, et al [4, 9] demonstrate a LQ feedback controller that attempts to recover the nominal return difference while maintaining the actuator bandwidth constraints. Given the unfailed control and state weighting matrices, failed linear dynamics, and some measure of uncertainty about the failed model, an algebraic Ricatti equation was solved on-line to determine the appropriate control action. This reconfiguration strategy was implemented in a B737 simulator. Failure analysis was performed at a single flight point for various control surface failures. The overall performance demonstrated that this technique was able to provide fault tolerant control for surface failures.

Another application of LQ control theory for reconfigurable flight control is presented in Ahmed-Zaid, et al [34]. This study explores the reconfiguration of an F-16 using the nominal control system with a linear quadratic regulator (LQR) as an adaptive outer loop. The only axis for which the adaptive LQG was added was the roll axis. The architecture contains on-line parameter estimation, a state observer, and a Riccati solver. Nonlinear simulation demonstrated that this approach was capable of accommodating control system failures while maintaining good performance. Casalino, et al [35] presented the theory of another adaptive LQ technique that updates the previous controller based on the additional information obtained from the parameter estimates. Thus, a complete redesign of the controller at every iteration is avoided. This is accomplished by solving a variational calculus problem and is applicable to either finite or infinite horizon LQ optimization.

Another aircraft flight control design approach is eigenstructure assignment [36]. An attractive feature of the eigenstructure assignment is that the feedback stabilization and the feedforward performance design objectives are separated. Jiang [37] and Napolitano et al [38] explored the use of eigenstructure assignment for aircraft reconfiguration. The general result of their studies is that the closed loop system is guaranteed to be stable for full-state feedback, but for output feedback, stability is uncertain. Again, the stability guarantee assumes that the plant is known; no robustness issues associated with uncertainty in the plant model were addressed.

Dynamic Inversion, or feedback linearization, has been suggested as an indirect adaptive control

approach. Dynamic Inversion has gained acceptance as a viable nonlinear control method for designing aircraft control laws [36, 39]. The general theory of feedback linearization is well documented in Khalil [40] and Soltine [41]. In this indirect adaptive approach, the identified plant parameters would be used to update the inverse dynamics, thus achieving reconfiguration. As Brinker and Wise [42] point out, the pitch plane stability and flying qualities are robust to aerodynamic parameter uncertainties, but the lateral directional flying qualities are sensitive to these uncertainties.

The general area of linear model reference control has been suggested as a suitable indirect adaptive control approach. An advantage of using a reference model approach is that the closed-loop desired performance can be specified entirely by the reference model. Bodson [27] presents an indirect adaptive control structure that is suitable for aircraft reconfiguration. Furthermore, Bodson [43] addresses the issue of control surface saturation for the indirect adaptive model reference control architecture by devising an outer loop command limiter. Because a pseudo-inverse is generally involved to determine the control parameters from the estimated plant parameters, there is some uncertainty whether the closed-loop plant will follow the reference model. Conditions for perfect model following have been developed and are well documented [10, 44, 45]. Perfect model following completely specifies the behavior of the closed-loop system for all times.

There are many other fringe controller redesign techniques in the literature. Caliskan and Vepa [46] propose a mini-max approach for reconfiguration. The primary objective of the algorithm is to obtain stability once a failure has occurred by minimizing the risk of instability. Performance issues before, during and after the onset of failure were not addressed. Balakrishnan and Biega [47] present a technique of solving a nonlinear optimal control problem by casting the dynamic programming equations into a neural network framework. This method is not well suited for on-line computations because the number of iterations required for convergence is uncertain. Chandler, Mears, and Pachter [48] suggest the use of a Hopfield neural network as the controller redesign module. Using the identified model, the Hopfield network generates an optimal model following open-loop control law, which is calculated at every sample, thus yielding a feedback control action. Assuring stability for any of these techniques is problematic.

Control Allocation

Keeping the indirect control modular structure in mind, several techniques have been suggested for on-line control surface allocation to address the problem of changes in controller effectiveness and to address the problem of control surface constraints.

The pseudo inverse method described in Section 1.2.1 has been the standard method of solving the control allocation problem online due to the ease with which it can be implemented based on the estimated plant. However, the same deficiencies described in Section 1.2.1 pertain here. Bordignon and Durham [49] have devised an alternative technique based on an attainable moment set that reflects control surface position constraints. This general technique was extended to include control surface rate saturation in [50]. Enns [51] discusses several approximation approaches to the constrained weighted least-square problem of minimizing the difference between the desired and achievable moments. For the general nonlinear case, the exact solution to the least-square problem might be impractical for on-line implementation because of computational complexity; therefore, several approximations for the control constraints are suggested. Buffington [52] presents a control allocation technique specifically oriented to dynamic inversion control. The objective function is the 1-norm of the difference between the desired and achievable moments subject to control surface position limits, control law command prioritization, and control law axes prioritization.

1.2.3 Direct Adaptive Control

The direct adaptive control approach alters the parameters that define the controller directly, thus skipping the plant identification step. Many of the results in direct adaptive control are derived from Lyapunov stability theory. The textbooks by Narendra and Annaswammy [53] and Åstrom and Wittenmark [54] provide a thorough overview of direct adaptive control. The advantages of direct adaptive approaches are as follows:

- The method generally ensures close-loop stability for unforeseen events.
- Complete knowledge of the post-failure mathematical model is not required.

- Control surface constraints can be incorporated into the design.

The disadvantages of this approach are:

- Conditions for stability can be quite stringent depending on the formulation.
- A suitable Lyapunov function must be found to prove stability.
- Controllers are inherently nonlinear and non-autonomous; thus, analysis can be difficult.
- There is no general unified approach to these problems.

A promising technique for commercial reconfiguration is presented by Bodson [27, 28]. A model reference input error formulation is derived for a linear time-invariant aircraft reconfiguration problem. The significance of this derivation is that a stability proof exists which requires minor assumptions that the plant is minimum phase and that the high frequency gain is nonsingular. Furthermore, the resulting adaptation regressor is linear in control parameters, and efficient least-squares algorithms can be used to update the parameters. Finally, as Annaswamy et al [55, 56, 57] demonstrated, the closed-loop system remains stable in the presence of control surface position saturation. The stability analysis reveals that for a linear time-invariant plant, the closed-loop nonlinear non-autonomous system is *globally* stable if the open-loop plant is stable and is *locally* stable if the open-loop plant is unstable. They also derived conditions on the state amplitude for the locally stable case.

Another promising technique for reconfiguration is the application of Neural Networks (NN) for control systems. Narendra and Parthasarathy [58] introduce the use of neural networks for identification and control of dynamical systems in 1990. The general theory of neural networks for identification and control is well documented by Narendra [59]. Since Narendra's paper, there have been a number of flight control applications reported that illustrate the on-line learning capability of neural networks. Troudet et al [60] designed a model-following multi-layer neural network for an integrated airframe/propulsion model for a modern fighter aircraft. Simulation demonstrated that the tracking performance of the neural network was superior to a similarly design H_∞ controller; however, the neurocontroller had poor stability robustness characteristics. An interesting point made in this paper is that through the choice of inputs (which included

proportional, integral, and derivative tracking errors), the linearized NN exhibits PID control-type behavior.

A neurocontroller approach for flight control has also been demonstrated by Calise, McFarland, and Kim [61, 62, 63] within the framework of dynamic inversion. The architecture consists of two neural networks which invert the nonlinearities: the first neural network is trained off-line for the operational envelope, and the second neural network adapts online to correct the inversion error cause by the first neural approximation and by potential failures. Both networks are composed of radial basis functions and sigma-pi units. Of primary significance is a proof of stability that is presented under mild assumptions on the nonlinearities representing the inversion error [61]. This nonlinear flight control approach has been demonstrated through simulations for a fighter aircraft [61], high-performance missile [62, 63], and a helicopter [64]. With regards to aircraft reconfiguration, Kim and Calise [61] present simulation results for an F-18 aircraft with and without a 30% loss of left stabilator, and inside and outside of the first neural network training envelope. The results demonstrated the the aircraft response resembled first-order tracking for these scenarios. McFarland [65] has extended this adaptive neural approach for augmenting a gain-scheduled missile autopilot. This extension is a promising technique for commercial reconfiguration.

More recently, this adaptive neural network, dynamic inversion control architecture was applied to Boeing's Reconfigurable Control for Tailless Aircraft (RESTORE) Program [66]. The aircraft used in this program is the Tailless Advanced Fighter Aircraft (TAFA) which is similar in configuration to the X-36. It is worth noting that this aircraft has highly nonlinear aerodynamic relationships which can be a problem for the parameter estimation techniques outlined in Section 1.2.2 [24, 26]. Calise et al [67] demonstrated through simulations that this approach was able to accommodate a large set of damage scenarios and failures while maintaining acceptable handling qualities. Based on this success, the RESTORE program is transitioning into the Self Adaptive Flight Control Experiment (SAFE) program to perform flight demonstrations of this reconfiguration technique. The X-36 will be used as the flight demonstration vehicle.

Yet another application of a neurocontroller within the framework of dynamic inversion is presented by Lin et al [68]. In this paper, a dynamic inversion neurocontroller scheme was used to control the depth and pitch of a submarine. A Fuzzy Cerebellar Model Arithmetic Computer (FCMAC) neural network was used to represent the inverse dynamics. The advantage of the FCMAC network is that it has learning rates that are at least an order of magnitude faster than conventional neural networks.

There are other nonlinear direct adaptive design techniques that show potential for aircraft reconfiguration, two of which are Sliding Mode Control (SMC) [41] and Backstepping [69]. These methods are recent advances in the field of nonlinear control, and their application to RFCS have not been fully explored. SMC is an attractive approach for aircraft reconfiguration because the method is inherently robust to plant uncertainties and because actuator constraints can be handled directly. Shtessel et al [70] propose a two loop SMC for a reconfigurable flight control system. The inner loop SMC provides robust performance of angular rate commands while the outer loop provides robust tracking of commanded mission angles such as angle of attack. Reconfiguration simulations were performed for an F-16; the controller demonstrated high accuracy tracking for a 50% loss of horizontal tail area.

1.3 Retrofit Module Requirements

Given the advantages and disadvantages of the available technologies outlined in the above section, a list of requirements for a low-cost retrofit reconfiguration module for commercial transports was compiled. The basic objective function of the problem statement is to maximize reconfiguration capability while minimizing the implementation and lift-cycle cost given the retrofit concept. This mini-max viewpoint results in a set of requirements that can be separated into two categories, architectural and algorithm requirements; the architectural requirements attempt to minimize the cost while the algorithms attempt to maximize the reconfiguration functionality.

Viewing the advantages and disadvantages of the available technologies, we desire a reconfig-

uration algorithm that can accommodate unforeseen failures, adapt quickly to destabilizing moments, and re-trim the vehicle post failure. Additionally, we desire a reconfiguration algorithm that provides adequate command following and disturbance rejection post failure. The algorithm must account for the realities of flight which include model uncertainty, control surface constraints, transport delays, sensor noise, and process noise. Finally, a proof of stability is sought that will give some assurance of how the algorithm will behave under realistic assumptions. Clearly, these algorithm requirements constitute a high degree of reconfiguration functionality.

Within the reconfiguration algorithm requirements, architectural requirements are sought to minimize implementation and life-cycle costs. Obviously, we desire a reconfiguration architecture that is applicable to a variety of aircraft, i.e., commercial transports, business jets, fighter aircraft. Thus, the structure must be independent of the existing control system. Also, we desire to implement the approach on existing aircraft; thus, the architecture must be retrofittable to existing control systems. As noted in the previous section, one of the major problems with existing reconfiguration techniques is that they are not able to recover nominal performance during healthy operations. Thus, we desire an architecture that does not interfere with the nominal control system when no failure is present.

The algorithm and architectural requirements are summarized as follows.

Algorithm Requirements

- Accommodate Unforeseen Failures
- Adapt Quickly
- Re-trim Vehicle Post-Failure
- Adequate Command Following Post-Failure
- Adequate Disturbance Rejection Post-Failure
- Robust to Model Uncertainty
- Robust to Control Surface Constraints
- Robust to Transport Delays
- Robust to Process and Sensor Noise
- Ensure Stability Post-Failure

Architecture Requirements

- Applicable to a Variety of Aircraft
- Independent of Existing Control System
- Applicable to Existing Aircraft
- Exhibit Nominal Performance without Failure
- Minimize Complexity

1.4 Proposed Retrofit Module Architecture and Algorithms

Based on the problem statement, literature review, and the above requirements, a direct adaptive approach using a model reference control architecture was selected. A direct adaptive approach was desirable because these methods can account for unforeseen events while eliminating the bottleneck problems of on-line system identification associated with the indirect approach. Furthermore, direct methods account for control surface constraints which, if not addressed directly, can result in significant degradation of tracking performance up to the loss of stability. Figure 1-2 illustrates the proposed architecture, which consists of three components: adaptive augmentor, reference model, and signal conditioning and estimation. Each of these components will be discussed as they pertain to the requirements outlined in the previous section. As mention previously, the inputs to the retrofit system are pilot inputs $r(t)$, aircraft state information $y(t)$, existing control law commands u_{k_c}, and the additional input is the reconfiguration control module commands u_{rcm_c}.

The adaptive augmentor component is the primary algorithm that determines the reconfiguration strategy. A direct adaptive, input error formulation similar to Bodson's [27] was selected because of its simplicity. An important observation about commercial transports — with current stability augmentation and under most maneuvering situations — is that they can be accurately modeled by linear time-invariant systems with slowly varying, flight-condition dependent coefficients. This fact makes the problem more tractable, so that more sophisticated nonlinear design techniques such as neurocontroller approaches are not warranted. The input error formulation

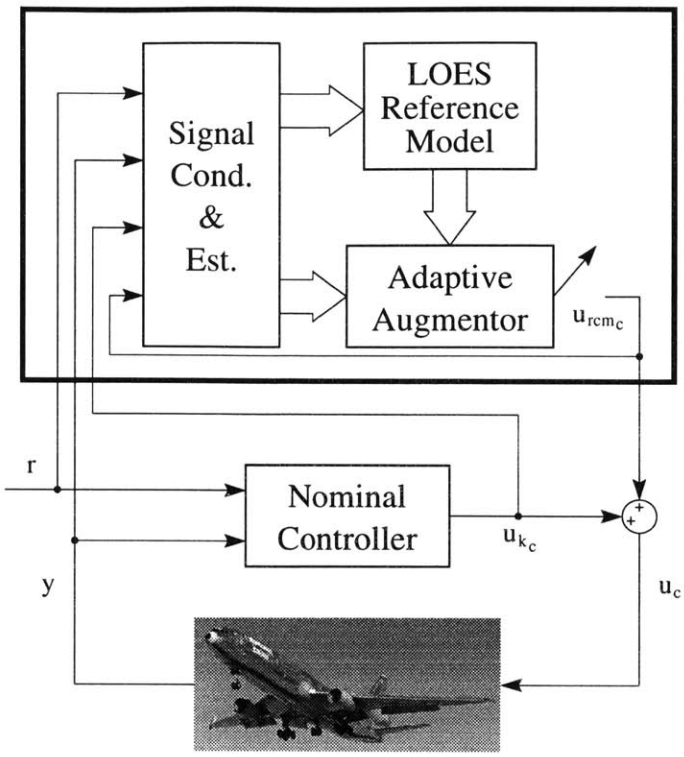

Figure 1-2: Adaptive Control Architecture for Reconfiguration Module

is attractive because the resulting regressor is linear; thus relatively efficient least-squares algorithms can be used to perform the control parameter updates [28, 29]. This satisfies the rapid adaptation requirement. Another attractive feature of the model reference approach presented by Bodson is that it includes a 'biased term' in the control law to accommodate failure disturbances and unmodeled dynamics. Also, a proof of stability in the presence of control position saturations exist [55, 56, 57]. Techniques to robustify this algorithm due to sensor and process noise also exist [2, 53].

Given these available technologies, several extensions will be required to apply them to the propose framework. The formulation must be extended for the closed-loop, retrofit architecture that tracks the desired trim condition. Additionally, methods that account for control surface position and rate saturation need to be addressed. Finally, the proof of stability needs to be extended to the MIMO, closed-loop formulation with rate and position saturation while addressing modeling uncertainty and sensor and process noise.

A logic switch is included in the adaptive augmentor component to ensure that nominal performance is achieved during healthy operation. Even thought the reconfiguration command must naturally exhibit zero or very small outputs when there is no failure, the switch ensures that no signals enter the closed-loop system until a failure is declared. Also, the switch allows the pilots to manually disengage the retrofit module. This logic could be as simple as a threshold on the augmentation signal. Such a threshold would also ensure that small variations in the augmentation signal, induced by reference model errors, which might occur during changes in flight conditions, do not enter the control loop, thus preserving the nominal performance.

Given that this is a model reference control framework, a reference model is required. The reference model provides the desired dynamics of the healthy aircraft to the adaptive augmentor component. The proposed reference model has two unique properties. The first property is that it is constructed as a flight-condition dependent, low-order equivalent system (LOES) representation of the aircrafts unfailed closed-loop behavior. This dependence can be captured using conventional means (table look-up or curve fits) or using neural networks that are trained off-line [59].

The simplicity of the reference model structure, together with the desire to minimize off-line analysis, implementation, and life cycle cost, motivates the second property of the reference model, which is adaptation. Very slow adaptation, with bounds on the allowable deviation of parameters from the nominal model, will be used to adjust the model during unfailed operations. Behavior that does not precisely match a LOES model, as well as gradual changes in the aircrafts behavior are thus not seen as failures, and cannot trip command augmentation. When a failure is detected, adaption of the reference model is halted. The reference model is then frozen at the most recent unfailed performance (perhaps modified very slightly by the effect of the failure during the short period when it went undetected). If the flight condition subsequently changes, the most recently adapted model at the new flight condition will be used. This is retained by proper construction of the adaptation algorithm and parameterization.

The signal conditioning and module is the next component. The primary function of this component is to provide the required signals to both the adaptive augmentor and reference

model. Many assumptions will be made about the available data to make simplifications to the other two components. Given that some of this data is not available on existing commercial transports, these assumptions will result in greater complexity for this component. Depending on the available sensor suites, Kalman filters and control surface models might be required to generate all of the data needed to accomplish the reconfiguration task.

1.5 Thesis Overview

The goal of this research is to develop and demonstrate a reconfiguration strategy that has the potential of improving aviation safety while keeping cost and complexity low. In doing so, the retrofit architecture illustrated in the previous section must be developed to satisfy the stated requirements in Section 1.3. To accomplish this, theoretical extensions are required to the current technology. The first part of this thesis presents the theoretical development of the retrofit module. Chapter 2 presents the adaptive augmentor formulation. Next, the reference model development is presented in Chapter 3. Finally, the signal conditioning and estimation component is outlined in Chapter 4.

Given the theoretical development, the reconfiguration potential needs to be verified in simulations. The second part of the thesis outlines the implementation of the retrofit module into two aircraft. Chapter 5 presents the implementation and simulation results for a F/A-18 fighter. Next, Chapter 6 presents the implementation in a narrow-body generic commercial transport. The particular simulation is the Advanced Concept Flight Simulator (ACFS) which resides at NASA Dryden Flight Research Center. This chapter is followed by the thesis conclusions and recommendations.

THIS PAGE LEFT BLANK INTENTIONALLY

Part I

Methodology

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 2

Adaptive Augmentor Module

In this chapter, we review the direct input error adaptive control algorithm that we have implemented for the 'Adaptive Augmentor' shown in Figure 1-2. The adaptive input error formulation was selected for the reasons stated in the previous section; other adaptive algorithms, i.e. output error and neurocontroller formulations, would still fit into the retrofit framework that has been suggested.

2.1 Closed-Loop Aircraft Model

The nonlinear, rigid-body, 6 degree-of-freedom, flat-earth equations of motion in stability axis coordinate system are well documented in [71] and can be represented in a general form:

$$\begin{aligned} \dot{x}_p &= j_p(x_p, \dot{x}_p, u, d) \\ x_{p_m} &= x_p + v \\ z &= C x_{p_m} \end{aligned} \tag{2.1}$$

where the state x_p and the measured state $x_{p_m} \in \mathbb{R}^n$, the implemented control $u \in \mathbb{R}^q$, the atmospheric disturbance $d \in \mathbb{R}^n$, the performance $z \in \mathbb{R}^m$, and the sensor noise $v \in \mathbb{R}^n$. It will be assumed that the performance z contains the rotational degrees of freedom:

Assumption 1 *The performance vector contains the rotational rates, p, q and r.*

This is a reasonable assumption considering that control surface failures produce undesirable rotational accelerations.

Linearizing the above equation, the aircraft model can be represented as a linear time-invariant system with slowly varying, flight-condition dependent coefficients and a nonlinear vector which represents higher-order dynamics:

$$\begin{aligned} \dot{x}_p &= A_p\, x_p + B_p\, u + f_p + w + g_p(x_p, \dot{x}_p, u, d) \\ x_{p_m} &= x_p + v \\ z &= C\, x_{p_m} \end{aligned} \tag{2.2}$$

where the dimensions are consistent with (2.1) and the disturbance vectors f_p and $w \in \mathbb{R}^n$ and the nonlinear vector $g_p \in \mathbb{R}^n$. As in (2.1), the plant state x_p represents the total state instead of perturbation state. The disturbance term f_p accounts for the trim status and potential failure disturbances, and w represents the linear part of d. Furthermore, g_p accounts for the higher-order nonlinearities in (2.1) such as inertial coupling, nonlinear aerodynamics, etc. In addition, we make the following assumptions:

Assumption 2 *The full state is measurable.*

Assumption 3 *The performance vector derivative $\dot{z}$ is measured or estimated.*

Assumption 4 *A static equilibrium condition exists $\forall\ t \geq 0$.*

Assumption 5 *The aircraft is stabilizable post failure.*

Assumption 6 *The state vector is detectable via the performance vector.*

Stability augmentation systems can be represented in a general form:

$$\begin{aligned} \dot{x}_k &= j_k(x_k, \dot{x}_k, x_{p_m}, r) \\ u_{k_c} &= l_k(x_k, x_p, r) \end{aligned} \tag{2.3}$$

where $x_k \in \mathbb{R}^v$, $x_{p_m} \in \mathbb{R}^n$, $u_{k_c} \in \mathbb{R}^q$, and $r \in \mathbb{R}^m$. Within a given mode of operation and under normal operating conditions, the flight control system can be modeled in a state-space format:

$$\begin{aligned} \dot{x}_k &= A_k\, x_k + B_{kx}\, x_{p_m} + B_{kr}\, r + f_k + g_k(x_k, \dot{x}_k, x_{p_m}, r) \\ u_{k_c} &= C_k\, x_k + D_{kx}\, x_{p_m} + D_{kr}\, r + h_k(x_k, \dot{x}_k, x_{p_m}, r) \end{aligned} \tag{2.4}$$

where the dimensions are consistent with (2.3) and the disturbance term $f_k \in \mathbb{R}^v$ and the higher-oder nonlinearities $g_k \in \mathbb{R}^v$ and $h_k \in \mathbb{R}^q$.

Knowing that the reconfiguration architecture adds perturbation commands to a subset of controller commands, the combined control command is as follows:

$$u_c = \begin{bmatrix} u_{k_c}(1) + u_{rcm_c}(1) \\ \vdots \\ u_{k_c}(m) + u_{rcm_c}(m) \\ u_{k_c}(m+1) \\ \vdots \\ u_{k_c}(q) \end{bmatrix} \tag{2.5}$$

Assumption 7 *The reconfiguration command u_{rcm_c} has the same dimension as the performance z and the reference command r, i.e. $u_{rcm_c} \in \mathbb{R}^m$.*

The flight control surface behavior, including servos and limits, can be represented in a general form:

$$\begin{aligned} \dot{x}_a &= j_a(x_a, u_c, \dot{u}_c) \\ u &= l_a(x_a, u_c, \dot{u}_c) \end{aligned} \tag{2.6}$$

where $x_a \in \mathbb{R}^w$ and u, u_c, and $\dot{u}_c \in \mathbb{R}^q$. The primary control surface nonlinearity is caused by deflection magnitude and rate saturation. The magnitude and rate constraints on the control

surfaces are of the form:

$$\begin{aligned} |u_i| &\leq u_{o_i} \\ |\dot{u}_i| &\leq \dot{u}_{o_i} \end{aligned} \tag{2.7}$$

where $i = 1 \ldots q$ (it is not necessary to assume that these limits are symmetric). Under normal operating conditions, the actuator saturation constraints can be quite significant; thus, an alternative model to (2.6) can be represented in state-space format by grouping the saturation nonlinearities Δu in the output equation and the piston/linkage higher-order nonlinearities g_k in the state equation as in [55, 56, 57]:

$$\begin{aligned} \dot{x}_a &= A_a x_a + B_a u_c + f_a + g_a(x_a, u_c, \dot{u}_c) \\ u &= C_a x_a + \Delta u \\ &= u_k + u_{rcm} + \Delta u \end{aligned} \tag{2.8}$$

where the dimensions are consistent with (2.6), the disturbance vector $f_a \in \mathbb{R}^w$, the higher-order nonlinearities $g_a \in \mathbb{R}^w$, and $\Delta u \in \mathbb{R}^q = u - u_k - u_{rcm}$ equals the difference between the actual control surface deflection u and the unsaturated control surface deflection $C_a x_a = u_k + u_{rcm}$.

Assumption 8 *The control surface saturation Δu is computable, e.g. a surface model or direct measurement is available.*

Using equations (2.2,2.4,2.5, and 2.8) the closed loop system can be constructed where the combined state $x \in \mathbb{R}^{n+v+w}$.

$$
\begin{aligned}
\begin{bmatrix} \dot{x}_p \\ \dot{x}_k \\ \dot{x}_a \end{bmatrix} &= \begin{bmatrix} A_p & 0 & B_p C_a \\ B_{kx} & A_k & 0 \\ B_a D_{kx} & B_a C_k & A_a \end{bmatrix} \begin{bmatrix} x_p \\ x_k \\ x_a \end{bmatrix} + \begin{bmatrix} 0 \\ B_{kr} \\ B_a D_{kr} \end{bmatrix} r \\
&+ \begin{bmatrix} 0 \\ 0 \\ B_a(:, 1:m) \end{bmatrix} u_{rcm_c} + \begin{bmatrix} B_p \\ 0 \\ 0 \end{bmatrix} \Delta u + \begin{bmatrix} f_p \\ f_k \\ fa \end{bmatrix} \\
&+ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} w + \begin{bmatrix} 0 \\ B_{kx} \\ B_a D_{kx} \end{bmatrix} v + \begin{bmatrix} g_p \\ g_k \\ g_a + h_k \end{bmatrix} \\
z &= \begin{bmatrix} C & 0 & 0 \end{bmatrix} \begin{bmatrix} x_p \\ x_k \\ x_a \end{bmatrix}
\end{aligned}
\tag{2.9}
$$

In general, the purpose of the feedback augmentation is to alter the aircraft's open-loop pole and zero locations so that flying qualities are satisfied. For our design purpose, it would be cumbersome to retain the states of the feedback augmentation and actuators in a design model; thus, we desire to have a reduced dimension model with only the aircraft states x_p. As illustrated in [71], the effect of the additional dynamics resulting from the feedback augmentation and actuator dynamics can be allowed for by determining an 'equivalent low-order system' of the form:

$$
\begin{aligned}
\dot{x}(t) &= A\,x(t) + B_r\,r(t-\tau) + B_u u_{rcm_c}(t) \\
&+ B_p \Delta u(t) + f + B_w w(t) + B_v v(t) \\
&+ g(t) \\
z(t) &= C\,x(t)
\end{aligned}
\tag{2.10}
$$

where $x \in \mathbb{R}^n$, and the rest of the dimensions are consistent with the state dimensions. The delay τ is included in the low-order design model to provide an equivalent time delay for matching high-frequency effects for reference command inputs. It is worth noting that the transfer function

from the input u_{rcm_c} to the performance z is 'square', and that Δu is measurable or computable but w, v, and g are not. Furthermore, the model reduction error has been grouped into the nonlinear vector g.

2.2 Closed-Loop Desired Dynamics

The reference model for this design is chosen to have the same dimensions as the closed-loop design model:

$$\begin{aligned} \dot{x}_m(t) &= A_m\, x_m(t) + B_m\, r(t-\tau) + f_m \\ z_m(t) &= C\, x_m(t) \end{aligned} \tag{2.11}$$

where $x_m, f_m \in \mathbb{R}^n$ and $r, z_m \in \mathbb{R}^m$ where $m \leq n$. Again, the desired state x_m is a total signal: perturbation plus trim.

Note that the reference model only reflects the linear dynamics of the design model. At a given flight point, the system matrices A_m and B_m reflect the nominal closed-loop handling qualities. In fact, the flying-qualities specifications — time constants, natural frequencies, and damping ratios — can be inferred from these system matrices. Also note that the desired disturbance f_m reflects the nominal trim status. By including this information, we ensure that the reconfiguration module will try to maintain the nominal trim attitude post failure. Finally, because the reference model does not contain the actuator saturation, noise, or modeling nonlinearity terms, special consideration must be made to account for these affects in the adaptation algorithm.

2.3 Model Reference Control Law

The goal of the MRAC is to make the system matrices A, B_r, and f in (2.10) look like the system matrices A_m, B_m, and f_m in (2.11). The control law chosen has the following form:

$$u^*_{rcm_c}(t) = K_x\, x(t) + K_r\, r(t-\tau) + K_f \tag{2.12}$$

where $K_x \in \mathbb{R}^{m\times n}, K_r \in \mathbb{R}^{m\times m}$ and $K_f \in \mathbb{R}^m$. Substituting the optimal control law into the design model and equating with the reference model, the algebraic conditions for model matching are obtained:

$$\begin{aligned} K_x &= (CB_u)^{-1}(CA_m - CA) \\ K_r &= (CB_u)^{-1}(CB_m - CB_r) \\ K_f &= (CB_u)^{-1}(Cf_m - Cf) \end{aligned} \tag{2.13}$$

Note that when the design model system matrices equal the reference model, the model reference control law is zero, i.e. this architecture is nonintrusive. Also, the algebraic conditions for model matching in (2.13) provide the information about the existence and uniqueness of the optimal gains. The assumptions required for this solution to be feasible are as follows:

Assumption 9 *The control surfaces used for reconfiguration must be unique and must have adequate control power post failure, i.e.* $(CB_u)^{-1}$ *must be nonsingular.*

Assumption 10 *The reference model must be stable, i.e.* A_m *is Hurwitz.*

Assumption 11 *The design model must be minimum phase to eliminate the potential of right-half-plane pole-zero cancellations.*

During post failure operations, the optimal gains are unknown because the failed design model (2.10) is uncertain. Thus, the reconfiguration module generates a perturbation command that is based on the best guess of the optimal gains at the current time:

$$u_{rcm_c}(t) = \hat{K}_x(t)\, x(t) + \hat{K}_r(t)\, r(t-\tau) + \hat{K}_f(t) \tag{2.14}$$

where the $\hat{}$ denotes an estimate. The control parameters in (2.14) are updated based on an input error formulation that will be derived in the next section.

2.4 Robust Adaptive Algorithm

The goal of the adaptive algorithm is to ensure that the gains approach their optimal values in a well-behaved manner over time. Furthermore, given the limitations of the reference model, the adaptation must be robust to uncertainties. In this section, the input error formulation will be derived.

A perturbation control input identity can be obtained by computing $\dot{z}$ from (2.10) and subtracting the known vectors $C(A_m x + B_m r + f_m)$ from both sides:

$$\begin{aligned} \dot{z} - C(A_m x + B_m r + f_m) &= C(A - A_m)x + C(B_r - B_m)r + C(f - f_m) \\ &+ CB_u u_{rcm_c} + CB_p \Delta u + CB_w w + CB_v v + Cg \end{aligned}$$

Note, the arguments have been dropped for compactness. Rearranging allows an expression for u_{rcm} to be written

$$\begin{aligned} u_{rcm_c} &= -(CB_u)^{-1}C(A - A_m)x - (CB_u)^{-1}C(B_r - B_m)r - (CB_u)^{-1}C(f - f_m) \\ &+(CB_u)^{-1}\left[\dot{z} - C(A_m x + B_m r + f_m + B_p \Delta u)\right] - V \end{aligned}$$

where $V = (CB_u)^{-1}(CB_w w + CB_v v + Cg)$ represents the higher-order nonlinerities, atmospheric disturbances, sensor noise, and model reduction errors. Using this expression in equation (2.13), and rearranging:

$$u_{rcm_c} = K_x x + K_r r + K_f + (CB_u)^{-1}[\dot{z} - C(A_m x + B_m r + f_m + B_p \Delta u)] - V$$

This expression is true *for all time for any* u_{rcm_c}, and does not represent the ideal control; it is the actual reconfiguration input in terms of quantities that are known to exist. Perfect tracking is achieved when the bracketed term equals zero, so that $u_{rcm_c} = u^*_{rcm_c}$ (see (2.12)). The bracketed term is zero when the closed-loop state trajectories satisfy the reference model performance equality in equation (2.11) and when V is insignificant.

To form an error on which to adapt, we can also write an expression based on the *current estimates* of the gain matrices and $(CB_u)^{-1}$. Define the 'modified control input' as:

$$\acute{u}_{rcm_c} = \hat{K}_x x + \hat{K}_r r + \hat{K}_f + \hat{J}\left[\dot{z} - C(A_m x + B_m r + f_m + B_p \Delta u)\right] \tag{2.15}$$

where $\hat{J}$ is the current estimate of $(CB_u)^{-1}$.

Using the modified control input, we can define the input error as:

$$\begin{aligned} e_i &= \acute{u}_{rcm_c} - u_{rcm_c} \\ &= \tilde{K}_x x + \tilde{K}_r r + \tilde{K}_f + \tilde{J}\left[\dot{z} - C(A_m x + B_m r + f_m + B_p \Delta u)\right] + V \end{aligned} \tag{2.16}$$

where $\tilde{K}_x = \hat{K}_x - K_x$, $\tilde{K}_r = \hat{K}_r - K_r$, $\tilde{K}_f = \hat{K}_f - K_f$, and $\tilde{J} = \hat{J} - (CB_u)^{-1}$ are the parameter errors. If the control parameters in (2.14) converge to (2.13), if J converges to $(CB_u)^{-1}$, and if the adaptive algorithm is robust to V, the control parameters will remain at their optimal values regardless to the saturation level, i.e. $\Delta u \neq 0$. In other words, the presence of saturation will not affect the convergence of the estimates.

Another useful realization of the input error can be obtained by subtracting (2.15) from (2.14):

$$\begin{aligned} e_i &= \acute{u}_{rcm_c} - u_{rcm_c} \\ &= \hat{J}\left[\dot{z} - C(A_m x + B_m r + f_m + B_p \Delta u)\right] \\ &= \hat{J} e_o \end{aligned} \tag{2.17}$$

where the output performance error e_o is defined as follows:

$$e_o = \dot{z} - C(A_m x + B_m r + f_m + B_p \Delta u) \tag{2.18}$$

Thus, the input error can be view as the output error performance e_o — the error between the actual and desired aircraft's performance — scaled to produce units of pseudo actuator deflection.

In order to calculate the input error, the control surface saturation signal must be calculated

$$\Delta u = u - (u_k + u_{rcm})$$

where u is the actual control surface deflection and $u_k + u_{rcm}$ is the unsaturated control surface deflection in (2.8). Both of these deflections are computed via a surface model and/or direct measurement.

Given that the input error is computable, a linear regressor can be constructed in terms of parameter errors

$$\begin{aligned} e_i &= \left[\tilde{K}_x \ \tilde{K}_r \ \tilde{K}_f \ \tilde{J}\right] \begin{bmatrix} x \\ r \\ 1 \\ \dot{z} - C(A_m x + B_m r + f_m + B_p \Delta u) \end{bmatrix} + V \\ e_i &= \Phi Z + V \end{aligned} \tag{2.19}$$

where $Z \in \mathbb{R}^{n+m+1+m}$ and $\Phi \in \mathbb{R}^{m \times (n+m+1+m)}$. It is worth mentioning that if $(CB_u)^{-1}$ is known *a priori*, then $\tilde{J} = 0 \ \ \forall \ t \geq 0$ and this this term can be eliminated from the regressor.

For this multivariable linear error model, a robust, stable adaptation law [53] is:

$$\dot{\Phi}(j,:) = \begin{cases} -e_i(j) Z^T \Gamma & \text{if } |e_i(j)| \geq V_o(j) \\ 0 & \text{else} \end{cases} \tag{2.20}$$

for $j = 1 : m$ and where $|V| \leq V_o$ is the upper bound on the regressor disturbance, and the adaptation gain Γ is a symmetric positive definite.

2.5 Asymptotic Properties

As mentioned previously, direct adaptive approaches are attractive because stability proofs are generally obtainable. In this section, the asymptotic properties of the closed-loop system with

the reconfiguration module will be explored under the previously mentioned assumptions and the following additional assumptions:

Assumption 12 *The open-loop aircraft (2.1) is stable.*

Assumption 13 *The pilot input r is uniformly continuous and bounded.*

Assumption 14 *The regressor disturbance V is uniformly continuous and bounded.*

Assumption 12 can be restrictive; current research is in progress to extend the asymptotic results to open-loop unstable aircraft.

The asymptotic properties are developed using the principle of Barbalat's Lemma [53, 41]. To do this, we first define a Lyapunov function for the parameter errors:

$$W = \frac{1}{2}\mathrm{Tr}[\Phi\Gamma^{-1}\Phi^T] \tag{2.21}$$

This equation is lower bounded, i.e. $W \geq 0$. Taking the derivative of (2.21) along the trajectory and inserting the robust adaptation law (2.20), we have that

$$\dot{W} = \begin{cases} -e_i(e_i - V) & \text{if } |e_i| \geq V_o \\ 0 & \text{else} \end{cases} \tag{2.22}$$

where again $|V| \leq V_o$ is the upper bound on the regressor disturbance. This implies that $\dot{W}$ is negative semi-definite, i.e. $\dot{W} \leq 0$.

Next, we need to show that $\dot{W}$ is uniformally continuous in time, i.e. $\ddot{W}$ is bounded. This requires that e_i and $\dot{e}_i$ be bounded. Given Assumptions 12, 4, and 5, the state and performance vectors are bounded, i.e. $x, z \in \mathcal{L}^\infty$, and the saturation and saturation rate Δu and $\dot{\Delta} u$ are bounded, i.e. $\Delta u, \dot{\Delta} u \in \mathcal{L}^\infty$. Also, $\Phi \in \mathcal{L}^\infty$ as a result of (2.21) and (2.22). Since $x, r, \Phi \in \mathcal{L}^\infty$ and the optimal gains (2.13) exist, the command control u_{rcm_c} (2.14) is bounded, i.e. $u_{rcm_c} \in \mathcal{L}^\infty$. It then follows that the state and performance derivatives are bounded, i.e. $\dot{x}, \dot{z} \in \mathcal{L}^\infty$. Given that

$x, r, \dot{z}$ and $\Delta u \in \mathcal{L}^\infty$, the regressor vector Z is bounded. Given that $\Phi, Z,$ and $V \in \mathcal{L}^\infty$, the input error e_i is bounded, i.e. $e_i \in \mathcal{L}^\infty$.

To show that $\dot{e}_i \in \mathcal{L}^\infty$, we must show that $\ddot{x}$ is bounded. First, given that $x, r, \dot{x}, \dot{r}, \Phi$ and $\dot{\Phi} \in \mathcal{L}^\infty$, then $\dot{u}_{rcm_c} \in \mathcal{L}^\infty$. Thus, since $\dot{x}, \dot{r}, \dot{u}_{rcm_c}, \dot{\Delta u}$ and $\dot{V} \in \mathcal{L}^\infty$, then from (2.10), $\ddot{x} \in \mathcal{L}^\infty$ which implies that $\dot{Z} \in \mathcal{L}^\infty$. Given that $\Phi, Z, \dot{\Phi}, \dot{Z},$ and $\dot{V} \in \mathcal{L}^\infty$, the input error derivative $\dot{e}_i$ is bounded, i.e. $\dot{e}_i \in \mathcal{L}^\infty$.

With the $e_i, V, \dot{e}_i,$ and $\dot{V} \in \mathcal{L}^\infty$, the derivative of the Lyapunov function $\dot{W}$ is uniformly continuous in time, i.e.

$$\ddot{W} \in \mathcal{L}^\infty \tag{2.23}$$

From Barbalat's Lemma, given (2.21 – 2.23)

$$\dot{W} \to 0 \text{ as } t \to \infty$$

which implies

$$\lim_{t\to\infty} |e_i| < V_o \quad \text{and} \quad \lim_{t\to\infty} \dot{\Phi} = 0$$

Furthermore, if Z is persistently excited and if the level of persistent excitation is large relative to V — that is, there is adequate signal-to-noise ratio — then

$$\lim_{t\to\infty} \Phi = 0.$$

Showing that $|e_i| < V_o$ as $t \to \infty$ ensures that the performance variables in z are driven to the values given by the reference model, i.e. $|z_m - z| \propto V_o$. Clearly, as V_o gets large, the achievable performance decreases.

2.6 Summary

In this chapter, the input error adaptive control algorithm for the model reference control architecture was presented. The input error was derived for a closed loop, low-order equivalent system. In one formulation, it was shown that the input error is linearly dependent in parameter errors, and a robust, stable adaptation law was suggested to update the unknown parameters. In another formulation, the input error was shown to be equivalent to the output performance error scaled to produce units of pseudo actuator deflection. Finally, asymptotic properties of the adaptation algorithm demonstrated that as $t \to \infty$, the input error would converge to the upper bound of the regressor noise boundary layer, i.e. $|e_i| < V_o$. Thus, the output performance error $|e_o| < \hat{J}^{-1}V_o$, which implies that the aircraft will follow the desired dynamics with the largest tracking error of $\hat{J}^{-1}V_o$.

In the following two chapters, the other two components of the RCM will be discussed. The primary functionality of the other two components is to provide the adaptive augmentor with the required information to compute the reconfiguration control solution.

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 3

Reference Model Module

The reference model is the second component of the RCM architecture. The reference model's primary function is to provide desired dynamics CA_m, CB_m, and Cf_m and an estimated of the aircraft's control power effectiveness CB_p to the adaptive augmentor component for the calculation of the input error. Recall from the previous chapter, the input error e_i, defined in (2.17), is simply the error between the actual and desired aircraft's performance, scaled to produce the units of pseudo actuator deflection:

$$\begin{aligned} e_i(t) &= \acute{u}_{rcm_c}(t) - u_{rcm_c}(t) \\ &= \hat{J}(t)\left[\dot{z}(t) - C(A_m x(t) + B_m r(t-\tau) + f_m + B_p \Delta u(t))\right] \\ &= \hat{J}(t) e_o(t) \end{aligned}$$

where the output performance error e_o was defined in (2.18) as follows:

$$e_o(t) = \dot{z}(t) - C[A_m x(t) + B_m r(t-\tau) + f_m + B_p \Delta u(t)]$$

As stated in Section 1.3, the input error is the key measure for updating the reconfiguration gains during failure operations. The previous chapter discussed the use of the input error for updating the reconfiguration gains. Also stated in Section 1.3, the input error is the key measure

for determining whether a failure has occurred. In order to use the input error as a measure for failure detection, the input error (or equivalently the output error) needs to be near zero for nominal operation. To achieve a near zero output error throughout the flight envelope, the reference model must be flight condition dependent. Given the additional requirements for low-cost implementation and life cycle maintenance, the reference model is to be generated on-line using system identification.

In this implementation, the objective of the on-line system identification is to minimize the mean-squared output performance error throughout the flight envelope:

$$\min_{\hat{\Theta}(\rho)} J[\hat{\Theta}(\rho)] = \lim_{T\to\infty} \frac{1}{T} \sum_{t\geq 0}^{T} e_o[t|\hat{\Theta}(\rho)]^2 \tag{3.1}$$

where the parameters $\hat{\Theta}(\rho)$ are the estimated desired dynamics — $\hat{CA}_m, \hat{CB}_m, \hat{Cf}_m$ and $\hat{CBp}$ — and the parameter ρ represents the flight condition dependency. In the sections that follow, a two stage identification process will be outlined that minimizes (3.1) during nominal operation. The data set and model structure selection will be detailed. Furthermore, procedures for determining the 'best' model from the model structure will be presented for both off-line and on-line analysis.

The use of system identification theory for constructing aircraft models is well documented. Ljung [17] and Mendel [18] contain a general overview of the system identification process, and a thorough overview of aircraft parameter estimation is contained in Maine and Iliff [72, 73]. Recent advances in on-line system identification will be utilized; specifically, the modified sequential least-square (MSLS) technique [19, 20, 21, 22, 23, 24] is used to generate the desired information on-line.

3.1 Identification Data Set and Model Structure

In this section, the model structure used to minimize the output error in (3.1) is formulated. The model structure represents a set of models that conform to a specific structure, i.e. dimensions and input/output relationships. As illustrated in Ljung [17], there are many different types of

model structures that can be used to obtain the desired information, but the model structure selected depends on the available data, i.e. the data set.

The adaptive augmentor formulation section listed several assumptions pertaining to the data set.

Assumption 2 *The full state is measurable.*

Assumption 3 *The performance vector derivative $\dot{z}$ is measured or estimated.*

Assumption 8 *The control surface saturation Δu is computable, e.g. a surface model or direct measurement is available.*

Since the data set contains states $x(t)$, performance derivatives $\dot{z}(t)$, pilot inputs $r(t-\tau)$, and saturation level $\Delta u(t)$, an Output-Error (OE) identification model structure can be used. The benefits of the OE model structure is that the resultant regressor is linear in unknown parameters. In contrast, if only state and pilot input data are used, an AutoRegressive Moving Average with eXternal input (ARMAX) model structure is require, and the resultant regressor is nonlinear in unknown parameters.

Given the assumptions on the data set, an OE model structure can be constructed, noting that the desire to make the output performance error e_o equal to zero is equivalent to:

$$\dot{z}(t) = CA_m(\rho)x(t) + CB_m(\rho)r(t-\tau) + Cf_m(\rho) + CB_p(\rho)\Delta u(t) \tag{3.2}$$

Since all of the signals, $\dot{z}(t), x(t), r(t-\tau)$, and $\Delta u(t)$, are known, each row of $\dot{z}(t)$ can be treated as an independent measurement. Thus,

$$\dot{z}(i;t) = CA_m(i,:;\rho)x(t) + CB_m(i,:;\rho)r(t-\tau) + Cf_m(i;\rho) + CB_p(i,:;\rho)\Delta u(t) \tag{3.3}$$

where the index i represents the i^{th} row the the matrix or vector and ranges from $1:m$, and colon index : represents all columns of the respective matrix. Rearranging the above equation,

the OE model structure is obtained:

$$\dot{z}(i;t) \;=\; \begin{bmatrix} x^T(t) & r^T(t-\tau) & 1 & \Delta u^T(t) \end{bmatrix} \begin{bmatrix} CA_m(i,:;\rho)^T \\ CB_m(i,:;\rho)^T \\ CF_m(i;\rho) \\ CB_p(i,:;\rho)^T \end{bmatrix} \tag{3.4}$$

Representing the above equation in the classical linear regressor notation:

$$Z(i;t) = H(t)\Theta(i;\rho) \tag{3.5}$$

where measurement $Z(i;t) = \dot{z}(i;t)$, the regressor matrix $H(t) = \begin{bmatrix} x^T(t) & r^T(t-\tau) & 1 & \Delta u^T(t) \end{bmatrix}$, and the unknown parameters $\Theta(i;\rho) = [CA_m(i,:;\rho)\ CB_m(i,:;\rho)\ CF_m(i;\rho)\ CB_p(i,:;\rho)]^T$. Again, since each row of the output error performance vector is an independent OE model structure, m parallel identification procedures can be performed:

$$\begin{aligned} Z(1;t) &= H(t)\Theta(1;\rho) \\ Z(2;t) &= H(t)\Theta(2;\rho) \\ &\vdots \\ Z(m;t) &= H(t)\Theta(m;\rho) \end{aligned} \tag{3.6}$$

Based on (3.1), an equivalent objective function can be stated for the model structure presented above:

$$\min_{\hat{\Theta}(i;\rho)} J[\hat{\Theta}(i;\rho)] = \lim_{T\to\infty} \frac{1}{T} \sum_{t\geq 0}^{T} \left[Z(i;t) - \hat{Z}[i;t|\hat{\Theta}(i;\rho)] \right]^2 \tag{3.7}$$

where,

$$\hat{Z}[i;t|\hat{\Theta}(i;\rho)] = H(t)\hat{\Theta}(i;\rho)$$

Minimizing the mean-squared error of the above model structure would achieve the identification objective; however, this model structure is not practical for two reasons. First, recall Assumption 10 which states that the reference model must be stable, i.e. A_m is Hurwitz. It is impossible to guarantee that A_m is stable given the identified subset CA_m. Secondly, to obtain

an unbiased estimate of CB_p, there must be adequate Δu excitation. Given that saturation occurs infrequently during normal maneuvers, intentional saturation would have to be injected to obtain good estimates. This is clearly undesirable. Thus, an alternative model structure to (3.4) is needed to obtain good estimated while satisfying required assumptions.

A two stage identification process is proposed that is well suited to on-line identification of the desired dynamics while satisfying the requirement that A_m be Hurwitz. The first stage will involve the identification of the aircraft's control effectiveness matrix CB_p. The second stage will involve the identification of the closed loop desired dynamics A_m and B_m. Note, the complete A_m and B_m versus CA_m and CB_m will be estimated; thus, an additional assumption about the data set is required:

Assumption 15 *The full state derivative is estimated or measured.*

With a complete estimate of A_m, the stability requirement can be satisfied. Finally, based on previous experience, f_m will be derived using the estimates A_m and B_m so that a feasible trim solution of wings level, zero sideslip angle, and zero state derivatives can be enforced. The discussion of the enforced stabilization and trim solution will be presented Section 3.4.

Since the identification process has been separated into two stages, two new model structures are required. Because we desire to maintain a linear regression in the identification parameters, the output error formulation will be retained for both. For the first stage, the aircraft's open-loop nonlinear equations of motion will be used to identify the control effectiveness matrix CB_p. In the second stage, a perturbation state space model of the aircraft's closed-loop dynamics will be used to identify the desired dynamics A_m and B_m. These model structures will be discussed in the following subsections.

3.1.1 Open-Loop Model Structure: Estimation of CB_p

As pointed out above, one of the shortcomings of the model structure in (3.4) is that there is no way to guarantee that Δu will have adequate excitation during normal operation to generate

good estimates of CB_p. An alternative model formulation using the aircraft's open-loop dynamics in the stability axis system can be used to estimate the control effectiveness matrix CB_p. The primary difference between this formulation and the formulation presented above is that the actual control surface deflection $u(t)$ is used instead of the control surface saturation level $\Delta u(t)$. During normal operation, many of control surfaces will have adequate persistent excitation. For control surfaces that do not, additional excitation may have to be injected. Nevertheless, such excitation is far more acceptable when applied to a subset of u then it would be for Δu.

Given Assumption 1, which states that the performance vector consists of the rotation degrees of freedom, the aircraft's rotational dynamics are need to estimate CB_p. The non-linear, stability axis moment equations of motion with linear aerodynamics in matrix-vector format are as follows:

$$\dot{\omega}_S + \Omega_R \omega_S + J_S^{-1} \Omega_S J_S \omega_S = CA_p(\rho)x + CB_p(\rho)u + Cf_p(\rho) \tag{3.8}$$

where the subscript S stands for stability axis, ρ represents the flight condition dependency, the linear aerodynamics $CA_p(\rho) \in \mathbb{R}^{m \times n}$ and $CB_p(\rho) \in \mathbb{R}^{m \times q}$, the aerodynamic intercept $Cf_p(\rho) \in \mathbb{R}^{m \times 1}$, and

$$\omega_S = \begin{bmatrix} P_S \\ Q_S \\ R_S \end{bmatrix} \qquad \Omega_R = \begin{bmatrix} 0 & 0 & -\dot{\alpha} \\ 0 & 0 & 0 \\ \dot{\alpha} & 0 & 0 \end{bmatrix} \qquad \Omega_S = \begin{bmatrix} 0 & -R_s & Q_s \\ R_s & 0 & -P_s \\ -Q_s & P_s & 0 \end{bmatrix}$$

$$\Gamma_s = J_{x_s} J_{z_s} - J_{xz_s}^2 \quad J_s^{-1} = \frac{1}{\Gamma_s} \begin{bmatrix} J_{z_s} & 0 & J_{xz_s} \\ 0 & \frac{\Gamma_s}{J_{y_s}} & 0 \\ J_{xz_s} & 0 & J_{x_s} \end{bmatrix} \quad J_s = \begin{bmatrix} J_{x_s} & 0 & -J_{xz_s} \\ 0 & J_{y_s} & 0 \\ -J_{xz_s} & 0 & J_{z_s} \end{bmatrix}$$

The derivation of these equations is contained in Appendix A. A few aspects of the above equations warrant additional discussion. First, the rotational rates and accelerations and the inertia matrix will have to be computed in the stability axis framework. Secondly, the matrix CA_p will be sparse since the motion is decoupled and only a subset of the state vector $x(t)$ is important.

Now, this is a well-posed model structure for identification of CB_p. Knowing that the intercept term Cf_p is not of interest, it can be removed from the equation by removing the mean from each of the signals:

$$\begin{aligned}[\dot{\omega}_S + \Omega_R\omega_S + J_S^{-1}\Omega_S J_S\omega_S - E\{\dot{\omega}_S + \Omega_R\omega_S + J_S^{-1}\Omega_S J_S\omega_S\}] \\ = CA_p(\rho)[x - E\{x\}] + CB_p(\rho)[u - E\{u\}]\end{aligned} \tag{3.9}$$

Rearranging the open-loop perturbation equation, the OE model structure is obtained:

$$\begin{aligned}&\begin{pmatrix} \dot{\omega}_S + \Omega_R\omega_S + J_S^{-1}\Omega_S J_S\omega_S - \\ E\{\dot{\omega}_S + \Omega_R\omega_S + J_S^{-1}\Omega_S J_S\omega_S\} \end{pmatrix}(i) \\ &= \begin{bmatrix}(x^T - E\{x^T\}) & (u^T - E\{u^T\})\end{bmatrix} \begin{bmatrix} \Theta_{CA_p}(:, i; \rho) \\ \Theta_{CB_p}(:, i; \rho) \end{bmatrix}\end{aligned} \tag{3.10}$$

where the index i reflects the fact that each row represents an independent regressor, $\Theta_{CA_p}(:, i; \rho) \in \mathbb{R}^{(n' < n)\times 1}$ is an estimate of $CA_p(i, :; \rho)^T$, and $\Theta_{CB_p}(:, i; \rho) \in \mathbb{R}^{(q' < q)\times 1}$ is an estimate of $CB_p(i, :; \rho)^T$. Representing the above equation in the classical linear reqressor notation, we have:

$$Z_{ol}(i; t) = H_{ol}(t)\Theta_{ol}(i; \rho)$$

A probabilistic model structure is obtained by assuming the the regressor noise is zero-mean broadband, stationary noise, which is realistic given the modeling. Thus, the open-loop model structure can be expressed as follows:

$$\begin{aligned} Z_{ol}(1; t) &= H_{ol}(t)\Theta_{ol}(1; \rho) + V_{ol}(1; t) \\ Z_{ol}(2; t) &= H_{ol}(t)\Theta_{ol}(2; \rho) + V_{ol}(2; t) \\ &\vdots \\ Z_{ol}(m; t) &= H_{ol}(t)\Theta_{ol}(m; \rho) + V_{ol}(m; t) \end{aligned} \tag{3.11}$$

where $V_{ol}(i; t) = \mathcal{N}(0, R_{ol}(i))$, and $R_{ol}(i) > 0$ is the open-loop model structure regressor noise covariance matrix.

3.1.2 Closed-Loop Model Structure: Estimation of A_m and B_m

For the second stage of the identification process, the first stage estimates will be treated as known information, and an alternative model structure to (3.4) will be formulated. Recall the equivalent formulation for making the output performance error e_o equal to zero:

$$\dot{z}(t) = CA_m(\rho)x(t) + CB_m(\rho)r(t-\tau) + Cf_m(\rho) + CB_p(\rho)\Delta u(t)$$

Since the control effectiveness matrix has be estimated in the first stage, the $CB_p(\rho)\Delta u$ vector can be move to the left side of the expression:

$$\dot{z}(t) - CB_p(\rho)\Delta u(t) = CA_m(\rho)x(t) + CB_m(\rho)r(t-\tau) + Cf_m(\rho) \tag{3.12}$$

Knowing that the entire desired dynamics are needed to satisfy the stability assumption, but only a subset of the control effectiveness matrix has been identified in the first stage, the formulation can be represented as follows:

$$\begin{aligned} \dot{z}(i;t) - CB_p(i,:;\rho)\Delta u(t) &= A_m(i,:;\rho)x(t) + B_m(i,:;\rho)r(t-\tau) + f_m(i;\rho) \\ \dot{x}(j;t) &= A_m(j,:;\rho)x(t) + B_m(j,:;\rho)r(t-\tau) + f_m(j;\rho) \end{aligned}$$

where the index i, which ranges from $1 : m$, represents the subset of states that are in the performance vector and the index j, which ranges from $m = 1 : n$, represents the subset of states that are not contained in performance vector. Again, these indices reflect the fact that each row represents an independent regressor, and the colon index : represents all columns of the respective matrix and the parameter ρ represents the flight condition dependency.

Now, this is a well-posed model structure, but based on previous experience, the author has chosen to remove the intercept — desired trim f_m — from the model structure and estimate it by enforcing a feasible trim solution. Previous experience has shown that small estimation errors in f_m can produce undesirable performance over long time scales. The intercept can be removed from the model structure by simply removing the mean from the data set used in the

structure:

$$\begin{aligned}\dot{z}(i;t) - CB_p(i,:;\rho)\Delta u(t) - E\{\dot{z}(i;t) - CB_p(i,:;\rho)\Delta u(t)\} &= A_m(i,:;\rho)[x(t) - E\{x(t)\}] \\ &+B_m(i,:;\rho)[r(t-\tau) - E\{r(t-\tau)\}] \\ \dot{x}(j;t) - E\{\dot{x}(j;t)\} &= A_m(j,:;\rho)[x(t) - E\{x(t)\}] \\ &+B_m(j,:;\rho)[r(t-\tau) - E\{r(t-\tau)\}]\end{aligned}$$

Rearranging the above equation, the OE closed-loop perturbation model structure is obtained:

$$\begin{gathered}\begin{pmatrix} \dot{z}(i;t) - CB_p(i,:;\rho)\Delta u(t) - \\ E\{\dot{z}(i;t) - CB_p\Delta(i,:;\rho)u(t)\} \end{pmatrix} = \\ \begin{bmatrix} \begin{pmatrix} x^T(t) - \\ E\{x^T(t)\} \end{pmatrix} & \begin{pmatrix} r^T(t-\tau) - \\ E\{r^T(t-\tau)\} \end{pmatrix} \end{bmatrix} \begin{bmatrix} \Theta_{A_m}(:,i;\rho) \\ \Theta_{B_m}(:,i;\rho) \end{bmatrix} \\ \begin{pmatrix} \dot{x}(j;t) - \\ E\{\dot{x}(j;t)\} \end{pmatrix} = \\ \begin{bmatrix} \begin{pmatrix} x^T(t) - \\ E\{x^T(t)\} \end{pmatrix} & \begin{pmatrix} r^T(t-\tau) - \\ E\{r^T(t-\tau)\} \end{pmatrix} \end{bmatrix} \begin{bmatrix} \Theta_{A_m}(:,j;\rho) \\ \Theta_{B_m}(:,j;\rho) \end{bmatrix}\end{gathered} \tag{3.13}$$

where $\Theta_{A_m}(:,i \text{ or} j;\rho) \in \mathbb{R}^{n\times 1}$ is an estimate of $A_m(i \text{ or} j,:;\rho)^T$ and $\Theta_{B_m}(:,i \text{ or} j;\rho) \in \mathbb{R}^{m\times 1}$ is an estimate of $B_m(i \text{ or} j,:;\rho)^T$. Representing the above equation in the classical linear regressor notation:

$$Z_{cl}(i;t) = H_{cl}(t)\Theta_{cl}(i;\rho)$$

The selection of the stability axis system results in longitudinal and lateral-directional decoupling; thus, the desired dynamics A_m and B_m become block diagonal. As a result, the above structure can be separated into longitudinal and lateral-directional structures with the obvious reductions in the regressor matrix dimension and the number of unknown parameters.

The final step required for the closed-loop model structure is a characterization of the regressor noise. The regressor noise is composed of measurement and process noise in the data set and of the modeling errors associated by the approximation of the closed-loop dynamics by a LOES.

Assumption 16 *The closed-loop model structure regressor noise is broadband, stationary noise with zero mean and variance , e.g.* $\mathcal{N}(0, R_{cl})$ *where* $R_{cl} > 0$.

For civil aviation aircraft, with typical stability augmentation systems and under most maneuvering situations, the closed-loop dynamics are accurately modeled by linear time-invariant systems with flight-condition dependent coefficients. Thus, given that the closed-loop model structure is based on a linear time-invariant formulation, the assumption that the modeling error is zero-mean broadband, stationary noise is a reasonable assumption. Furthermore, the assumption that the process and sensor noise is zero-mean broadband noise is not out of the ordinary.

Finally, given (3.13) and Assumption 16, the classical probabilistic linear regressor of the complete closed-loop model structure can be expressed as follows:

$$\begin{aligned} Z_{cl}(1;t) &= H_{cl}(t)\Theta_{cl}(1;\rho) + V_{cl}(1;t) \\ Z_{cl}(2;t) &= H_{cl}(t)\Theta_{cl}(2;\rho) + V_{cl}(2;t) \\ &\vdots \\ Z_{cl}(n;t) &= H_{cl}(t)\Theta_{cl}(n;\rho) + V_{cl}(n;t) \end{aligned} \tag{3.14}$$

where $V_{cl}(i) = \mathcal{N}(0, R_{cl}(i)), R_{cl}(i) > 0$ is the closed-loop model structure regressor noise covariance matrix.

3.2 Batch Least-Squares Identification

As seen in the previous sections, the OE model structures result in a classical linear regression problem. Given this formulation, there are several algorithms available to obtain an estimate of the desired dynamics. The methodology selected is the minimum variance parameter estimation technique which is well documented in [18, 17]. Given the methodology, there are several different algorithms available to arrive at an unbiased, minimum variance estimate. These algorithms can be either batch or sequential. In the batch algorithms, a large data set is post-processed to generate the estimate. The sequential or on-line algorithm generates an estimate recursively

as data becomes available. This section will present the batch identification algorithm and the sequential algorithm will follow in the next section.

It is worthwhile to note that batch identification is generally not well suited for on-line applications. However, batch identification provides a valuable design tool for generating initial model estimates and for post-processing of reconfiguration results.

The generic OE model structure for a given measurement is as follows:

$$Z(t) = H(t)\Theta + V(t) \tag{3.15}$$

where the known measurement $Z(t) \in \mathbb{R}^1$, the known linear dependency $H(t) \in \mathbb{R}^{1\times j}$, the unknown parameter $\Theta \in \mathbb{R}^{j\times 1}$, and the measurement noise $V(t) \in \mathbb{R}^1, \mathcal{N}(0, R)$, is a broadband stationary random process with correlation time much shorter than the sampling interval of the discrete time identification system, with $R > 0$. Hence, the noise at each sample time should be viewed as a white sequence. The dimension j is generic.

Given that the identification data set record has $(L+1)$ measurements corresponding to unique times, the classical linear regressor can be formulated by concatenating each measurement equation into a vector-matrix formulation:

$$\mathbf{Z} = \mathbf{H}\Theta + \mathbf{V} \tag{3.16}$$

where the known measurement $\mathbf{Z} \in \mathbb{R}^{(L+1)\times 1}$, the known regressor matrix $\mathbf{H} \in \mathbb{R}^{(L+1)\times j}$, the unknown parameter $\Theta \in \mathbb{R}^{j\times 1}$, and the regressor noise $\mathbf{V} \in \mathbb{R}^{(L+1)\times 1}, \mathcal{N}(0, \mathbf{R})$, and $\mathbf{R} \in \mathbb{R}^{(L+1)\times(L+1)} > 0$.

Define the measurement estimate, which is a function of the parameter estimate:

$$\hat{\mathbf{Z}}(\hat{\Theta}) = \mathbf{H}\hat{\Theta} \ ,$$

and the parameter and measurement errors

$$\left.\begin{array}{l}\tilde{\Theta} = \Theta - \hat{\Theta} \\ \tilde{\mathbf{Z}} = \mathbf{Z} - \hat{\mathbf{Z}}(\hat{\Theta})\end{array}\right| \quad \Rightarrow \quad \tilde{\mathbf{Z}}(\hat{\Theta}) = \mathbf{H}\tilde{\Theta} + \mathbf{V} = \mathbf{Z} - \mathbf{H}\hat{\Theta} \; . \tag{3.17}$$

Given this regressor, the weight least-squares optimization problem is formulated as:

$$\begin{array}{rcl} \min_{\hat{\Theta}} J(\hat{\Theta}) & = & \left[\mathbf{Z} - \hat{\mathbf{Z}}(\hat{\Theta})\right]^T W \left[\mathbf{Z} - \hat{\mathbf{Z}}(\hat{\Theta})\right] \\ & = & \tilde{\mathbf{Z}}(\hat{\Theta})^T W \tilde{\mathbf{Z}}(\hat{\Theta}) \end{array} \tag{3.18}$$

where the weighting matrix $W \in \mathbb{R}^{(L+1)\times(L+1)} > 0$. The weighting matrix permits the designer to weight certain ranges of the data more heavily than other data.

The batch, unbiased, weighted least-squares estimate is

$$\begin{array}{rcl} \Theta_{ls} & = & \arg\min_{\hat{\Theta}} J(\hat{\Theta}) \\ & = & \left[\mathbf{H}^T W \mathbf{H}\right]^{-1} \mathbf{H}^T W \mathbf{Z} \; , \end{array} \tag{3.19}$$

and knowing that the regressor noise $\mathbf{V}$ has zero mean and that the regressor matrix $\mathbf{H}$ is deterministic, the estimation covariance is

$$P_{ls} = E\{\tilde{\Theta}_{ls}\tilde{\Theta}_{ls}^T\} = \left[\mathbf{H}^T W \mathbf{H}\right]^{-1} \mathbf{H}^T W \mathbf{R} W \mathbf{H} \left[\mathbf{H}^T W \mathbf{H}\right]^{-1} . \tag{3.20}$$

The weighted least-squares algorithm above is a 'valid' estimator, i.e., the estimate does not depend explicitly on the parameters that we are trying to estimate. However, the weight least-squares estimator is not a minimum-variance estimator for any generic weighting matrix. As shown in [18], a minimum-variance estimator is obtained when the weighting matrix $W = \mathbf{R}^{-1}$. This weighting matrix stresses measurements will small covariances and unstresses measurements with large covariances. The resulting batch minimum variance unbiased estimator is :

$$\Theta_{mv} = \left[\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}\right]^{-1} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{Z} \; , \tag{3.21}$$

and the estimation covariance is

$$P_{mv} = E\{\tilde{\Theta}_{mv}\tilde{\Theta}_{mv}^T\} = \left[\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right]^{-1}. \tag{3.22}$$

An important property of this formulation is that because multiple measurements are obtained by the same sensors at each sample time (versus by different sensors at a single time), the measurement covariance at each sample time can be assumed to be the same. Therefore, the regressor covariance matrix $\mathbf{R}$ is an identity matrix scaled by the measurement covariance R

$$\mathbf{R} = R\mathbf{I} \tag{3.23}$$

As a result, the minimum variance estimate is independent of the measurement noise covariance; hence

$$\Theta_{mv} = \left[\mathbf{H}^T\mathbf{H}\right]^{-1}\mathbf{H}^T\mathbf{Z}\ , \tag{3.24}$$

and the estimation covariance is

$$P_{mv} = E\{\tilde{\Theta}_{mv}\tilde{\Theta}_{mv}^T\} = R\left[\mathbf{H}^T\mathbf{H}\right]^{-1}. \tag{3.25}$$

As illustrated in [18], the measurement noise covariance can be estimated using the measurement error, $\tilde{\mathbf{Z}}$ (3.17), by the following unbiased estimator:

$$R = \frac{\tilde{\mathbf{Z}}^T\tilde{\mathbf{Z}}}{\dim[\mathbf{V}] - \dim[\Theta]} \tag{3.26}$$

Thus, the batch, minimum variance unbiased estimation procedure is as follows:

1) Calculate the minimum variance estimate Θ_{mv} using (3.24).

2) Calculate the measurement error $\tilde{\mathbf{Z}}$ using (3.17).

3) Calculate the measurement noise covariance R using (3.26).

4) Calculate the estimation covariance $P_{mv} = E\{\tilde{\Theta}_{mv}\tilde{\Theta}_{mv}^T\}$ using (3.25).

3.3 Modified Sequential Least-Squares Identification

In the previous section, the batch, minimum variance estimation algorithm was presented. As noted, this algorithm is ideally suited for off-line analysis where a long data record is obtained such that all signals of interest have adequate excitation during some portion of the time history. It is well known that the primary source of biased estimates is a lack of persistent excitation in the measurement vector $\mathbf{Z}$ or in the regressor matrix $\mathbf{H}$ [17, 18]; additionally, if the regressor matrix has collinear columns, a biased estimate will result. For on-line analysis, the data record generally needs to be small so that the matrix inverse in (3.24,3.25) does not become a computational burden; there is no way to guarantee that all relevant signals have adequate excitation within a short regressor window. As a result, the batch approach is not well suited to on-line estimation.

To circumvent these deficiencies, the Modified Sequential Least-Square (MSLS) algorithm has been devised. This algorithm, originally developed at the Air Force Research Lab and refined by Barron Associates, has been successfully applied to several simulation flight vehicles and was flight tested on the F-16 VISTA. The foundation for this technique is the minimum variance estimator and is composed of both batch and sequential algorithms. This algorithm is the main part of the RCM architecture that takes a mature military technology and transitions it to commercial applications.

The underlying philosophy of MSLS algorithm is to use current flight data to update *a priori* data on-line such that an unbiased, minimum variance estimate is obtained. The *a priori* data is defined as the data that is stored in the reference model lookup tables. Thus, some form of *a priori* data must exist. Define

$$\Theta_{ap} \text{ and } P_{ap} \tag{3.27}$$

where the initial estimation $\Theta_{ap} \in \mathbb{R}^j$ and the estimation covariance $P_{ap} = E\{\tilde{\Theta}_{ap}\tilde{\Theta}_{ap}^T\} \in \mathbb{R}^{j\times j} > 0$. The *a priori* data can be generated by previous identification runs or through information obtained from wind tunnel for flight tests. The covariance matrix conveys the uncertainty of the *a priori* data to the identification algorithm.

The generic output error model structure for a given measurement is as follows:

$$Z(t) = H(t)\Theta + V(t) \tag{3.28}$$

where the known measurement $Z(t) \in \mathbb{R}^1$, the known linear dependency $H(t) \in \mathbb{R}^{1\times j}$, the unknown parameter $\Theta \in \mathbb{R}^{j\times 1}$, and the measurement noise $V(t) \in \mathbb{R}^1, \mathcal{N}(0, R)$, and $R > 0$. The dimension j is again generic.

As in the batch algorithm, a finite data window with $(L+1)$ measurements corresponding to unique times is used. The classical linear regressor can be formulated by concatenating each measurement equation into a vector-matrix formulation:

$$\mathbf{Z} = \mathbf{H}\Theta + \mathbf{V} \tag{3.29}$$

where the known measurement vector $\mathbf{Z} \in \mathbb{R}^{(L+1)\times 1}$, the known regressor matrix $\mathbf{H} \in \mathbb{R}^{(L+1)\times j}$, the unknown parameter vector has the same definition as above, i.e. $\Theta \in \mathbb{R}^{j\times 1}$, and the regressor noise vector $\mathbf{V} \in \mathbb{R}^{(L+1)\times 1}, \mathcal{N}(0, \mathbf{R})$, and $\mathbf{R} \in \mathbb{R}^{(L+1)\times(L+1)} > 0$. However, unlike the batch algorithm, the data window will change at each sample time. At each sample time, the new data will replace the last row of the regressor, and the subsequent data will be shifted up by a row, which eliminates the oldest data. Thus, the regressor should be view as a data window moving through a time history.

For a given data window, it is possible that the regressor signals do not have adequate excitation to generate an unbiased estimate. As a result, RMS threshold values are used to shutoff the identification when the RMS of the measurement vector falls below the quiescent RMS, i.e. the RMS of the sensor and process noise. Previous experience [24] suggests that the threshold should be double the quiescent RMS. Knowing that the measurement vector is zero mean, based on the model structures above, the measurement RMS is calculated as follows:

$$\text{RMS}_{\mathbf{Z}} = \left(\frac{\mathbf{Z}^T\mathbf{Z}}{L+1}\right)^{1/2} \tag{3.30}$$

If there is insufficient excitation in some signals, the regressor is reparameterized and the identification task is separated into two steps:

1) Batch minimum variance estimation of the reparameterized regressor, and

2) Sequential updating of *a priori* data using the reparameterized results.

The regressor reparameterization is a regularization technique that robustifies the algorithm against potential sources of bias, which include bad regressor column scaling, regressor collinearity, and inadequate column excitation.

The reparameterized regressor will be formulated by first scaling each column of the regressor matrix by its Frobenius norm. Define the scaled regressor as follows:

$$\mathbf{H}_{scaled} = \mathbf{H}\Lambda^{-1} \Rightarrow \mathbf{H}_{scaled}\Lambda = \mathbf{H} \tag{3.31}$$

where $\mathbf{H}_{scaled} \in \mathbb{R}^{(L+1)\times j}$ and the scaling matrix $\Lambda \in \mathbb{R}^{j\times j}$ is a diagonal matrix with each diagonal element

$$\Lambda(i,i) = \begin{cases} 1 & \text{if } \|\mathbf{H}(:,i)\|_F = 0 \\ \|\mathbf{H}(:,i)\|_F & \text{otherwise} \end{cases}$$

Now, singular value decomposition will be performed on the scaled regressor $\mathbf{H}_{scaled}$ to eliminate a potentially large condition number. If the regressor has a high condition number, which implies that either some columns are nearly collinear or that a column lacks adequate excitation, the parameter estimates will be biased. Recall from the model structure discussion that the mean is removed from all signals; thus, inadequate excitation will result in a column of near zeros in the regressor matrix.

Performing singular value decomposition on the scaled regressor:

$$\mathbf{H}_{scaled} = U_{\mathbf{H}_{scaled}}\Sigma_{\mathbf{H}_{scaled}}V^T_{\mathbf{H}_{scaled}} \tag{3.32}$$

where the unitary matrices $U_{\mathbf{H}_{scaled}} \in \mathbb{R}^{(L+1)\times j}$ and $V_{\mathbf{H}_{scaled}} \in \mathbb{R}^{j\times j}$, and the diagonal singular values matrix $\Sigma_{\mathbf{H}_{scaled}} \in \mathbb{R}^{j\times j}$. Note, the subscript has been included so as not to confuse notation.

If $\mathbf{H}_{scaled}$ is not full rank, $\Sigma_{\mathbf{H}_{scaled}}$ will not be full rank. Knowing that a high condition number leads to biased estimates, removing the contribution of small singular values from the regressor will reduce the condition number while not significantly affecting the induced gain of the regressor. Engineering judgment is required to decide what constitutes small, but previous experience [24] suggests that a condition number below 250 is adequate to generate unbiased estimates. Of course, a different ratio may be preferred depending on the application. Removing the small singular values and the accompanying columns of the unitary matrices, the regressor matrix can be expressed as follows:

$$\mathbf{H}_{scaled} \approx \mathcal{U}_{\mathbf{H}_{scaled}} \mathcal{S}_{\mathbf{H}_{scaled}} \mathcal{V}^T_{\mathbf{H}_{scaled}} \tag{3.33}$$

where the unitary matrices $\mathcal{U}_{\mathbf{H}_{scaled}} \in \mathbb{R}^{(L+1)\times k} = U_{\mathbf{H}_{scaled}}(:, 1:k)$ and $\mathcal{V}_{\mathbf{H}_{scaled}} \in \mathbb{R}^{j\times k} = V_{\mathbf{H}_{scaled}}(:, 1:k)$, the diagonal singular values matrix $\mathcal{S}_{\mathbf{H}_{scaled}} \in \mathbb{R}^{k\times k} = \Sigma_{\mathbf{H}_{scaled}}(1:k, 1:k)$, and $k < j$ is the number of singular values kept.

With the scaling and the singular value decomposition, the regressor matrix is approximated as follows:

$$\mathbf{H} \approx \mathcal{U}_{\mathbf{H}_{scaled}} \mathcal{S}_{\mathbf{H}_{scaled}} \mathcal{V}^T_{\mathbf{H}_{scaled}} \Lambda \tag{3.34}$$

Define the following reparameterized variables:

$$\begin{aligned} \mathbf{H}_{rp} &= \mathcal{U}_{\mathbf{H}_{scaled}} \\ \mathbf{K}_{rp} &= \mathcal{S}_{\mathbf{H}_{scaled}} \mathcal{V}^T_{\mathbf{H}_{scaled}} \Lambda \\ \Theta_{rp} &= \mathbf{K}_{rp} \Theta \end{aligned} \tag{3.35}$$

where reparameterized regressor matrix $\mathbf{H}_{rp} \in \mathbb{R}^{(L+1)\times K}$, the reparameterization relationship matrix $\mathbf{K}_{rp} \in \mathbb{R}^{k\times j}$, reduced parameter set $\Theta_{rp} \in \mathbb{R}^k$, and the subscript rp stands for reparameterization. Thus, the regressor in (3.29) can be reparameterized as follows:

$$\mathbf{Z} = \mathbf{H}_{rp} \Theta_{rp} + \mathbf{V} \tag{3.36}$$

where the known measurement $\mathbf{Z} \in \mathbb{R}^{(L+1)\times 1}$, the known reparameterized regressor matrix $\mathbf{H}_{rp} \in \mathbb{R}^{(L+1)\times k}$, the unknown reduced parameter set $\Theta \in \mathbb{R}^{k\times 1}$, and the regressor noise $\mathbf{V} \in \mathbb{R}^{(L+1)\times 1}, \mathcal{N}(0, \mathbf{R})$, and $\mathbf{R} \in \mathbb{R}^{(L+1)\times(L+1)} = R\mathbf{I} > 0$ — recall Equation (3.23)— where $\mathbf{I}$ is the

identity matrix.

As presented in the previous section, the batch minimum variance unbiased estimate of the reduced parameter set is :

$$\Theta_{rp_{mv}} = \left[\mathbf{H}_{rp}^T \mathbf{H}_{rp}\right]^{-1} \mathbf{H}_{rp}^T \mathbf{Z} \ , \tag{3.37}$$

and the estimation covariance is

$$P_{rp_{mv}} = E\{\tilde{\Theta}_{rp_{mv}} \tilde{\Theta}_{rp_{mv}}^T\} = R \left[\mathbf{H}_{rp}^T \mathbf{H}_{rp}\right]^{-1} \tag{3.38}$$

where again, R can be estimated using the return difference $\tilde{\mathbf{Z}}_{rp}$:

$$R = \frac{\tilde{\mathbf{Z}}_{rp}^T \tilde{\mathbf{Z}}_{rp}}{\dim[\mathbf{V}] - \dim[\Theta_{rp}]} \ , \text{ where } \tilde{\mathbf{Z}}_{rp} = \mathbf{Z} - \mathbf{H}_{rp} \Theta_{rp_{mv}} \ . \tag{3.39}$$

At this point, we have an estimate of the reduced parameter along with a linear relationship with the desired parameter in (3.35):

$$\Theta_{rp_{mv}} = \mathbf{K}_{rp} \Theta + \mathbf{V}_{rp} \tag{3.40}$$

where the regressor noise $\mathbf{V}_{rp} \in \mathbb{R}^{k \times k}, \mathcal{N}(0, P_{rp_{mv}})$. To solve for Θ, we could treat this equation as a linear regression; however, there are more unknowns than measurements. Instead, we will treat this equation as a stochastic constraint on the *a priori* data. Thus, *a priori* data will represent an initial guess that is updated by the stochastic constraint in a sequential formation. The above equation should be viewed as an additional measurement.

Using the *a priori* data in (3.27) and the additional measurement equation in (3.40), the minimum variance sequential estimate of Θ can be computed as follows:

$$\Theta_{mv} = \Theta_{ap} + P_{ap} \mathbf{K}_{rp}^T \left[\mathbf{K}_{rp} P_{ap} \mathbf{K}_{rp}^T + P_{rp_{mv}}\right]^{-1} \left(\Theta_{rp_{mv}} - \mathbf{K}_{rp} \Theta_{ap}\right) \ , \tag{3.41}$$

and the estimation covariance is

$$P_{mv} = E\{\tilde{\Theta}_{mv}\tilde{\Theta}_{mv}^T\} = \left[\mathrm{I} - P_{ap}\mathbf{K}_{rp}^T\left(\mathbf{K}_{rp}P_{ap}\mathbf{K}_{rp}^T + P_{rp_{mv}}\right)^{-1}\mathbf{K}_{rp}\right]P_{ap} \tag{3.42}$$

where $\Theta_{mv} \in \mathbb{R}^j$ and $P_{mv} \in \mathbb{R}^{j\times j}$. Note:

1) If $k < j$, i.e. $\Theta \in \mathbb{R}^j$ and $\Theta_{rp} \in \mathbb{R}^k$, then $j-k$ parameters will be dominated by *a priori* data;

2) If the uncertainty in the reparameterized estimate is large, i.e. $P_{rp_{mv}} \gg 0$, then the parameter estimate Θ_{mv} will be dominated by *a priori* data;

3) Likewise, if the uncertainty in the reparameterized estimate is small, i.e. $P_{rp_{mv}} \approx 0$, then the parameter estimate Θ_{mv} will be dominated by the reparameterized estimate.

3.4 Stabilization of Closed-Loop Model and Estimation of f_m

In this section, the stabilization of the closed-loop model and the estimation of the desired trim state will be detailed. In the previous sections, the two stage identification process was discussed for the identification of the desired dynamics A_m, B_m and the aircraft's control effectiveness matrix CB_p. A model structure of the aircraft's open-loop dynamics was derived for the estimation of CB_p, and a model structure of the aircrafts closed-loop dynamics has been derived for the estimation of Am and Bm. Furthermore, two minimum variance unbiased alogrithms — batch and sequential — have been presented to generate the estimates. All that remains for the required reference model is to insure stability of the desired dynamics A_m and to estimate the desired trim f_m. Recall from the previous discussion that, in order to have a feasible reference model, the desired dynamics A_m must be strictly stable. Furthermore, as pointed out in the closed-loop model structure development, Section 3.1.2, previous experience suggests that it is better to derive the desired trim state instead of adding the intercept as an unknown parameter to the model structure.

To ensure that the closed-loop reference model satisfies the assumption of strict stability, the Phugoid and spiral modes are artificially stabilized. Even though these modes must be stable for Level I handling qualities, they are generally difficult to identify since they have very large

time constants and are attenuated by the nominal controllers. This stabilization operation is straight forward. The Phugoid mode is artificially stabilized by mirroring the respective poles about the $j\omega$ axis. Eigenanalysis is performed on the longitudinal block of A_m, and eigenvectors and eigenvalues are extracted. If the Phugoid eigenvalues are unstable, the poles are mirrored about the $j\omega$ axis to stabilize the mode. Then, a new longitudinal block in A_m is constructed using the same eigenvectors and the stabilized eigenvalues.

Flight mechanics are used to artificially stabilize the spiral mode. The stability requirement for the spiral mode is presented in [71, 74]:

$$L_\beta N_r - N_\beta L_r > 0 \tag{3.43}$$

If this inequality is not satisfied, the spiral mode will be unstable. To artificially stabilize this mode, the dimensional derivatives need to be change without altering the other lateral-directional modes. For traditional tail-aft configurations, L_β and $N_r < 0$ while N_β and $L_r > 0$; thus to satisfy this inequality $L_\beta N_r > 0$ must be greater than $N_\beta L_r > 0$. The only dimensional derivative that can be altered without having a 1st order impact on the Dutch roll mode is the yaw rate induced rolling acceleration L_r. Thus, to artificially stabilize the spiral mode, we let

$$L_r = \frac{L_\beta N_r - \epsilon}{N_\beta} > 0 \tag{3.44}$$

where $\epsilon > 0$ is a small number, which reduces L_r without changing its sign. This stabilization procedure replaces the identified L_r with the computed value above. In fact, the procedure states that if inequality in (3.43) is violated, then the magnitude of L_r has been overestimated and needs to be reduced.

The next topic of this subsection is the estimation of f_m. In the closed-loop model structure section Section 3.1.2, is was noted that it is desirable to derive the desired trim f_m instead of estimating it since small errors can produce undesirable performance over large time scales. For example, a small error in the $\dot{q}(t)$ trim, which should be zero, can produce an undesirable time history over a ten minute span since this error is integrated twice into the pitch angle $\theta(t)$. Also recall from the previous discussion that the intercept was removed from the regressor by

removing the mean from the data set used in the model structure. This mean information along with the estimated desired dynamics A_m and B_m will be used to derive a feasible f_m. Note, a feasible f_m is defined as one which achieves a desired trim ($\dot{x}(t) = 0$ and $p(t), q(t), r(t) = 0$) for wings level and zero sideslip ($\phi, \beta = 0$); thus, only the true velocity $v_t(t)$, the angle of attack $\alpha(t)$, and the pitch angle $\theta(t)$ may be non-zero at trim.

We begin with the LOES desired dynamics:

$$\dot{x}(t) = A_m x(t) + B_m r(t - \tau) + f_m \tag{3.45}$$

Clearly, f_m can be estimated as follows:

$$f_m = \mathrm{E}\{\dot{x}(t)\} - A_m \mathrm{E}\{x(t)\} + B_m \mathrm{E}\{r(t - \tau)\} \tag{3.46}$$

Given this estimate, knowing that $\dot{x}(t)$ and $r(t - \tau)$ are null at trim, an estimate of the non-zero longitudinal states can be obtained by performing a pseudo inverse of a subset of A_m and f_m:

$$\begin{bmatrix} Vt_{trim} \\ \alpha_{trim} \\ \theta_{trim} \end{bmatrix} = \frac{-f_m(i)|_{i=1:4}}{A_m(i,j)|_{i=1:4,\ j=1:3}} \tag{3.47}$$

where the index $1 : 4$ represents the 4 rows corresponding to the longitudinal states and the index $1 : 3$ represents the 3 columns corresponding to the non-zero trim states. Given this pseudo inverse data, the entire trim vector f_m can be constructed as follows:

$$f_m = -A_m x_{m_{trim}} \text{ , where } x_{m_{trim}} = \begin{bmatrix} Vt_{trim} \\ \alpha_{trim} \\ \theta_{trim} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.48}$$

3.5 Storing Reference Model Parameters

As stated in the introduction to this chapter, on-line system identification generates estimates of A_m, B_m, f_m and CB_p throughout the flight envelope, with the goal of achieving a near-zero output performance error $e_o(t)$. Figure 3-1 illustrates a schematic of a flight envelope. Given

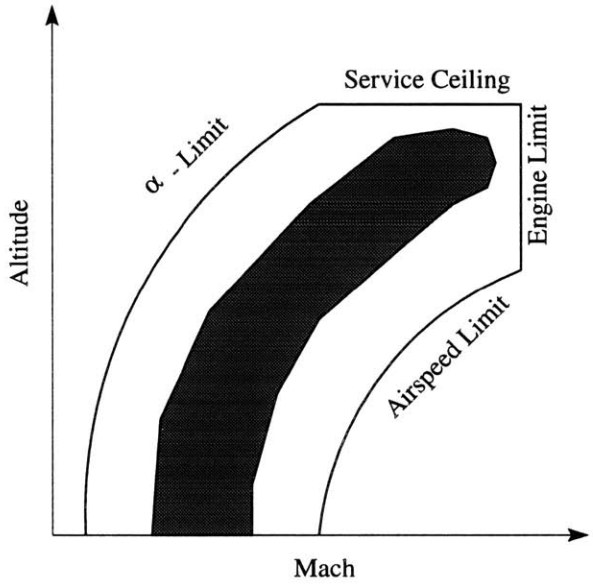

Figure 3-1: Flight Envelope Schematic

that a typical flight profile is altered by destination and air-traffic control, a series of 'flight cycles' will traverse the center region of this envelope. The shaded area in Figure 3-1 is defined as the nominal operation region for this discussion. During a series of flight cycles, the system identification procedure outlined in the previous sections will generate data throughout this region. Thus, an accurate reference model of a healthy aircraft is obtainable in this region. This raises two important questions: 1) how do we obtain data in other regions of the flight envelope, and 2) how do we store this data.

Recall that the key measure for failure detection and reconfiguration is the output performance error, which is computed using the flight condition reference model. Thus, if a failure occurs, the aircraft should be within the shaded area, though it may exit this region due to the failure. Given that a good reference model is obtainable in this region, the failure detection task should

be easy since the nominal output performance error is small. As mentioned, a failed aircraft may exit the shaded region due to a failure. This may be due to the transient nature of the failure, or due to the fact that the post-failure trim drag is dramatically higher. In any event, a reference model must be available outside the shaded region. There are several options for providing this data outside the region of normal operation. First, wind tunnel or simulator/flight test data can be used to construct the reference model data. If this information is not readily available, the aircraft can be flown into these regions intentionally to obtain a reference model. As a last resort, a reference model can be specified using empirical approaches. It is important to note that a highly accurate reference model of the nominal performance is not necessary outside the normal operating region. If a failure forces an aircraft into the region, nominal performance is a luxury, and the primary concern is the controlled return to the normal operating region so that an emergency landing can be conducted. Due to the possibilities of large modeling error outside the normal operational envelope, if a healthy aircraft enters this region for some reason or another, a false alarm might occur. In this event, the pilot would be required to manually disengage the RCM failure switch and allow the system identification module to generate an adequate reference model. Note, in this situation, a false alarm will not cause any sever transients or adverse behavior since the retrofit module naturally exhibits zero or very small outputs when there is no failure.

The second issue associated with the flight-condition dependent reference model is how to store that data. Given that estimates of A_m, B_m, f_m and CB_p are available throughout the flight envelope, each element of these matrices and vectors, which represents a dimensional derivative, needs to be stored:

$$A_{m_{i,j}}|_{i=1:n,\ j=1:n} \quad B_{m_{i,j}}|_{i=1:n,\ j=1:m} \quad f_{m_i}|_{i=1:n} \quad B_{m_{i,j}}|_{i=1:m,\ j=1:q}$$

Recall from elementary flight dynamics that the partial derivatives of aerodynamic forces and moments with respect to state and input variables are termed *dimensional derivates*. Thus, there are potentially $n \times (n + m + 1) + m \times q$ dimensional derivatives that must be stored. As will be demonstrated in subsequent chapters, enforcing the proper aircraft dynamics will significantly reduce the number of dimensional derivatives that require storage. Nevertheless, there is a lot

of information that needs to be stored.

It is recommended that the reference model data be stored in a dimensionless coefficient format versus a dimensional coefficient format. Recall, dimensional derivates can be converted to dimensionless derivatives or coefficients by normalizing the dimensional derivatives using reference lengths and areas, dynamic pressure, and aircraft inertias [71, 74]. There are two good reasons for this approach. First, inertia changes and density variations will be easily accounted for in the reference model. Second, dimensionless coefficients vary smoothly through the flight envelope; this property will be utilized in the following paragraphs. It is well known that for commercial transport and general aviation aircraft the clean configuration dimensionless coefficients are approximately constant throughout the flight envelope. These coefficients will vary due to dynamic pressure and compressibility effects, but given that these aircraft only enter the low transonic regime, variations should be smooth. For unclean configurations, i.e. flaps down, gear down, in-ground effects, increment data would be required.

As illustrated in Figure 3-1, the flight envelope is generally parameterized with two variables. Some common variables used are Mach, altitude, dynamic pressure, total pressure, and static pressure. For this discussion, Mach and altitude will be used as envelope parameters. Each dimensionless coefficient must be stored relative to these two parameters. This can be performed simply using a discrete 2-D table or using polynomials or neural networks. Since the data is being stored as dimensionless coefficients, the fluctuations between identification results should be benign. Given the smooth nature of dimensionless coefficients throughout the envelope, low order polynomials and neural networks can be used to approximate the identification results.

There are two important issues associated with the data storage mechanism: ability to adapt the parameters and storage space requirements. These issues will be addressed for each of the data storage approaches. A discrete table is by far the most common data storage mechanism. Here, the dimensionless coefficients are stored at a discrete number of flight conditions. Updates to these points are performed using a covariance weighted average of identification data obtained in a neighborhood of the discrete point. Although this is a simple approach, the discrete table storage mechanism suffers from large memory requirements. On the other end of the spectrum,

neural networks provide a highly efficient data storage mechanism. Storing stability and control derivatives in neural networks has been shown to be an effective means by Peterson [75]. However, adaptation of the neural networks can be problematic because over-training can result in large fluctuations between training points. An alternative to the neural network approach is to use polynomial functions with a fixed number of coefficients to approximate data. As mentioned above, a low-order polynomial with a fixed number of coefficients can accurately represent the coefficient distribution over the flight envelope. Since there will only be a few coefficients per stability derivative, this provides an effect data storage mechanism. Given that the estimation error covariances can be retained in this approach, the polynomial constants can be updated using a covariance weighting approach (i.e. weighted least-squares).

3.6 Summary

The reference model is the second component of the RCM architecture. The reference model's primary function is to provide desired dynamics CA_m, CB_m, and Cf_m and an estimate of the aircraft's control power effectiveness CB_p to the adaptive augmentor component for the calculation of the output performance error $e_o(t)$. The output performance error has a dual functionality in the RCM architecture. During failure operations, a scaled version of $e_o(t)$, which is the input error $e_i(t)$, is used to update the reconfiguration gains, and during nominal operations, $e_o(t)$ is the key measure for determining whether a failure has occurred. In order to use the output performance error as a measure for failure detection, $e_o(t)$ needs to be near zero during nominal operation. To achieve a near zero output error throughout the flight envelope, the reference model must be flight condition dependent. Given the additional requirements for low-cost implementation and life cycle maintenance, the reference model is to be generated on-line using system identification.

The objective of the on-line system identification is to minimize the mean-squared output per-

formance error throughout the flight envelope:

$$\min_{\hat{\Theta}(\rho)} J[\hat{\Theta}(\rho)] = \lim_{T\to\infty} \frac{1}{T} \sum_{t\geq 0}^{T} e_o[t|\hat{\Theta}(\rho)]^2$$

where the parameters $\hat{\Theta}(\rho)$ are the estimated desired dynamics — $\hat{CA}_m, \hat{CB}_m, \hat{Cf}_m$ and $\hat{CBp}$ — and the parameter ρ represents the flight condition dependency.

To achieve this goal, a two stage identification process is outlined which is well suited for on-line identification of the desired dynamics. The first stage involves the identification of the aircraft's control effectiveness matrix CB_p, and the second stage involves the identification of the closed loop desired dynamics A_m and B_m. Note, the complete A_m and B_m (versus CA_m and CB_m) are estimated; the complete A_m is required to enforce Assumption 10 which states that the reference model must be stable, i.e. A_m is Hurwitz. Finally, f_m is derived using the estimates A_m and B_m so that a feasible trim solution of wings level, zero sideslip angle, and zero state derivatives can be enforced.

Given this reference model development approach, the identification data structure, model structure, and optimization algorithms were presented. Additionally, procedures for enforcing stability of the reference model and estimating f_m were presented. Finally, issues associated with storing the reference model were addressed.

Chapter 2 listed several assumptions pertaining to the data set that is available to the RCM. Given the requirement of enforcing stability, an additional assumption was required, Assumption 15, which states that the full state derivative is estimated or measured. For the first stage model, the aircraft's open-loop nonlinear equation of motion will be used to identify the control effectiveness matrix CB_p and a perturbation state space model of the aircraft's closed-loop dynamics will be used to identify the desired dynamics A_m and B_m.

Given the data set and dynamical models, an Output-Error (OE) identification model structure can be used; thus, the identification problem is translated into the classical linear regression problem. Given this formulation, there are several algorithms available to obtain an estimate of the desired dynamics. The methodology selected is the minimum variance parameter estimation

technique. Given the methodology, there are several different algorithms available to arrive at an unbiased, minimum variance estimate. These algorithms can be either batch or sequential, both of which are presented in this chapter. In the batch algorithms, a large data set is post-processed to generate the estimate. Though batch identification is generally not well suited for on-line applications, it does provide a valuable design tool for generating initial model estimates and for post-processing of reconfiguration results. The sequential or on-line algorithm generates an estimate recursively as data becomes available. The sequential algorithm selected is the Modified Sequential Least-Square (MSLS) algorithm. This algorithm, originally developed at the Air Force Research Lab and refined by Barron Associates, has been successfully applied to several simulation flight vehicles and was flight tested on the F-16 VISTA. The foundation of this technique is based on the minimum variance estimator and is composed of both batch and sequential algorithms. This algorithm is the main part of the RCM architecture that takes a mature military technology and transitions it to commercial applications.

Since the reference model is flight condition dependent, the identification results need to be stored for various flight conditions. This data can simply be stored in two-dimensional tables; however, a two-dimensional polynomial approach is proposed for the data storage. In this arrangement, fixed dimension polynomials are used to approximate the dimensionless coefficient variations through the flight envelope. This approach minimizes the data storage requirements and computational requirements of generating reference model data throughout the flight envelope. Furthermore, slow adaptation of the reference model can be accomplished using sequential, unbiased minimum variance techniques like the ones presented in this chapter.

In summary, this chapter outlines the procedures for generating a reference model that minimizes the RCM implementation and life cycle cost. The accuracy of the reference model is the key issue associated with the RCM operational performance. Inaccuracies in the reference model will lead to false alarms and ultimately limit the achievable post-failure performance. Given these and other functional requirements for the reference model component of the RCM architecture, an on-line system identification process is outlined that satisfies the functional requirements.

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 4

Signal Conditioning and Estimation Module

The signal conditioning and estimation module is the final component of the RCM architecture. Its primary function is to provide the required signals to the adaptive augmentor and reference model components. As outlined in the previous two chapters, the required signals include the stability axis state vector $x(t)$, the stability axis state derivative vector $\dot{x}(t)$, the pilot input commands $r(t-\tau)$, the control surface deflections $u(t)$, and the control surface saturation levels $\Delta u(t)$. Recall that in many cases, the assumptions about the available data were made to simplify the algorithms in the other two modules. Given that some of this data is not available on existing commercial transports, these previous simplifications result in greater complexity for this component.

In this chapter, the generation of the required information via conditioning or estimation will be discussed in the context of realistic sensor suites in existing commercial transports. The contents of this chapter do not represent a contribution to control theory or practice. Instead, this chapter simply presents definitions, a Kalman filter formulation, and a surface model architecture for the purpose of completeness. Thus, it may be skipped at this point and used solely as reference in subsequent chapters.

4.1 State Vector $x(t)$

The stability axes state vector $x(t)$ generally includes the states listed in Table 4.1. With the

Table 4.1: Stability Axes Aircraft States $x(t)$

Variables	Description
$v_t(t)$	True Airspeed
$\alpha(t)$	Angle of Attack
$\theta(t)$	Pitch Attitude
$q_s(t)$	Pitch Rate
$\beta(t)$	Sideslip Angle
$\phi(t)$	Bank Angle
$p_s(t)$	Roll Rate
$r_s(t)$	Yaw Rate

current stability augmentation systems and autopilots, it is reasonable to assume the nominal control system already has either measured, filtered, and or estimated the entire state vector. The only conditioning of the state vector that might be required is the conversion of the angular rates from body-axes to stability-axes coordinate system. If the angular rates are in body-axes system, they can be transformed to the stability-axes system via the following coordinate transformation:

$$\begin{bmatrix} p_s(t) \\ q_s(t) \\ r_s(t) \end{bmatrix} = S_\alpha \begin{bmatrix} p_b(t) \\ q_b(t) \\ r_b(t) \end{bmatrix} \text{where} S_\alpha = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{4.1}$$

where the subscripts s and b stand for stability-axes and body-axes.

4.2 State Derivative Vector $\dot{x}(t)$

The stability axes state derivative vector $\dot{x}(t)$ are listed in Table 4.2. Many of these derivatives will not be accessible from the nominal control system for existing commercial transports; thus, they must be estimated. The state derivative will be estimated using a bank of Kalman filters:

Table 4.2: Stability Axes Aircraft State Derivatives $\dot{x}(t)$

Variables	Description
$\dot{v}_t(t)$	True Airspeed Derivative
$\dot{\alpha}(t)$	Angle of Attack Derivative
$\dot{\theta}(t)$	Pitch Attitude Derivative
$\dot{q}_s(t)$	Pitch Acceleration
$\dot{\beta}(t)$	Sideslip Angle Derivative
$\dot{\phi}(t)$	Bank Angle Derivative
$\dot{p}_s(t)$	Roll Acceleration
$\dot{r}_s(t)$	Yaw Acceleration

each state will have an associated Kalman filter. The dynamical model that we will use for the Kalman filter is illustrated in Figure 4-1. The output of this model is one of the state

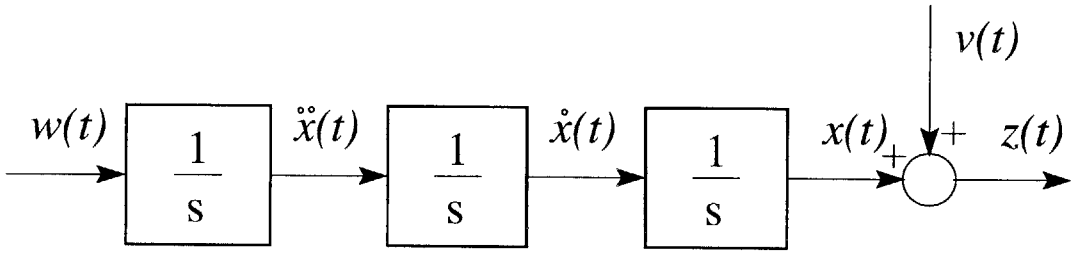

Figure 4-1: Kalman Filter Model for Estimation $\dot{x}(t)$

variables, which is corrupted by measurement noise $v(t)$. The other two states of this model are the state derivative and state acceleration, the later being driven by process noise $w(t)$. For example, if an estimate of the pitch acceleration $\dot{q}(t)$ is sought, then the Kalman filter state vector $x(t) = [q(t)\dot{q}(t)\ddot{q}(t)]^T$. For this application, both the process and sensor noise are assumed to be continuous time, white Gaussian noise with zero mean; thus,

$$\begin{aligned} w(t) \in \mathbb{R}^1, \mathcal{N}(0, Q), \text{ or } E\{w(t)w(\tau)\} = Q\delta(t-\tau), Q > 0 \\ v(t) \in \mathbb{R}^1, \mathcal{N}(0, R), \text{ or } E\{v(t)v(\tau)\} = R\delta(t-\tau), R > 0 \end{aligned} \tag{4.2}$$

Furthermore, it is assumed that the process and sensor noise are uncorrelated. Based on this dynamical model, we define the process model as follows:

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t) \\ \dot{x}(t) &= Fx(t) + Gw(t) \end{aligned} \tag{4.3}$$

Additionally, we define the measurement model as:

$$\begin{aligned} z(t) &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x(t) + v(t) \\ z(t) &= Hx(t) + v(t) \end{aligned} \tag{4.4}$$

Under the stated assumptions on the process and sensor noise, there are several optimality criteria that result in the same 'optimal' estimator, the continuous time Kalman filter. The Kalman filter is well documented [76, 77, 78]. The formulation of the continuous time Kalman filter is summarized in Table 4.3. Additionally, since the process model (4.3) and measurement

Table 4.3: Summary of Continuous Kalman Filter Equations (Copied from [76])

Process Model	$\dot{x}(t) = Fx(t) + Gw(t),\ w(t) \sim \mathcal{N}(0, Q)$
Measurement Model	$z = Hx(t) + v(t),\ v(t) \sim \mathcal{N}(0, R)$
Initial Conditions	$E\{x(0)\} = \hat{x}_o,\ E\{(x(0) - \hat{x}_o)(x(0) - \hat{x}_o)^T\} = P_o$
State Estimate	$\dot{\hat{x}}(t) = Fx(t) + K(t)\left[z(t) - H\hat{x}(t)\right]$
Estimation Error Covariance	$\dot{P}(t) = FP(t) + P(t)F^T + GQG^T - K(t)RK^T(t)$
Kalman Gain	$K(t) = P(t)H^T R^{-1}$

model (4.4) are controllable and observable respectively, the Kalman filter is guaranteed to be stable and the estimation error covariance matrix $P(t)$ is positive definite and unique.

In this application, the only parameter that is available to tune the Kalman filter is the process noise covariance Q, since the measurement noise covariance is a function of the particular sensor. The key measures of performance for the Kalman filter are the estimation error covariance matrix $P(t)$ and time delay τ associated with the estimation. The relationship between the

design parameter Q and the performance parameters $P(t)$ and τ are best illustrated by using the steady-state solution to the Kalman filter. Given that the dynamical model is an LTI system and that the statistics are stationary, the Kalman filter will reach a steady state condition, i.e. $P(t) \rightarrow P_\infty$ which is a constant matrix. The steady-state solution to this particular problem is present in Example 11B of [78]. Only the pertinent information will be duplicated. The steady-state estimation error covariance matrix is as follows:

$$P_\infty = \begin{bmatrix} 2Q^{\frac{1}{6}}R^{\frac{5}{6}} & 2Q^{\frac{1}{3}}R^{\frac{1}{3}} & Q^{\frac{1}{2}}R^{\frac{1}{2}} \\ & 3Q^{\frac{1}{2}}R^{\frac{1}{2}} & 2Q^{\frac{2}{3}}R^{\frac{1}{3}} \\ \text{sym} & & 2Q^{\frac{5}{6}}R^{\frac{1}{6}} \end{bmatrix} \tag{4.5}$$

Additionally, a transfer function from the filter input $z(t)$ to the desired output $\dot{x}(t)$ is as follows:

$$\frac{\dot{X}(s)}{V(s)} = \frac{2\Omega^2 s^2 + \Omega^3 s}{s^3 + 2\Omega s^2 + 2\Omega^2 s + \Omega^3} \tag{4.6}$$

where s is the Laplace variable and $\Omega = (Q/R)^{\frac{1}{6}}$ is the ratio of process to sensor noise covariances. The poles of this transfer function are $s_1 = -\Omega$ and $s_{2,3} = -\Omega(\frac{1}{2} \pm \frac{\sqrt{3}}{2}j)$. Thus, the Kalman filter's bandwidth is proportional to the ratio of process and sensor noise. Using this transfer function, a time delay can be associated with the filter by modeling the changes in phase due to the filter as a transport delay:

$$\tau(\omega) = \frac{\phi(0) - \phi(\omega)}{\omega}, \quad \text{where} \quad \phi(\omega) = \angle\dot{X}(j\omega) - \angle V(j\omega) \tag{4.7}$$

The maximum time delay associated with a given process to sensor noise covariance ratio can be determined by finding the maximum of (4.7).

A few comments about the Kalman filters performance are in order. For a given measurement noise covariance, a large process noise covariance translates into a high filter bandwidth, a small maximum time delay, and a large estimation error covariance. So how should Q be chosen? Since the estimates from the resulting Kalman filter are used in the system identification module, the bandwidth is the most critical.

To tune the filter, the bandwidth requirements must be determined by considering the physics of the signal. For example, if an estimate of the pitch acceleration is required, the control surface rate limit gives an approximate lower limit for the required bandwidth. Given this bandwidth requirement, the assumed process noise covariance can be determined using the undamped natural frequency $\Omega = (Q/R)^{\frac{1}{6}}$. Likewise, an estimate of the maximum time delay and an estimate of the estimation error covariance can be made via (4.7) and (4.5). The time delay information should then be used in the identification process to ensure that all data is transformed to equivalent times.

To implement the Kalman filter in the RCM architecture, a discrete formulation is necessary. A discrete representation of the process model in (4.3) is as follows:

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1 & \Delta T & \frac{1}{2}\Delta T^2 \\ 0 & 1 & \Delta T \\ 0 & 0 & 1 \end{bmatrix} x_k + w_k \\ x_{k+1} &= \Phi x_k + w_k \end{aligned} \tag{4.8}$$

where ΔT is the sampling interval and w_k is the zero mean, discrete time white noise sequence:

$$E\{w_k w_i^t\} = \begin{cases} Q_k & k = i \\ 0 & k \neq i \end{cases}, \quad \text{where} \quad Q_k = \begin{bmatrix} \frac{\Delta T^5}{20} & \frac{\Delta T^4}{8} & \frac{\Delta T^3}{6} \\ & \frac{\Delta T^3}{3} & \frac{\Delta T^2}{2} \\ \text{sym} & & \frac{\Delta T}{1} \end{bmatrix} Q \tag{4.9}$$

Likewise, the discrete representation of the measurement model in (4.4) is as follows:

$$\begin{aligned} z_k &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x_k + v_k \\ z_k &= H x_k + v_k \end{aligned} \tag{4.10}$$

where measurement noise v_k is the zero mean, discrete time white noise sequence:

$$E\{v_k v_i^t\} = \begin{cases} R_k & k = i \\ 0 & k \neq i \end{cases}, \quad \text{where} \quad R_k = \frac{R}{\Delta T} \tag{4.11}$$

The main results of the discrete Kalman filter are well documented [76, 77]. The formulation are summarized in Table 4.4.

Table 4.4: Summary of Discrete Kalman Filter Equations (Copied from [76])

Process Model Measurement Model	$x_k = \Phi x_{k-1} + w_{k-1},\ w_k \sim \mathcal{N}(0, Q_k)$ $z_k = H x_k + v_k,\ v_k \sim \mathcal{N}(0, R_k)$
Initial Conditions	$E\{x(0)\} = \hat{x}_o,\ E\{(x(0) - \hat{x}_o)(x(0) - \hat{x}_o)^T\} = P_o$
State Estimate Extrapolation Error Covariance Extrapolation	$\hat{x}_k(-) = \Phi \hat{x}_{k-1}(+)$ $P_k(-) = \Phi P_{k-1}(+)\Phi^T + Q_{k-1}$
State Update Estimation Error Covariance Kalman Gain	$\hat{x}_k(+) = \hat{x}_k(-) + K_k\left[z_k - H\hat{x}_k(-)\right]$ $P_k(+) = \left[I - K_k H\right] P_k(-)$ $K_k = P_k(-)H^T\left[HP_k(-)H^T + R_k\right]^{-1}$

4.3 Pilot Input $r(t-\tau)$

There are two important issues associated with the pilot reference inputs. The first is the types of inputs (i.e. stick, pedal, throttle, autopilot, etc.), and the second is the delay associated with these inputs; both will be discussed in this section. Given that the reconfiguration performance, Assumption 1, contains the rotational rates of the aircraft, the pilot commands that are of interest include all commands that affect the rotation response of the aircraft. These commands include manual stick and pedal inputs or certain autopilot modes that the pilot may engage. Table 4.5 presents the definitions of pilot inputs that are used in subsequent chapters. Note,

Table 4.5: Pilot Inputs $r(t-\tau)$ Definitions

Variables	Description
$dap(t)$	Roll Stick Input
$dep(t)$	Pitch Stick Input
$drp(t)$	Yaw Pedal Input

throttle commands are not include since the auto-throttle functionality is not critical for control surface reconfiguration. Generally, the units will be a length that represents a deflection from

the free position.

The generation of these signal requires special consideration when the autopilot is engaged. In general, the autopilot functions as an outer loop control system that generates equivalent pilot inputs to perform a specified task, i.e, attitude hold, altitude hold, heading hold, etc. Given that we desire to perform both identification and reconfiguration with and without the autopilot engaged, we need to be able to measure or construct $r(t-\tau)$ with or without the autopilot engaged. For some commercial transports with yoke columns, autopilot commands are converted to equivalent stick inputs to back-drive the stick; thus, they provide cues to the pilots of the autopilots commands. Given this situation, stick and pedal movements are directly measurable and no signal conditioning or estimation is required. For other transport aircraft, especially with side stick devices, autopilot commands do not back-drive the stick. In this situation, the signal conditioning and estimation module must generate the equivalent stick commands as though they were to be used to back-drive the stick. This can be accomplished by taking an autopilot command and converting it to equivalent stick commands by inverting the stick logic.

The next issue associated with the generation of $r(t-\tau)$ is the delay τ. This transport delay is a result of representing a high-order system with a low-order equivalent system as is done in Section 2.2. The delay is used in the low-order model for matching high-frequency effects of the higher order model [71]. These high-frequency effects include actuator dynamics, control systems filters, and possible structural dynamics. Though the time delay is generally frequency dependent, a constant τ is assumed. This delay needs to be determined prior to installation of the retrofit module, and is to remain constant thereafter. Given that pilot inputs (Table 4.5) generate accelerations about the three axes, an estimate of the time delay can be obtained by injecting a step input into each axis and then measuring the time delay associate with the rotation accelerations.

4.4 Control Surface Deflections $u(t)$

The next set of signals that are required are the control surface deflections $u(t)$, which are used solely in the first stage identification of CB_p. Differing from the other required signals, the control surface signals are aircraft dependent (for obvious reasons). Thus, all that can be done here is to set up standard definitions for deflections and discuss the measurement or estimation of the signals.

Aircraft generally have redundant control surfaces, e.g. spoilers and ailerons for roll, split rudders for yaw, split elevators or stabilizers for pitch. Depending how these surfaces are deflected, they can produce either longitudinal or lateral-directional moments or both. For example, symmetrically deflected horizontal stabilators — both surfaces deflecting the same amount in the same direction — produce pure pitching moment. Differentially deflected horizontal stabilators — both surfaces deflecting the same amount in opposite directions — produce a rolling and yawing moment. If only one surface is deflected, pitching, rolling and yawing moments will result.

In an attempt to reduce the dimensions of the control space and to decouple control surface deflections, equivalent surface deflections are defined as average deflections for each particular control surface type. The term *symmetric* deflection will signify a set of control surface deflections that produce primarily pitching moment, and *differential* deflections will signify deflections that produce primarily lateral-directional moments. Table 4.6 contains definitions of equivalent control surface deflections that are used in subsequent chapters. In this table, the subscripts l and r signify left and right surfaces, the summation accounts for multiply left and right surfaces, and k is an averaging constant. The selection of the averaging constant is up to the designer

Table 4.6: Equivalent Control Surface Deflections $u(t)$

Variables	Description
u_{sym}	$\sum_i (u_{l_i} + u_{r_i})/k$
u_{dif}	$\sum_i (u_{l_i} - u_{r_i})/k$

and generally reflects the number and type of control surface.

With the equivalent surface definitions defined, we proceed to the discussion of measuring or estimating the control surface deflections. It is not uncommon that a control system provides direct measurement of the control surface positions; thus, the only signal conditioning required is the conversion of individual surfaces into the equivalent deflections defined above. For other aircraft, the control surface deflections need to be estimated. The control surface deflections can be accurately estimated using a surface model like the one illustrated in Figure 4-2. The

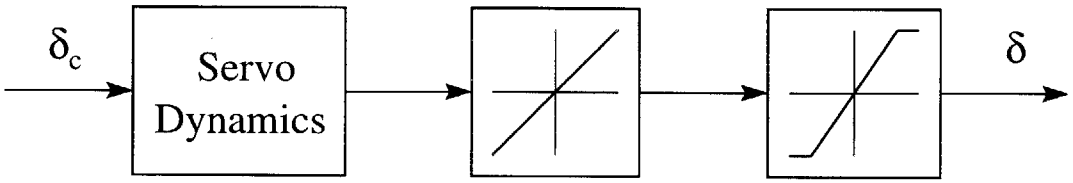

Figure 4-2: Control Surface Model for Estimating $u(t)$ and $\Delta u(t)$

information required to implement this surface model is a model of the linear servo dynamics and the saturation characteristics. This information is readily available for all commercial transports.

4.5 Control Surface Saturation Levels $\Delta u(t)$

The final set of signals that is required is the control surface saturation levels $\Delta u(t)$. Recall from Section 2.1 and Equation (2.8) that the control surface saturation level is defined as the difference between the actual control surface deflection u and the unsaturated control surface deflection $u_k + u_{rcm}$:

$$\Delta u(t) = u - (u_k + u_{rcm}) \tag{4.12}$$

Based on the discussion in the previous section, equivalent control surface saturation levels are defined in order to reduce the number of signals. Table 4.7 contains the generic definitions of the control surfaces deflections that are used in subsequent chapters. In this table, the subscripts l and r signify left and right surfaces saturation levels, the summation accounts for multiple left and right surfaces, and k is an averaging constant. As before, the selection of the averaging constant is up to the designer and generally reflects the number and type of control surface.

Table 4.7: Generic Definitions of Control Surface Saturation Levels $\Delta u(t)$

Variables	Description
Δu_{sym}	$\sum_i (\Delta u_{l_i} + \Delta u_{r_i})/k$
Δu_{dif}	$\sum_i (\Delta u_{l_i} - \Delta u_{r_i})/k$

The generation of $\Delta u(t)$ can be accomplished using direct measurement and a surface model or just a surface model. Direct measurements alone cannot be used since the unsaturated control surface deflection $u_k + u_{rcm}$ is obtained by filtering the control surface command $u_{k_c} + u_{rcm_c}$ through the servo dynamics is Figure 4-2.

Since $\Delta u(t)$ is used during post-failure operation, special consideration must be given to the saturation level associated with the failed surface. Recall from Section 2.4 that the $\Delta u(t)$ signal is used to robustify the adaptation algorithm to potentially destabilizing control surface saturation. This robustification was performed by subtracting out the acceleration due to the saturating surface $CB_p\Delta u(t)$ in the input error, which is duplicated here for convenience:

$$e_i(t) \quad = \quad \hat{J}(t)\left[\dot{z}(t) - C(A_m x(t) + B_m r(t-\tau) + f_m + B_p \Delta u(t))\right]$$

If a control surface fails, its contribution to $CB_p\Delta u(t)$ should be zero. This fact can be arrived at by two separate arguments. First, it is clear that if a surface experiences a fixed position failure, the surface is not moving; thus, it does not really experience saturation. As a result, (4.12) does not accurately portray the saturation level of a failed surface. Stated another way, a failed surface represents a null column in the control effectiveness matrix in the LTI framework, i.e. the linearization of a constant is zero. This argument is extended to floating surface failures since the failed deflection is a function of the aircraft state; thus changing CB_p. In any event, we can for this discussion assume that for a failed surface,

$$CB_p\Delta u(t)|_i = 0 \;\forall\; t \geq t_f, \tag{4.13}$$

where i represents the failed control surface and t_f is the failure time. However, given that *a priori* knowledge of the failure is not required in this architecture, $CB_p|_k \neq 0 \;\forall\; t \geq t_f$ where

k represents the column associated with the failed surface. Likewise, based on the definition of the control surface saturation level (4.12), $\Delta u(t)|_j \neq 0 \ \forall \ t \geq t_f$ where j represents the failed surface. As a result, the equality (4.13) is not enforced, i.e. $CB_p \Delta u(t)|_i \neq 0 \ \forall \ t \geq t_f$!

Feeding this erroneous signal into the input error can result in poor reconfiguration results. There are two possible solutions to this problem. If the surfaces are directly measurable, the failed surface can be determined via logic, and the resulting saturation level can be nulled, i.e. $\Delta u(t)|_j = 0 \ \forall \ t \geq t_f$ where j represents the failed surface. If the surfaces are not directly measurable, there is no simple way to determine which surface has failed. Using the alternative viewpoint from above, we can say that CB_p is uncertain and added it to the unknown reconfiguration parameters $\Phi(t)$ in the adaptation regressor. This will be demonstrated below. Recall the robust adaptation law, which is defined in Equation 2.20:

$$\dot{\Phi}(j,:) = \begin{cases} -e_i(j) Z^T \Gamma & \text{if } |e_i(j)| \geq V_o(j) \\ 0 & \text{else} \end{cases}$$

for $j = 1 : m$ and where $|V| \leq V_o$ is the upper bound on the regressor disturbance, $Z = [x, r, 1, \dot{z} - (CA_m x + CB_m r + Cf_m + CB_p \Delta u)]^T \in \mathbb{R}^{n+m+1+m}$, $\Phi = [\tilde{K}_r \ \tilde{K}_x \ \tilde{K}_f \ \tilde{J} \ ] \in \mathbb{R}^{m \times (n+m+1+m)}$, and the adaptation gain Γ is a symmetric positive definite. Now, removing CB_p from the regressor results in the following measurement error $Z = [x, r, 1, \dot{z} - (CA_m x + CB_m r + Cf_m), \Delta u]^T \in \mathbb{R}^{n+m+1+m+m}$ and the reconfiguration parameter matrix $\Phi = [\tilde{K}_r \ \tilde{K}_x \ \tilde{K}_f \ \tilde{J} \ \tilde{J}' \ ] \in \mathbb{R}^{m \times (n+m+1+m+m)}$ where $\tilde{J}'$ is an estimate error of $(CB_u)^{-1} CB_p$.

4.6 Summary

The primary function of the signal conditioning and estimation module is to provide the required signals to the adaptive augmentor and reference model components. As outlined in the previous two chapters, the required signals include the stability axes state vector $x(t)$, the stability axes state derivative vector $\dot{x}(t)$, the pilot input commands $r(t - \tau)$, the control surface deflections $u(t)$, and the control surface saturation levels $\Delta u(t)$.

The primary purpose of this chapter was to present definitions of signals that are used in subsequent chapters and to address the potential complexity issues associated with providing the required signals. Depending on the available sensors suites, Kalman filters may be required to generate $\dot{x}(t)$ and control surface models may be required to generate $u(t)$ and $\Delta u(t)$. Other issues addressed include conversion of the rotation rates from body-axes to stability axes, determination of the pilot reference commands with the auto-pilot engaged, determination of the reference input delay τ, and determination of the saturation level $\Delta u(t)$ for the failed surface.

THIS PAGE LEFT BLANK INTENTIONALLY

Part II

Implementation

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 5

F/A-18 Simulation Results

The reconfiguration architecture discussed in the previous sections was implemented on a high-fidelity F/A-18 simulator. The purpose of this implementation is to demonstrate the retrofit reconfiguration at a single flight point and to serve as a precursor to the implementation on a generic commercial transport simulation. The chapter will present a brief discussion of the implementation parameters and will present several failure scenarios.

5.1 Implementation

The F/A-18 simulator used in this implementation is a Simulink version of NASA Dryden's high-fidelity, fixed base, piloted F/A-18 simulator. The Simulink version, illustrated in Figure 5-1, is a complete nonlinear 6 degree-of-freedom simulator that includes full envelope aerodynamics, engine dynamics, actuator dynamics, and the Control Augmentation System (CAS). The Simulink version has been validated against the fixed-base piloted simulator and has been used successfully for design modifications that resulted in piloted simulations [79, 80].

The flight point selected for the evaluation is Mach 0.5 at an altitude of 15,000 feet. At this flight point, the aircraft is trimmed at $\alpha = 4.9^o$ and a true airspeed of $V_t = 313$ kts. Atmospheric disturbances, sensor dynamics and noise are neglected in the following simulations.

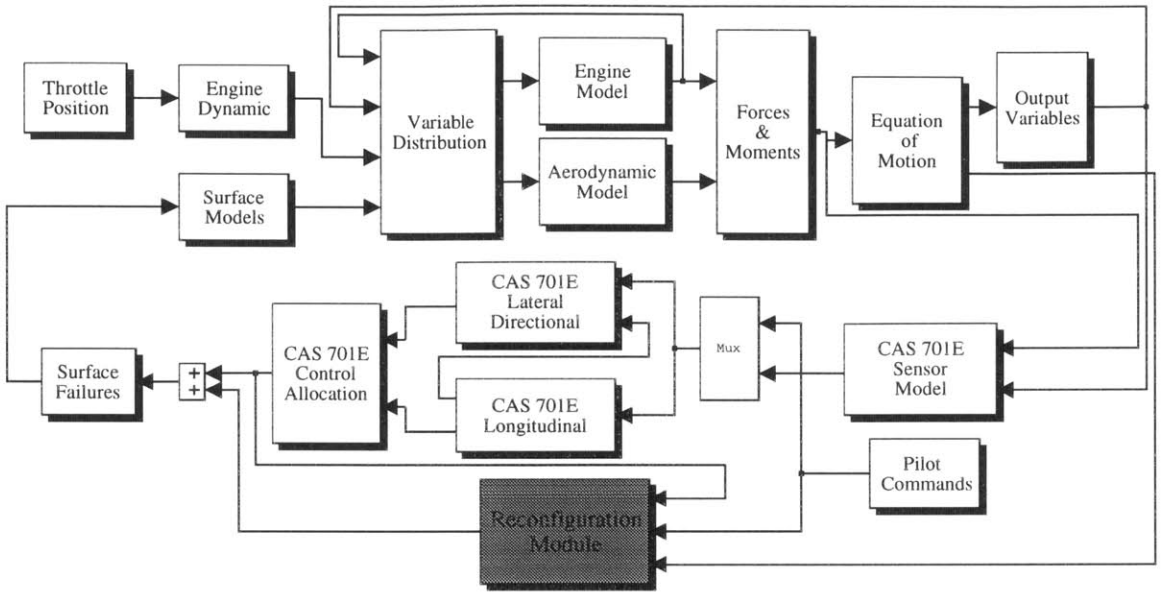

Figure 5-1: F/A-18 6DOF Nonlinear Simulator

In this simulator, the pilot commands aircraft rotational rates; thus, the performance vector z contains p, q, and r. The reconfiguration architecture outlined in the previous section requires the following information from the reference model: desired linear dynamics CA_m and CB_m, desired trim condition CF_m, the time delay τ associated with each pilot input channel, and the control power dimensional derivatives CB_p associated with the rotational rates. As stated, the intention of this implementation is to demonstrate 'Adaptive Augmentation' at a single flight point; thus, online adaptation of the reference model is not included. To generate the required reference model information, off-line system identification was used to determine the model structure in (2.11) and a separate off-line identification was performed to determine the control power derivatives CB_p using the model structure in (2.2). Minimum variance, equation-error estimation techniques [18, 17] were used, and special attention was devoted to ensuring that the modeling error is small. The level of modeling error achieved eliminated the tradeoff between the upper bound of modeling error and the achievable performance inherent in robust adaptation.

A reconfiguration module was added to the baseline F/A-18 simulator. The inputs to the reconfiguration module include the aircraft's total stability axis states $x = [V_t\ \alpha\ \theta\ q\ \beta\ \phi\ p\ r]^T$, the stability axis rotation accelerations $\dot{z} = [\dot{p}\ \dot{q}\ \dot{r}]^T$, pilot stick and pedal inputs traditionally

associated with elevator, aileron, and rudder commands $r = [dep\ dap\ drp]^T$, and the nominal control command u_{k_c} from the CAS. For this implementation, signal conditioning and estimation were not required since atmospheric and sensor noise were not included. Furthermore, the rotational accelerations were measured directly. The reconfiguration module contains a surface module which was used to determine the level of saturation.

The reconfiguration module outputs perturbation signals to the control surfaces. The control surfaces used for the reconfiguration task are symmetric horizontal tail for pitch, differential aileron for roll, and symmetric rudder for yaw[1]. Because of the redundancy of the surfaces on the F/A-18, other choices are possible. However, we have chosen one surface per axis for reconfiguration to be consistent with the eventual application to civil aviation.

Initial estimates for the control gains in (2.14) are zero to reflect the nominal condition, and the initial estimate of $\hat{J}$ in (2.15) is set to $(CB_p)^{-1}$ which is obtained from the reference model. The adaptation gain Γ in (2.20) was selected to normalize the regressor and to permit rapid adaptation to failure disturbance while not chattering. Finally, because the modeling errors in the reference model are minor and the pilot commands used are moderate, a dead-zone was not required.

5.2 Simulation Results

In this section, a few simulation results will be presented. The failure scenarios illustrated are a rudder hard-over failure, a stabilator hard-over failure, and a stabilator floating failure. For all of the simulation results that follow, pilot stick inputs are independent of the failure, i.e. the pilot is operating 'open-loop'. The input command consists of two pitch doublets followed by a roll doublet.

[1]Symmetric deflection is defined as $(\delta_{left} + \delta_{right})/2$ and differential deflection is defined as $(\delta_{left} - \delta_{right})/2$

5.2.1 Rudder Failure

The first failure scenario that is presented is a hard-over failure to the left rudder. At 2.5 seconds into the simulation, the left rudder moves from its trim deflection to a 30^o deflection. For the F/A-18, the rudder limits are 30^o to -30^o. Figures 5-2 and 5-3 illustrate the longitudinal and lateral-directional response to the input command.

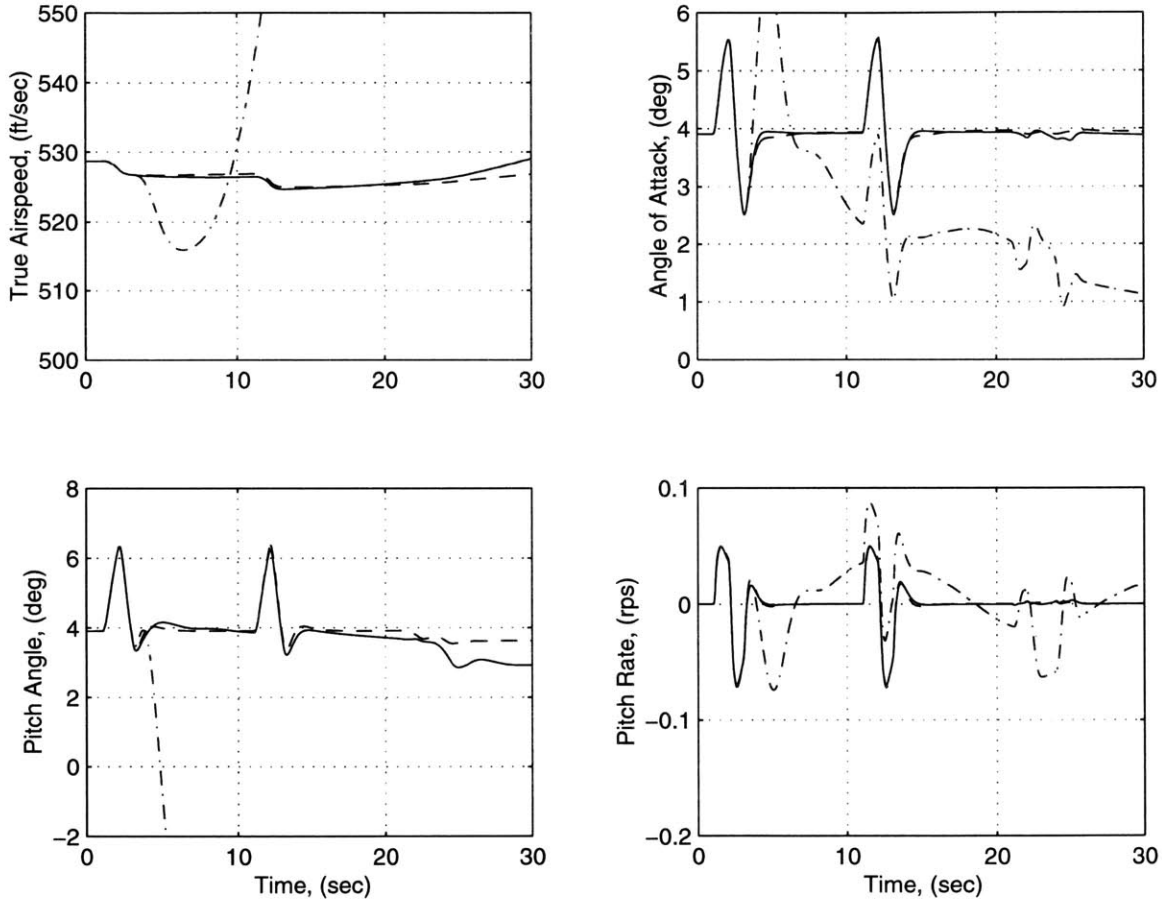

Figure 5-2: Longitudinal Response for Rudder Hard-Over Failure: $\delta_{rud_l} = 30^o$ (— Failure - - Nominal - · - Failure w/o RCM)

These figures show that without reconfiguration, the aircraft departs rapidly. With reconfiguration, however, the failure disturbance is rapidly attenuated and good performance is recovered. Yaw rate and sideslip performance are degraded, primarily due to significant magnitude saturation of the unfailed rudder. To attenuate the large DC yaw disturbance, the reconfiguration

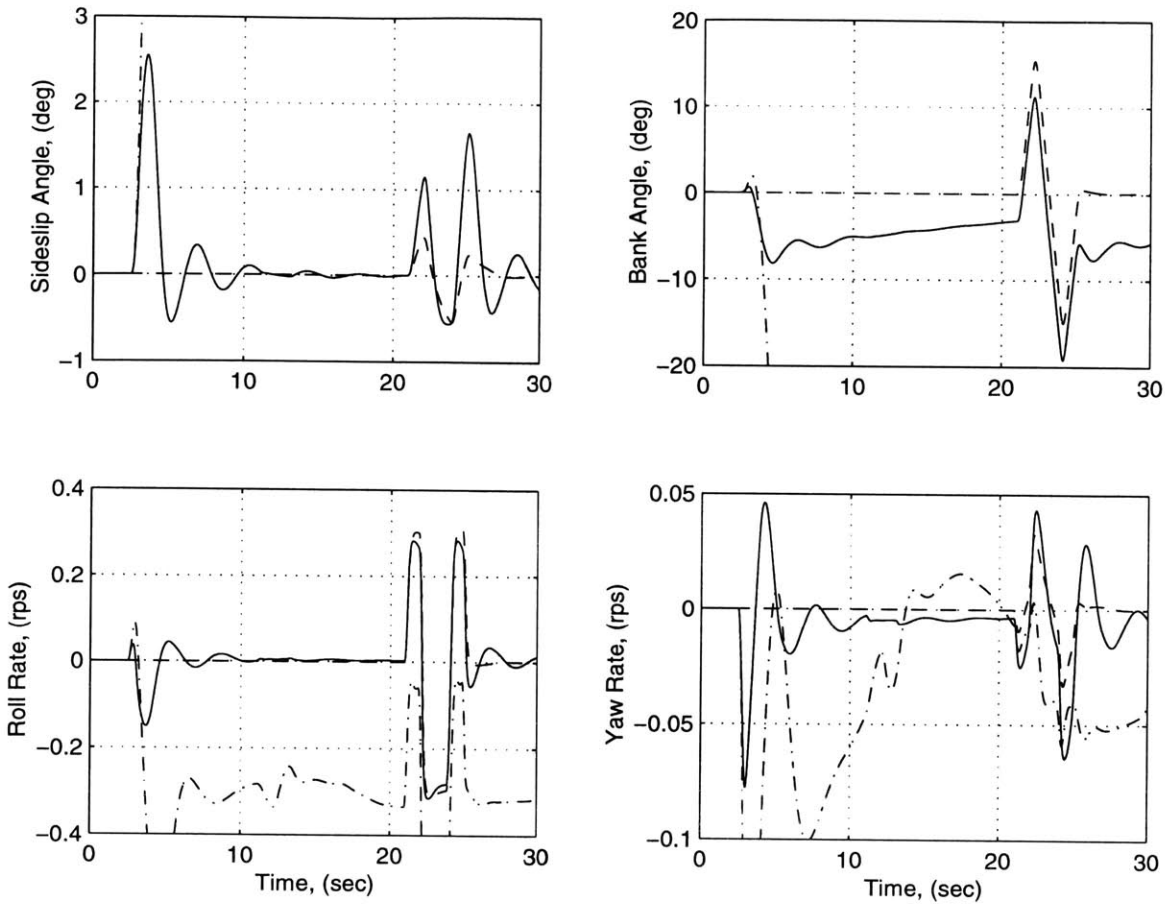

Figure 5-3: Lateral-Directional Response for Rudder Hard-Over Failure: $\delta_{rud_l} = 30^o$ (— Failure - - Nominal - · - Failure w/o RCM)

module commands the right rudder to approximately -30^o deflection; from that deflection, the rudder can only move in the positive direction during the roll doublet command (21-25 seconds). It is worth noting that the reconfigured aircraft remains stable in the presence of significant saturation. Finally, the slow decay shown in the bank angle response is attributed to the long time constant of the spiral mode and to the fact that the pilot makes no corrections (i.e. the input commands do not change).

5.2.2 Horizontal Tail Failure

The next failure scenario is a left horizontal tail hard-over failure. At 2.5 seconds into the simulation, the left horizontal tail moves to its upper limit of 10.5^o deflection. For the F/A-18, the horizontal tail limits are 10.5^o to -30^o. Figures 5-4 and 5-5 illustrate the longitudinal and lateral directional response to the open-loop reference commands.

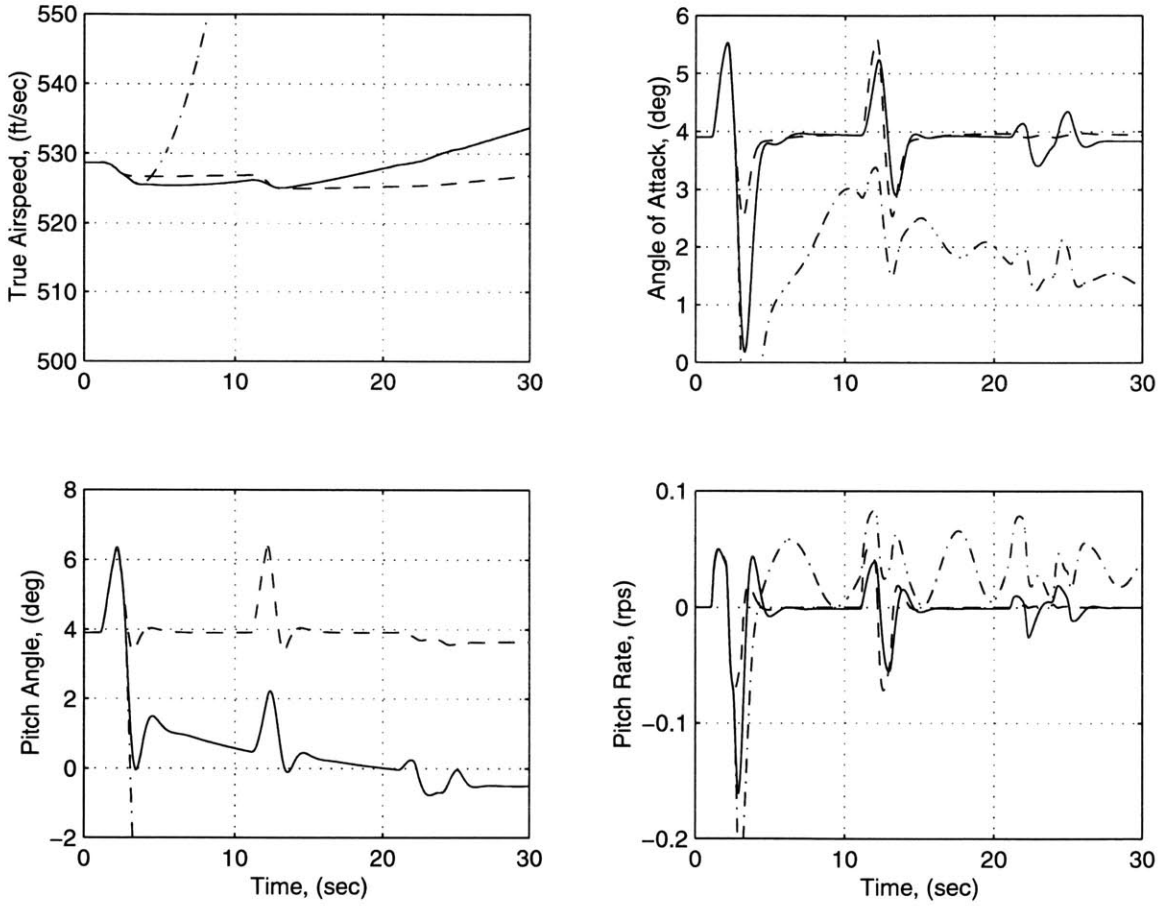

Figure 5-4: Longitudinal Response for Horizontal Tail Hard-Over Failure: $\delta_{hzt_l} = 10.5^o$ (— Failure - - Nominal - · - Failure w/o RCM)

In these figures we see that without reconfiguration, the aircraft departs rapidly in both pitch and roll. But, with reconfiguration, the failure disturbance is rapidly attenuated and near nominal performance is recovered. By the second pitch rate doublet, the reconfigured aircraft is able to generate the desired pitch rate and resulting angle of attack response. The pitch

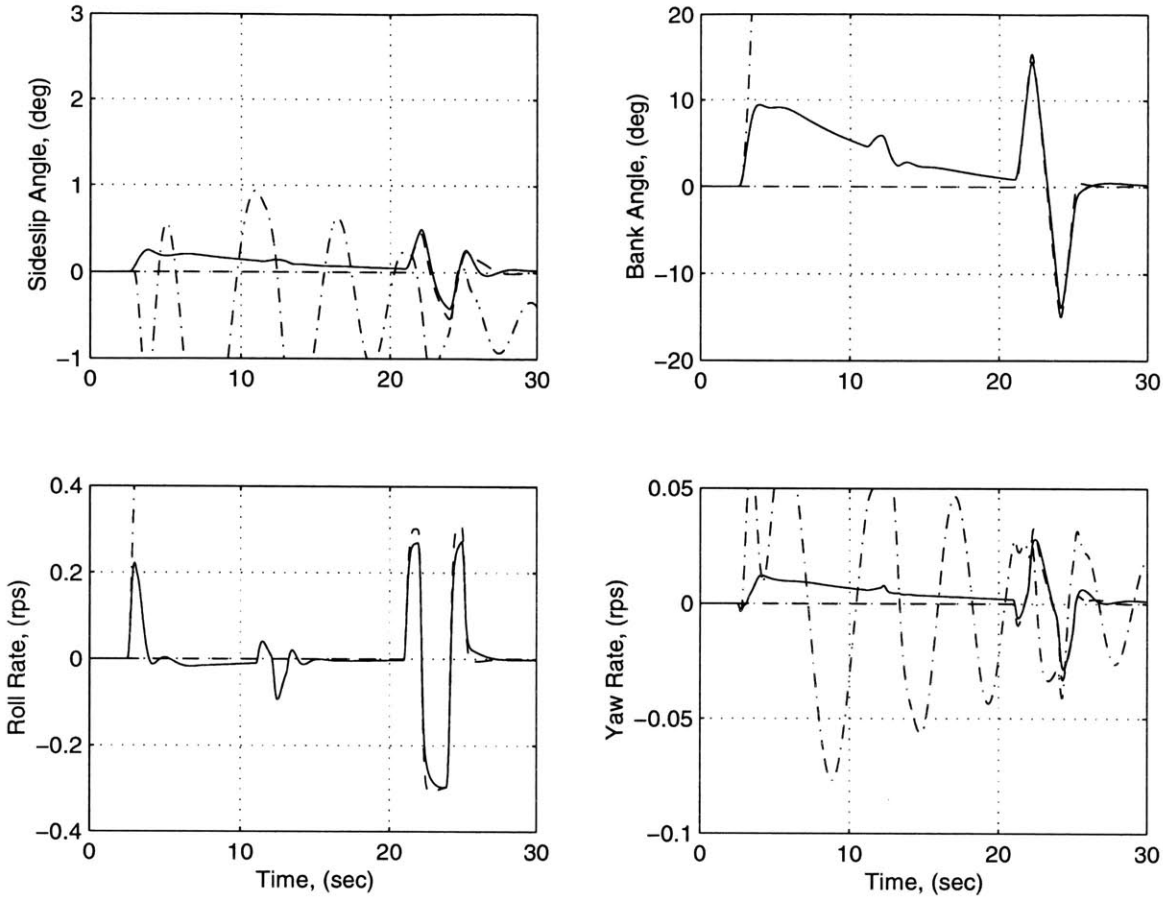

Figure 5-5: Lateral-Directional Response for Horizontal Tail Hard-Over Failure: $\delta_{hzt_l} = 10.5^o$ (— Failure - - Nominal - · - Failure w/o RCM)

angle offset is a result of the fact that the pitch angle is the integral of the pitch rate so that the pitch rate disturbance is integrated. Closed-loop pilot commands would correct this discrepancy. For the roll rate doublet, the nominal performance is achieved despite the fact that differential horizontal tail is required for this maneuver. Finally, it is seen that the reconfigured performance does exhibit some cross-axis coupling, indicating that the control gains have not converge to their optimal values. Nevertheless, stable and near nominal performance is achieved.

5.2.3 Floating Horizontal Tail Failure

The next failure scenario is a left horizontal tail hard-over failure. At 0.25 seconds into the simulation, the right horizontal tail floats, i.e. $\delta_{hzt_r} = \alpha$ to 1st order. Figures 5-6 and 5-7 illustrate the longitudinal and lateral directional response to the open-loop reference commands.

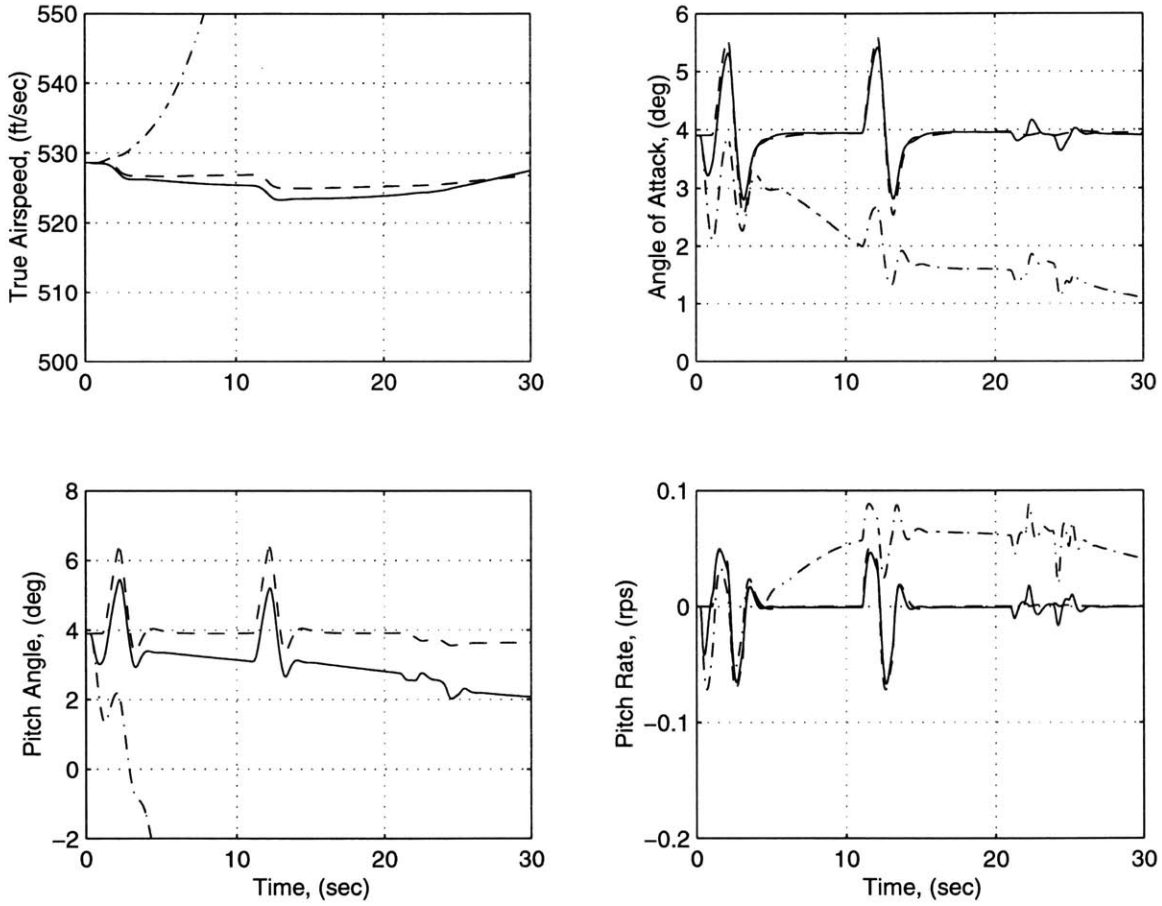

Figure 5-6: Longitudinal Response for Floating Horizontal Tail Failure: $\delta_{hzt_l} = \alpha$ (— Failure - - Nominal - · - Failure w/o RCM)

Again, the aircraft departs rapidly without reconfiguration. With reconfiguration, the failure disturbance is rapidly attenuated and near nominal performance is recovered.

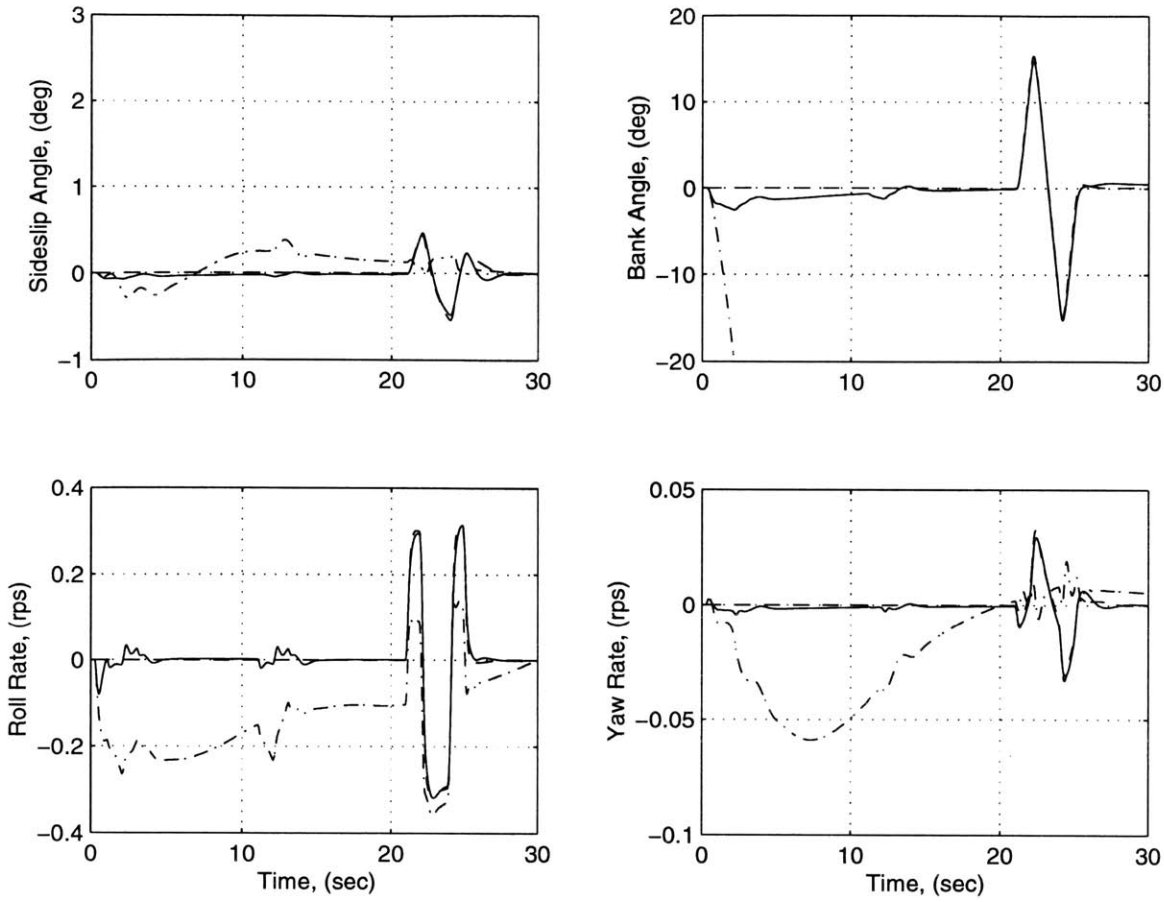

Figure 5-7: Lateral-Directional Response for Floating Horizontal Tail Failure: $\delta_{hzt_l} = \alpha$ (— Failure - - Nominal - · - Failure w/o RCM)

5.3 Conclusions

The purpose of this implementation was to demonstrate the modular nature of the architecture and to provide initial simulation results. A wide variety of control surface failures were performed and reported: rudder fixed-position failure, stabilator fixed position failure, stabilator floating failure. Comparisons were made between the failed aircraft with and without reconfiguration and between the failed-with-reconfiguration and the healthy aircraft. Simulation revealed that the retrofit module was able to stabilize the aircraft in the presence of large failure disturbances and control surface saturation. Furthermore, nominal performance was achieved if there was adequate control authority post-failure. Without reconfiguration, the aircraft departs for the

failures considered.

Chapter 6

ACFS Implementation

This chapter details the implementation of the retrofit reconfiguration system on a generic commercial transport simulation. The transport simulator used is the Advanced Concept Flight Simulator (ACFS) that resides at NASA Dryden Flight Research Center.

In the following sections, a description of the aircraft and simulation will be presented. A detailed account of the implementation will be discussed, and general issues of commercial transport reconfiguration will be highlighted. Finally, simulation results that compare the nominal aircraft to a failed aircraft with and without reconfiguration will be presented.

6.1 Aircraft and Simulator Description

The ACFS is a high fidelity, six degree-of-freedom flight simulator that represents a generic narrow-body commercial transport aircraft. The ACFS provides full mission functionality and appears similar to a high fidelity training simulator but is unique in that no flight hardware is used [81]. The physical dimensions and performance are similar to the Boeing 757-200 [82] aircraft with the exception that the ACFS has a T-tail. Figure 6-1 presents a schematic representation of the planform layout.

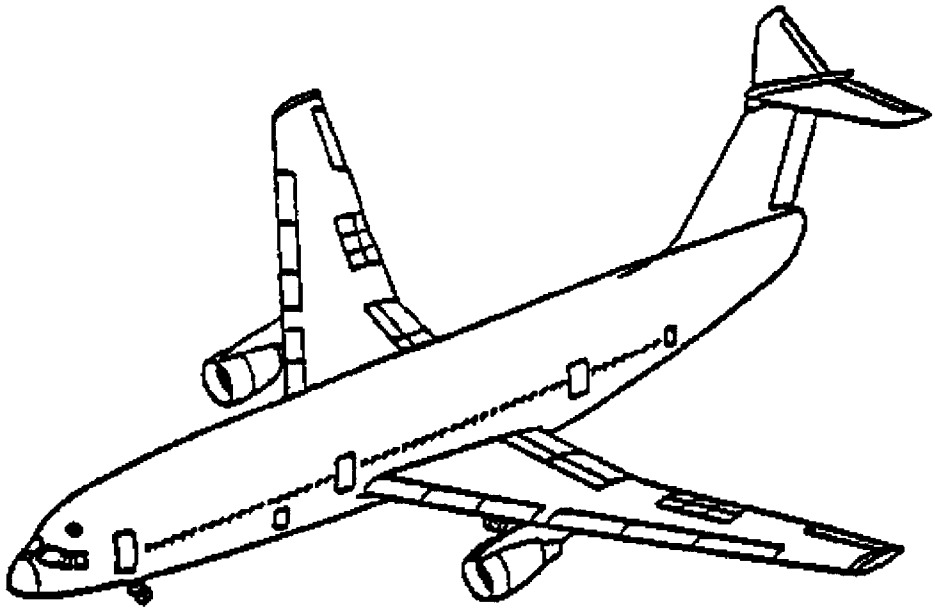

Figure 6-1: Schematic of ACFS Planform Layout

The ACFS was originally developed as a joint venture between NASA Ames, NASA Langley, and Lockheed Company. The simulator has primarily been used to study many aspects of human factors in aviation safety as well as methods to improve aviation operation efficiency. Outside the humans factors and air-traffic control research, the simulator has also been used to study Propulsion Controlled Aircraft (PCA) [83], which is an emergency backup fly-by-throttle control system for use in the case of complete hydraulic failure.

The aircraft has two engines mounted on the wings, and each engine produces a maximum of 41,000 lbs of thrust. The gross take-off weight is 225,000 lbs, and can accommodate 200 passengers. The ACFS control surface layout is typical for this class of aircraft. Table 6.1 lists the control effectors and their respective position and rate limits. The aircraft has an elevator which acts as the primary pitch device, and a variable incidence horizontal tail which acts as the primary trim device. The lateral axis consists of a pair of outboard ailerons and six spoiler panels, three on each wing. In the directional axis, there is a single full span rudder. For high-lift devices, the wing has trailing edge flaps that extend to the ailerons and full span leading edge flaps. The aircraft has four ground spoilers extending from the wing/fuselage intersection to the

wing crank. Finally, the engine throttle position, also known as Power Lever Angle (PLA), is included as a control effector since symmetrical and differential throttles can produce both pitch and yaw moments.

Table 6.1: Control Surface Position and Rate Limits

Control Effector	Position (^{o}) Limit	Rate ($^o/sec$) Limit
Elevator	$-25.0 \rightarrow 25.0$	80.0
Horizontal Tail	$-12.0 \rightarrow 4.0$	0.15
Ailerons	$-25.0 \rightarrow 15.0$	80.0
Rudder	See Figure 6-2	80.0
Spoilers	$-60.0 \rightarrow 0.0$	80.0
Flaps	$0.0 \rightarrow 40.0$	$\sim$
Slats	$0.0 \rightarrow 22.0$	$\sim$
PLA	$5.0 \rightarrow 100$	3.57

The rudder deflection position limits vary as a function of dynamic pressure due to empennage loads considerations. Figure 6-2 illustrates the deflection limits. It is seen from this figure that the allowable rudder deflection drops exponentially for increasing dynamic pressure, and at cruise dynamic pressures, the allowable deflection is below 8.0^o.

The aircraft model includes full envelope aerodynamics and full envelope engine dynamics. The surface dynamics account for position and rate saturation but lack servo dynamics. The aircraft model does contain a turbulence model with RMS and bandwidths that are representative of values specified in Military Specifications 8785 D of April 1989. No sensor dynamics or sensor noise are included.

The version of the ACFS that was used for this implementation is referred to as the 'Stone Soup' Simulator of the Miniature Advanced Concept Flight Simulator (sss.miniACFS). The miniature version represents a desktop computer version of the full motion, piloted simulator that resides at the NASA Ames Crew-Vehicle Systems Research Facility. The stone soup moniker represents a version of the ACFS in which the proprietary portions of the simulator have been replaced to enable limited distribution. In this version, the flight management, flight control, and autopilot systems were replaced. Of particular interest, the flight control laws were replaced by modified

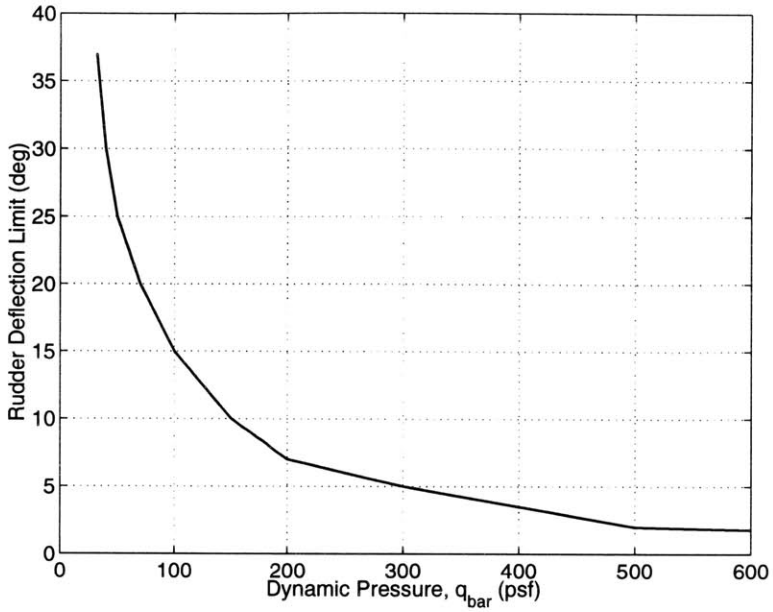

Figure 6-2: Rudder Deflection Limit

Boeing 737 point-mass control laws obtained from MIT's Aeronautical Systems Laboratory's (ASL). From this point on, the acronym ACFS will refer to the Stone Soup Simulation of the Miniature Advanced Concept Flight Simulator.

The desktop simulator provides graphical displays for the various control panels and cockpit displays. The relevant displays for this implementation include the Primary Flight Displays (PFDs), which include an Attitude Direction Indicator (ADI) and a Horizontal Situation Indicator (HSI). There are also Secondary Flight Displays (SFDs), which provide information on aircraft systems. Hardware devices are emulated through additional graphical displays. The Mode Control Panel (MCP) provides access to the autopilot and autothrottle systems. The Aircraft Control Panels (ACPS) provide access to additional hardware control such as the stick, throttles, flaps, speed brake, and landing gear. All pilot inputs are introduced using a mouse by clicking on the respective object on the computer monitor.

The ACFS is configured to run on one or more SGI workstations. The source code has evolved over more than 20 years and is composed of both FORTRAN and C programming languages. The architecture of the ACFS is based upon a client/server approach. The aircraft model and

graphical displays run in parallel, and execution frequency and frame timing is controlled by the programmer. The core program runs at 30 Hz, and is implemented in a discrete fashion.

6.2 Reconfiguration Module Implementation

This section discusses the implementation of the reconfiguration module into the ACFS architecture. Each component of the RCM will be presented: the signal conditioning and estimation component, then the reference model component, and finally the adaptive augmentor component. The goal of the discussion is to provide the foundation of the simulation results that follow while highlighting issues associated with commercial transport reconfiguration.

The reconfiguration module was added to the baseline ACFS by using the PCA implementation as a template. The reconfiguration module was included as an additional process that runs in parallel to the rest of the simulator. The RCM augmentor module was added to the process list and given an execution priority equivalent to the flight dynamics, and frame timing was performed after the aircraft model was executed. The required inputs to each of the module components were obtained by accessing the shared memory structures, and the outputs were injected to the individual surfaces in a similar manner.

Given the modular nature of the RCM architecture, the steps required to implement the module are:

- Gather the required inputs,
- Devise a reconfiguration control allocation scheme, and
- Add the reconfiguration signals to the control surface commands.

Unlike the F/A-18 implementation, the implementation of these steps was non-trivial. Initial simulation demonstrated some anomalies that had to be resolved in order to proceed with the reconfiguration study. A majority of the anomalies were attributed to the shared memory, aerodynamic model, and the substituted control laws. The original intent of this simulator was to perform air traffic control and cockpit/pilot interaction research. Due to anomalies,

the implementation required two months of effort to generate equivalent results to the F/A-18 implementation which required two weeks. Of this two months, 75% of the time was devoted to resolving the anomalies. This required an in-depth knowledge of the aerodynamics and control characteristics, which should not be required in a normal implementation.

6.2.1 Signal Conditioning and Estimation Component Implementation

The signal conditioning and estimation module provides the required signals to the reference model and adaptive augmentor components of the RCM architecture. The required signals include the stability axes state vector $x(t)$, the stability axes state derivative vector $\dot{x}(t)$, the pilot input commands $r(t-\tau)$, the control surface deflections $u(t)$, and the control surface saturation levels $\Delta u(t)$. The determination of each of these signals will follow.

State Vector $x(t)$

The state vector $x(t)$ consisted of the stability axes states: $x(t) = [v_t \ \alpha \ \theta \ q \ \beta \ \phi \ p \ r]^T$. As expected, all of these states were accessible via shared memory since they are used in the nominal controller; however, the rotational rates are in body axes. The conversion of the rotation rate from body-axes to stability axes is as follows:

$$\begin{bmatrix} p_s(t) \\ q_s(t) \\ r_s(t) \end{bmatrix} = S_\alpha \begin{bmatrix} p_b(t) \\ q_b(t) \\ r_b(t) \end{bmatrix} \quad \text{where} \quad S_\alpha = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{6.1}$$

where the subscripts s and b stand for stability-axes and body-axes. Table 6.2 summaries the state vector $x(t)$ used for the ACFS implementation.

Table 6.2: Aircraft States $x(t)$ for ACFS Implementation

Variables	Description	Units
$v_t(t)$	True Airspeed	ft/sec
$\alpha(t)$	Angle of Attack	rad
$\theta(t)$	Pitch Angle	rad
$q_s(t)$	Pitch Rate	rad/sec
$\beta(t)$	Sideslip Angle	rad
$\phi(t)$	Bank Angle	rad
$p_s(t)$	Roll Rate	rad/sec
$r_s(t)$	Yaw Rate	rad/sec

State Derivative Vector $\dot{x}(t)$

For the state derivative vector $\dot{x}(t)$, not all of the derivatives where available via shared memory. The derivatives that are accessible include the linear accelerations in body-axes, the Euler angle derivatives, and the rotational accelerations in body-axes. This data was used in conjunction with kinematic and transform relationships to obtain the stability axes state derivative vector. First, the true airspeed and aerodynamic angles derivatives can be obtained by taking the derivative of there respective definitions. As illustrated in [71]:

$$\begin{aligned} \dot{v}_t &= \frac{U\dot{U} + V\dot{V} + W\dot{W}}{v_t} \\ \dot{\alpha} &= \frac{U\dot{W} - W\dot{U}}{U^2 + W^2} \\ \dot{\beta} &= \frac{v_t\dot{V} - V\dot{v}_t}{v_t^2 \cos\beta} \end{aligned} \tag{6.2}$$

where U, V, and W are the translational body-axes velocities in the X, Y, and Z directions. Next, the body axes rotational accelerations are converted to stability axes. To perform this

conversion to stability axes, the derivative of (6.1) is required:

$$\begin{bmatrix} \dot{p}_s(t) \\ \dot{q}_s(t) \\ \dot{r}_s(t) \end{bmatrix} = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} \dot{p}_b(t) \\ \dot{q}_b(t) \\ \dot{r}_b(t) \end{bmatrix} + \begin{bmatrix} -\dot{\alpha}\sin\alpha & 0 & \dot{\alpha}\cos\alpha \\ 0 & 0 & 0 \\ -\dot{\alpha}\cos\alpha & 0 & -\dot{\alpha}\sin\alpha \end{bmatrix} \begin{bmatrix} p_b(t) \\ q_b(t) \\ r_b(t) \end{bmatrix}$$

Table 6.3 summaries the state vector $x(t)$ used for the ACFS implementation. Also, given Assumption 1, the performance vector derivative $\dot{z}(t)$ contains the stability axes rotational accelerations which are summarized in 6.4.

Table 6.3: Aircraft States $\dot{x}(t)$ for ACFS Implementation

Variables	Description	Units
$\dot{v}_t(t)$	True Airspeed Derivative	ft/sec^2
$\dot{\alpha}(t)$	Angle of Attack Derivative	rad/sec
$\dot{\theta}(t)$	Pitch Angle Derivative	rad/sec
$\dot{q}_s(t)$	Pitch Acceleration	rad/sec^2
$\dot{\beta}(t)$	Sideslip Angle Derivative	rad/sec
$\dot{\phi}(t)$	Bank Angle Derivative	rad/sec
$\dot{p}_s(t)$	Roll Acceleration	rad/sec^2
$\dot{r}_s(t)$	Yaw Acceleration	rad/sec^2

Table 6.4: Performance Vector Derivative $\dot{z}(t)$ for ACFS Implementation

Variables	Description	Units
$\dot{q}_s(t)$	Pitch Acceleration	rad/sec^2
$\dot{p}_s(t)$	Roll Acceleration	rad/sec^2
$\dot{r}_s(t)$	Yaw Acceleration	rad/sec^2

For existing commercial transports, many of the state derivatives in $\dot{x}(t)$ will not be accessible from the nominal control system ; thus, they must be estimated. The state derivatives will be estimated using a bank of Kalman filters: each state will have an associated Kalman filter. The impact of using state derivative estimates $\hat{\dot{x}}(t)$ in the context of system identification and computation of the output performance error $e_o(t)$ will be explored in Section 6.3 of this chapter. Here, the issues associated with the implementation of the Kalman filters will be presented.

As stated in Section 4.2, in order to implement the Kalman filters, the process noise covariance Q and measurement noise covariance R must be determined for each element of the state vector. Since the measurement noise covariance is a function of the sensor hardware, the process noise covariance is the only parameter available to tune the Kalman filter. The selection of the process noise covariance affects the estimator's bandwidth, estimation time delay τ, and estimation error covariance matrix $P(t)$. Due to system identification considerations, the estimator's bandwidth is the critical performance measure that should be used to determine the process noise covariance Q.

To tune the filter, the bandwidth requirements must be determined by considering the physics of the signal. For example, if an estimate of the pitch acceleration is required, the bandwidth is based on maximum achievable pitch jerk. Note, maximum achievable jerk is the product of the maximum achievable moment times the rate limit. Given the bandwidth requirements, the process noise covariance can be determined using the undamped natural frequency $\Omega = (Q/R)^{\frac{1}{6}}$ of the estimator. Likewise, an estimate of the maximum time delay τ_{max} and an estimate of the estimation error covariance P_∞ can be made via (4.7) and (4.5). The time delay information will be used in the identification process to ensure that all data is transformed to equivalent times.

For this implementation, the measurement noise covariances were assumed to have unity magnitudes. This value is quite large and is unrepresentative of most sensors; however, its actual value only affects the predicted value of the estimation error covariance P_∞, which is not all that important. All that is desired is a signal that can be used to identify the plant and used in the computation of the output performance error $e_o(t)$. As stated above, these results will be presented in Section 6.3.2. Viewing the physics of each element of the state derivative vector $\dot{x}(t)$, the process noise covariance were determined. The process noise covariances for the estimation of the rotational accelerations $\dot{p}, \dot{q}$ and $\dot{r}$ are $Q = 10^7$. Using these process noise covariances, the Kalman filters were able to capture the high frequency dynamics of the aircraft, i.e. pitch, roll and yaw jerks (rate of change of acceleration). As for the other required state derivatives, the process noise covariance for true airspeed derivative $\dot{v}_t$ is $Q = 10^{-2}$, for angle of attack derivative $\dot{\alpha}$ is $Q = 10^5$, and for angle of sideslip derivative $\dot{\beta}$ is $Q = 10^2$.

Pilot Input Vector $r(t-\tau)$

The next signal that is required is the pilot input $r(t-\tau)$. The pilot input commands included the pitch stick input $dep(t)$, the roll stick input $dap(t)$, and the yaw pedal input $drp(t)$. The generation of these signal requires special consideration when the autopilot is engaged. In general, the autopilot functions as an outer loop control system that generates equivalent pilot inputs to perform a specified task, i.e, attitude hold, altitude hold, heading hold, etc. Given that we desire to perform both identification and reconfiguration with and without the autopilot engaged, we need to be able to measure or construct $r(t-\tau)$ with or without the autopilot engaged. For some commercial transports with yoke columns, autopilot commands are converted to equivalent stick inputs to back-drive the stick; thus, providing a cue to the pilots of the autopilots commands. Given this situation, stick and pedal movements are directly measurable and no signal conditioning or estimation is required.

For other transports, especially with side stick devices, autopilot commands do not back-drive the stick. In this situation, the signal conditioning and estimation module must generate the equivalent stick commands as though they were to be used to back-drive the stick.

For this implementation, actual pilot inputs were obtained by accessing the shared memory structures, and the autopilot was not 'intentionally' used. However, the longitudinal inner-loop control system defaults to an attitude hold mode for stick-free operations, i.e. no stick inputs. Recall from Section 6.1 that the proprietary control laws were replaced with point-mass control laws; an attitude hold mode for the longitudinal inner loop is not representative of operational control laws. In any event, the attitude hold mode commands were converted to equivalent pitch stick commands by inverting the stick logic.

Finally, the reference input delay τ, Section 4.3, needs to be determined. For the ACFS, input delays were negligible. Table 6.5 summaries the state vector $x(t)$ used for the ACFS implementation.

Table 6.5: Pilot Inputs $r(t-\tau)$ for ACFS Implementation

Variables	Description	Units
$dap(t)$	Roll Stick Input	in
$dep(t)$	Pitch Stick Input	in
$drp(t)$	Yaw Pedal Input	in

Control Surface Deflection Vector $u(t)$

The next set of signals required are the control surface deflections. As illustrated in Section 6.1, the ACFS has many control effectors that can be used for reconfiguration. Define the following deflections:

$$\begin{aligned}
\delta_{elv_{sym}} &= \delta_{elv} \\
\delta_{stb_{sym}} &= \delta_{stb} \\
\delta_{pla_{sym}} &= (\delta_{pla_l} + \delta_{pla_r})/2.0 \\
\delta_{ail_{dif}} &= (\delta_{ail_l} - \delta_{ail_r})/2.0 \\
\delta_{rud_{dif}} &= \delta_{rud} \\
\delta_{spl_{dif}} &= (\delta_{spl_l} - \delta_{spl_r})/3.0 \\
\delta_{pla_{dif}} &= (\delta_{pla_l} - \delta_{pla_l})/2.0
\end{aligned} \tag{6.3}$$

where the generic subscripts sym and dif are used to distinguish longitudinal and lateral-directional effectors. Note, this generic notation is used since some transports may have independent elevator, horizontal tails, or rudders. Furthermore, symmetrical spoiler and ailerons are not included since they produce negligible pitch authority. Finally, the three roll spoilers on each wing are ganged into left and right spoilers because there appears to be little benefit in commanding individual spoilers.

Since the aircraft systems management display contains information on the control surface positions, it is assumed that the control surface positions are measured. Both the Boeing 757 and the Airbus A320 have these displays. As a result, the control surface positions were accessed from the shared memory structures. All of the control effectors required to compute the defined deflections in (6.3) were obtainable except for the implemented throttle commands δ_{pla_l} and

δ_{pla_l}. When the throttles are repositioned in the cockpit, it takes time for the engine to spool up or down. Note, the throttle levers should be viewed as the commanded throttle position. To obtain the implemented throttle position, a surface model is required. Note, the implemented throttle position is a fictitious signal that is used to convey throttle saturation to the RCM. Taking the viewpoint that the engine is like all the other servos, a surface model was implemented using rate and position saturation limiters. Given that the engine control effectiveness is small, this first order approximation will not affect the reconfiguration performance. The rate limit was determined by measuring the rate of change of the Engine Pressure Ratio (EPR) which is an available measurement on existing transport aircraft. The position limits correspond to the position limits on the throttle console.

Table 6.6 summarizes the control vector $u(t)$ used for the ACFS implementation.

Table 6.6: Control Surface Positions $u(t)$ for ACFS Implementation

Variables	Description	Units
$\delta_{elv_{sym}}$	Symmetrical Elevator Deflection	*deg*
$\delta_{stb_{sym}}$	Symmetrical Horizontal Tail Deflection	*deg*
$\delta_{pla_{sym}}$	Symmetrical Throttle Deflection	*deg*
$\delta_{ail_{dif}}$	Differential Aileron Deflection	*deg*
$\delta_{rud_{dif}}$	Differential Rudder Deflection	*deg*
$\delta_{spl_{dif}}$	Differential Spoiler Deflection	*deg*
$\delta_{pla_{dif}}$	Differential Throttle Deflection	*deg*

Control Surface Saturation Level Vector $\Delta u(t)$

The final set of signals required are the control surface saturation levels. Recall from Section 2.1 and Equation (2.8) that the control surface saturation level is defined as the difference between the actual control surface deflection u and the unsaturated control surface deflection $u_k + u_{rcm}$:

$$\Delta u(t) = u - (u_k + u_{rcm})$$

As noted in Section 6.1, the aircraft simulator lacks servo dynamics; thus, the saturation level for each surface is computable via shared memory data, i.e. the unsaturated control surface deflection equals the control surface command in Section 4.5. Also, the saturation level of a failed surface is nulled as discussed in Section 4.5. Using the same control surface definitions, Table 6.7 summarizes the control surface saturation level $\Delta u(t)$ for the ACFS implementation.

Table 6.7: Control Surface Saturation Levels $\Delta u(t)$ for ACFS Implementation

Variables	Description	Units
$\Delta u_{elv_{sym}}$	Symmetrical Elevator Saturation Level	*deg*
$\Delta u_{stb_{sym}}$	Symmetrical Horizontal Tail Saturation Level	*deg*
$\Delta u_{pla_{sym}}$	Symmetrical Throttle Saturation Level	*deg*
$\Delta u_{ail_{dif}}$	Differential Aileron Saturation Level	*deg*
$\Delta u_{rud_{dif}}$	Differential Rudder Saturation Level	*deg*
$\Delta u_{spl_{dif}}$	Differential Spoiler Saturation Level	*deg*
$\Delta u_{pla_{dif}}$	Differential Throttle Saturation Level	*deg*

6.2.2 Reference Model Component Implementation

The reference model component provides the desired linear dynamics CA_m and CB_m, desired trim condition Cf_m, and an estimate of the control power dimensional derivatives CB_p to the adaptive augmentor component of the RCM. The required inputs included all of the signal conditioning and estimation outputs, as well as the flight condition parameters Mach and altitude. The identification problem is formulated in Chapter 3. As stated in this chapter, the generation of the required data has been separated into a two stage identification process which is well suited for on-line estimation. The first stage involves the identification of the aircraft's control effectiveness matrix CB_p, and the second stage involves the identification of the desired dynamics A_m and B_m. Recall that A_m and B_m versus CA_m and CB_m will be estimated so that the strict stability assumption of A_m can be enforced. Finally, the desired trim f_m will be derived using the estimated desired dynamics A_m and B_m so that a feasible trim solution is obtained.

The first stage model structure for estimating CB_p is presented in Section 3.1.1. The governing equation for this model structure is the non-linear, stability axes moment equations with linear

aerodynamics. The matrix-vector representation of this equations is presented in Equation 3.8 and is duplicated here:

$$\dot{\omega}_S + \Omega_R \omega_S + J_S^{-1} \Omega_S J_S \omega_S = CA_p(\rho)x + CB_p(\rho)u + Cf_p(\rho)$$

where the subscript S stands for stability axes, ρ represents the flight condition dependency, the linear aerodynamics $CA_p(\rho) \in \mathbb{R}^{m \times n}$ and $CB_p(\rho) \in \mathbb{R}^{m \times q}$, and the aerodynamic intercept $Cf_p(\rho) \in \mathbb{R}^{m \times 1}$. All other variables are explained in Section 3.1.1.

The unknown matrices in the above equation are CA_p, CB_p, and Cf_p. As stated in Section 3.1.1, the intercept Cf_p can be removed from the model structure by simply removing the mean from the data set. Given the state vector in Table 6.2 and the control surface deflections in Table 6.6, the aerodynamic matrices CA_p and CB_p have the following format:

$$CA_p(\rho) = \begin{bmatrix} M_{v_t} & M_\alpha & 0 & M_q & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & L_\beta & 0 & L_p & L_r \\ 0 & 0 & 0 & 0 & N_\beta & 0 & N_p & N_r \end{bmatrix} \tag{6.4}$$

$$CB_p(\rho) = \begin{bmatrix} M_{\delta_{elv_{sym}}} & M_{\delta_{stb_{sym}}} & M_{\delta_{pla_{sym}}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & L_{\delta_{ail_{dif}}} & L_{\delta_{rud_{dif}}} & L_{\delta_{spl_{dif}}} & L_{\delta_{pla_{dif}}} \\ 0 & 0 & 0 & N_{\delta_{ail_{dif}}} & N_{\delta_{rud_{dif}}} & N_{\delta_{spl_{dif}}} & N_{\delta_{pla_{dif}}} \end{bmatrix} \tag{6.5}$$

where ρ signifies that these dimensional derivatives are flight condition dependent and it is assumed that $z(t)$ is order as follows: $z(t) = [q(t)\ p(t)\ r(t)]^T$.

In order to obtained unbiased estimates of these dimensional derivatives, there must be adequate excitation for each variable in the state and control vectors. During normal operations, the nominal control system does not use the horizontal tail, symmetrical or differential throttles, or spoilers; thus, excitation must be injected into these surfaces to obtain a complete estimate of the control effectiveness matrix CB_p. This can be done during a flight test phase of implementation and updated during ferry or transport operations.

The second stage model structure for estimating the desired dynamics A_m and B_m is presented in Section 3.1.2. The governing equation for this model structure is the output performance error e_o(t). The matrix-vector representation of this equation is presented in Equation 3.12 and is duplicated here:

$$\dot{z}(t) - CB_p(\rho)\Delta u(t) = CA_m(\rho)x(t) + CB_m(\rho)r(t-\tau) + Cf_m(\rho)$$

Recall that since the control effectiveness matrix is estimated in the first stage, the $CB_p\Delta u$ vector can be treated as known and moved to the left side of the expression. Also, knowing that the entire desired dynamics are needed to satisfy the stability assumption, but only a subset of the control effectiveness matrix has been identified in the first stage, the formulation can be represented as follows:

$$\begin{aligned} \dot{z}(i;t) - CB_p(i,:;\rho)\Delta u(t) &= A_m(i,:;\rho)x(t) + B_m(i,:;\rho)r(t-\tau) + f_m(i;\rho) \\ \dot{x}(j;t) &= A_m(j,:;\rho)x(t) + B_m(j,:;\rho)r(t-\tau) + f_m(j;\rho) \end{aligned}$$

where the index i, which ranges from $1 : m$, represents the subset of states that are in the performance vector and the index j, which ranges from $m+1 : n$, represents the subset of states that are not contained in performance vector. Again, these indices reflect the fact that each row represents an independent regressor.

The unknown matrices in the above equation are the desired dynamics A_m and B_m and the desired trim f_m. As stated in Section 3.1.2 the desired trim f_m is computed instead of estimated; thus, it can be removed from the structure by removing the mean from the data set. Given the state vector in Table 6.2 and the pilot inputs in Table 6.5, the desired dynamics A_m and B_m

have the following format:

$$A_m(\rho) = \begin{bmatrix} X_{v_t} & X_\alpha & X_\theta & 0 & 0 & 0 & 0 & 0 \\ Z_{v_t} & Z_\alpha & Z_\theta & Z_q & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ M_{v_t} & M_\alpha & M_\theta & M_q & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & Y_\beta & Y_\phi & Y_p & Y_r \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & L_\beta & L_\phi & L_p & L_r \\ 0 & 0 & 0 & 0 & N_\beta & N_\phi & N_p & N_r \end{bmatrix} \tag{6.6}$$

$$B_m(\rho) = \begin{bmatrix} 0 & X_{dep} & 0 \\ 0 & Z_{dep} & 0 \\ 0 & 0 & 0 \\ 0 & M_{dep} & 0 \\ Y_{dap} & 0 & Y_{drp} \\ 0 & 0 & 0 \\ L_{dap} & 0 & L_{drp} \\ N_{dap} & 0 & N_{drp} \end{bmatrix} \tag{6.7}$$

where ρ signifies that these dimensional derivatives are flight condition dependent. Finally, the ACFS version used for this implementation does not contain an interface for the rudder pedals; as a result, rudder pedal inputs were injected via a timing sequence in the rudder code so that unbiased estimates of the rudder pedal effectiveness could be obtained.

Given the two structures above, both batch and sequential identification results were obtained. The primary purpose of the batch results were to tune the sequential algorithm. In both instances, the identification results were obtained off-line in Matlab. The on-line identification algorithm was simulated by feeding the algorithm new data at every sample time. The algorithm was not fully integrated into the ACFS because of the coding time that would be required.

The batch identification algorithm is presented in Section 3.2. The batch identification process is quite simple. Given a data record in which all pertinent variables are excited, select a weighting matrix, if needed, such that good identification results are obtained. For this implementation, a weighting matrix was only required for the longitudinal regressors in the second stage of the identification process. The ACFS does contain some non-linear coupling from the lateral-directional axes to the longitudinal axes, and this coupling was deemphasized using the weighting matrices to obtain good estimates.

With good batch estimates obtained, the sequential estimation algorithm was tuned. The sequential identification algorithm is presented in Section 3.3. The primary design parameters used to tune the sequential algorithm are the data window lengths, persistence of excitation thresholds, the maximum condition number of the re-parameterized regressor, the *a priori* data used for the sequential updating, and low-pass filtering of the estimates. The selection of each of these design parameters will be discussed in the following paragraphs.

There are a total of nine independent regressors, three regressors for the first stage and six regressors for the second stage, required to generate the required information. Note, there are only six second stage regressors since the pitch and bank angle kinematic equations do not contain unknown dimensional derivatives. Each of these regressors may have different data window lengths. As new data is sampled, the new data replaces the last row of the regressor, and subsequent data is shifted up by a row. Because of the model structure used for the first and second stages, the eleven individual regressors were reduced to four master regressors, longitudinal and lateral-directional for the first stage and likewise for the the second stage. The length of the master data windows correspond to the individual regressor that has the largest data window. For this implementation, both longitudinal data window lengths were chosen to be 12 seconds which corresponds to 360 samples. Likewise, both lateral-directional data window lengths were chosen to be 20 seconds or 600 samples. Recall that the ACFS runs at 30 Hz. Since the ID results are not being used in real-time for reconfiguration (because the ID results are stored), the inherent trade-off between estimation speed and identification accuracy is avoided, i.e. large data windows are acceptable. It is also worth noting that if the data window lengths are set to the size of the input file, the sequential ID results should duplicate the ID results obtained

from the minimum variance batch least-squares results. This provides a useful debugging tool.

The next important design parameters that are required are the persistence of excitation thresholds. These thresholds are used to turn the identification algorithm off when persistent excitation is not present. Recall that low excitation conditions result in biased estimates. In addition to turning the identification off during low excitation, the thresholds can be used to turn off the identification when the aircraft is not operating in a nominal condition. For example, the desired dynamics identification should be turned off when excitation is being injected into either the horizontal tail, spoilers, or throttle levers for the first stage identification. For the first stage of the identification, the rotation acceleration RMS levels were used to judge the excitation level. The thresholds were set to double the quiescent RMS levels. The quiescent RMS is based on the sensor noise and atmospheric disturbances. For the second stage, the stick and pedal RMS levels were used to turn the identification process on. Clearly, given adequate stick and pedal excitation, the aircraft states will be excited, and good estimates will be obtainable.

The next tuning parameter is the maximum condition number of the re-parameterized regressor. Singular value decomposition is performed on the regressor matrix to remove collinearities. Collinearity implies that two columns are linearly related or that a single column has low excitation. Recall that all of the regressor columns are zero mean; thus, a column with low excitation results in a column of zeros. The collinearity of a regressor can be quantified by its condition number. It is well known that if the condition number of the regressor matrix is high, biased estimates will result [20]. As illustrated in Section 3.3, singular value decomposition can be used to reduce the condition number. The lower the maximum condition number is, the greater the impact the re-parameterization has on the induced gain of the regressor matrix. For this implementation, a maximum condition number of 250 was used for the longitudinal regressors and 50 was used for the lateral-directional regressors.

It was assumed initially that no estimates were available, thus the *a priori* information consisted of null matrices and the *a priori* variances where set to 100, which means that this information is completely uncertain. As sequential estimates were generated, this estimate became the *a priori* data for the next update. The sequential minimum variance update algorithms operate

in the following fashion. If the new estimate had a lower covariance than the *a priori*, then the new estimate was favored in the sequential updating. Likewise, if the new estimate had a larger covariance than the *a priori* data, the new estimate was discarded. Finally, if the new estimate had the same covariance as the *a priori* and the estimate was different, the estimate is averaged and the covariance is reduced by half. Using the new estimates as *a priori* data generates rapid convergence which low covariances.

Finally, a discrete low pass filter was implemented to reduce the fluctuations in the estimation parameters. These filters were required since the aerodynamic model is noisy. This noise was attributed to poor interpolation routine used in the aerodynamic tables. The algorithm of discrete filter blended 95% of the *a priori* data with 5% of the new estimates. In affect, this algorithm added damping to the identification process.

6.2.3 Adaptive Augmentor Component Implementation

The adaptive augmentor component outputs perturbations signals to the control surfaces and throttles to perform the task. Reconfiguration control commands are $u_{rcm_c} = \left[\delta_{elv_{sym}}\ \delta_{stb_{sym}}\ \delta_{pla_{sym}}\ \delta_{ail_{dif}}\ \delta_{rud_{dif}}\ \delta_{spl_{dif}}\ \delta_{pla_{dif}}\right]^T$. These deflections are defined in Equation 6.3. The inputs for this component include following outputs from the signal conditioning and estimation component: the state vector $x(t)$ defined in Table 6.2, the performance vector derivative $\dot{z}(t)$ defined in Table 6.4, the pilot input $r(t)$ defined in Table 6.5, and the control surface saturation level $\Delta u(t)$ defined in Table 6.7. Additionally, the following reference model outputs are required: the desired linear dynamics CA_m and CB_m, desired trim condition Cf_m, and an estimate of the control power dimensional derivatives CB_p. Given these inputs, the input error $e_i(t)$ can be computed and the reconfiguration gains can be updated based on this computation.

In the following subsections, the core functionality of the adaptive augmentor will be presented for the ACFS implementation. First, the control allocation scheme chosen for this application will be explored. Then, the process of updating the control parameters will be discussed.

Reconfiguration Control Allocation

Given that $u_{rcm_c} \in \mathbb{R}^7$ and $z \in \mathbb{R}^3$, a control allocation scheme is required to ensure that CB_u^{-1} is square and exists (Assumption 9) for all possible single event control surface failures. The reconfiguration control surface allocation developed for this implementation is a basic approach that blends all of the control effectors in a given axis. Many elaborate control allocation techniques exist in the literature for over-determined systems. For this application, there is not an abundant supply of control effectors; thus, these approaches were not warranted.

The parameters that are important for reconfiguration are the achievable moments that a surface can produce and the rate at which the moment can be generated. The maximum achievable moment depends on the control power effectiveness — which is flight condition dependent — and the maximum deflection from the trim position. The rate of achieving the moment depends on the rate saturation limit of the actuator. These parameters affect aircraft reconfiguration in three ways. During post failure operations, the available maximum moment must be greater than the failure disturbance for a static equilibrium to exist. Secondly, even though the available moment to cancel the failure disturbance may exist, the moment must be generated fast enough to prevent the aircraft from entering an unrecoverable attitude. Finally, if the aircraft is unstable, there must be additional available control moment to stabilize the aircraft.

In the following paragraphs, typical control effectiveness levels, along with position and rate limits, for the control effectors in each axis will be compared. Based on this data, a control allocation logic will be devised which translates pseudo deflections to specific effector commands. Since this logic is only used in the RCM, it will not alter the basic control operation during nominal operation. Given this gearing scheme, the pseudo control power effectiveness will be derived for the initial estimate of CB_u^{-1}. Finally, the impact of the control allocation and estimates of the pseudo control effectiveness on the reconfiguration task will be explored.

Longitudinal Axis Reconfiguration Control Allocation The longitudinal control effectors include the elevator, horizontal tail, and symmetric throttles. The elevator is the primary pitch control effector in the longitudinal axis, and this surface can generate a large moment at

a fast rate. The horizontal tail is the primary longitudinal trim device, and it can generate the largest moment but at a slow rate. The engine thrust is a secondary pitch control effector, and the pitch authority depends on the engine disposition but generally the authority is small for wing and aft-fuselage mounted engines and the rate is low.

For the ACFS, the reconfiguration control allocation for the longitudinal axis is composed of the elevator, horizontal tail, and collective thrust. The collective thrust is realized in terms of collective power-lever-angle (PLA). The control power, position limits, and rate limits for each of the surfaces are illustrated in Table 6.8 for a cruise altitude of 35,000 feet and Mach 0.82. Again, the dimensional coefficients are flight condition dependent, but the relative magnitudes are approximately equivalent throughout the flight envelope. Note, the position limits on the throttle positions indicate that the reconfiguration module is limited to $\pm 10.0^o$ about the trim throttle setting; thus, the reconfiguration module has a low-authority control over the throttles. The reasoning is that the pilots will still need to control airspeed, and if full authority is granted to the throttles, independent speed control would be lost.

Table 6.8: Longitudinal Achievable Accelerations, 0.82 M, 35,000 ft

Control Effector	M_δ $(rad/s^2/^o)$	Position $(^o)$ Limit	Rate $(^o/sec)$ Limit
$\delta_{elv_{sym}}$	-0.0282	$-25.0 \rightarrow 25.0$	80.0
$\delta_{stb_{sym}}$	-0.0771	$-12.0 \rightarrow 4.0$	0.15
$\delta_{pla_{sym}}$	0.0004	$-10.0 \overset{trim}{\rightarrow} 10.0$	3.57

It is seen from this table that the achievable moments as well as the rates to achieve these moments vary greatly. The result of this variation is that the RCM will not be able to accommodate some hard-over failures. For example, a horizontal tail hard-over failure to -8.0^o will produce a moment that the elevator and engines cannot cancel; in other words, a static equilibrium does not exist. Likewise, it is apparent that the horizontal tail and engines cannot produce a moment fast enough to prevent the aircraft from entering an unrecoverable attitude for an elevator hard-over failure. These limitations will be explored in simulations.

The longitudinal reconfiguration control allocation scheme devised for the ACFS blends these

three effectors. A one degree pitch commands translates into a one degree elevator deflection, a one degree horizontal tail command, and a 0.4 degree collective PLA command:

$$1.0^o\ \delta_{dep} \Rightarrow \begin{array}{l} 1.0^o\ \delta_{elv_{sym}} \\ 1.0^o\ \delta_{stb_{sym}} \\ 0.4^o\ \delta_{pla_{sym}} \end{array}$$

The gearing of the throttles was chosen so that the elevator and throttle positions saturated together.

Using this strategy, the pseudo pitch effectiveness $M_{\delta_{dep}}$ needs to be constructed for the CB_u matrix. An equivalent M_δ can be calculated using the ganging strategy and control effectiveness from Table 6.8:

$$M_{\delta_{dep}} = M_{\delta_{elv_{sym}}} + M_{\delta_{stb_{sym}}} + M_{\delta_{pla_{sym}}} 10/25 \tag{6.8}$$

This estimate of the pseudo pitch effectiveness will be too large if there is a failure to either the elevator or the horizontal tail. The result of this overestimation is that the initial required reconfiguration command to attenuate the failure disturbance will be underestimated. Even though the estimate will eventually converge, this initial underestimation can result in departure, especially when the low-bandwidth horizontal tail is being used to attenuate an elevator failure. There are two basic things that can be done about the initial errors in CBu^{-1}: (1) underestimate the pseudo control derivative $M_{\delta_{dep}}$ and thus overestimate the required deflection, and (2) have a high adaptation gain on the $\hat{J}$ matrix. Both approaches were implemented in the ACFS. The initial estimate of the pseudo pitch effectiveness was equated to the horizontal tail pitch effectiveness thus overestimating the required control surface deflection:

$$M_{\delta_{dep}} = M_{\delta_{stb_{sym}}} \tag{6.9}$$

Additionally, the adaptation gain for $\hat{CB}_u$ was set high. Thus, if the elevator fails, the initial estimate of $\hat{CB}_u$ is close to the true value, and the horizontal tail captures the failure easily without a slow rise time or excessive overshoot. For a stabilizer fixed position failure, the initial estimated deflection required would be under-estimated by approximately $M_{\delta_{elv}}/M_{\delta_{stb}}$. But, given the high adaptation gain and high-bandwidth of the elevator, the failure disturbance should be captured.

Lateral Axis Reconfiguration Control Allocation The lateral axis control effectors include two ailerons and six roll spoilers. The redundancy in this axis exists because differential spoilers provide roll control at high dynamic pressure (aileron control effectiveness decreases as dynamic pressure increases and eventually reverses sign) and collective spoiler deflections function as speed brakes. Typical control effectiveness values, along with position and rate limits for the lateral axis are compared in Table 6.9 for a cruise altitude of 35,000 feet and Mach 0.82. As before, the dimensional coefficients are flight condition dependent; however in this case, the relative magnitudes do not scale to other flight points because of aeroelastic effects. Therefore, Table 6.9 is used for explanation purposes only.

Table 6.9: Lateral Achievable Accelerations, 0.82 M, 35,000 ft

Control Effector	L_δ $(rad/s^2/^o)$	Position $(^o)$ Limit	Rate $(^o/s)$ Limit
$\delta_{ail_{dif}}$	0.0297	$-20.0 \rightarrow 20.0$	80.0
$\delta_{spl_{dif}}$	0.0502	$-60.0 \rightarrow 0.0$	80.0

The lateral reconfiguration control allocation scheme blends these surfaces. A one degree pseudo roll command translates into a one degree differential aileron deflection and a three degree differential spoiler command. This gearing is based on equating the position limits. Recall the definitions of differential deflections in Equation 6.3.

$$1.0^o\ \delta_{dap} \Rightarrow \begin{matrix} 1.0^o\ \delta_{ail_{dif}} \\ 3.0^o\ \delta_{spl_{dif}} \end{matrix} \tag{6.10}$$

Based on this scheme, the pseudo roll and yaw control power derivatives for the pseudo roll command are:

$$\begin{aligned} L_{\delta_{dap}} &= L_{\delta_{ail_{dif}}} + L_{\delta_{spl_{dif}}} 3.0 \\ N_{\delta_{dap}} &= N_{\delta_{ail_{dif}}} + N_{\delta_{spl_{dif}}} 3.0 \end{aligned} \tag{6.11}$$

As in the longitudinal case, if one of the lateral control surfaces fails, the roll control effectiveness will be overestimated, and this translates into an underestimation of the pseudo deflection required to attenuate the failure disturbance. However, in contrast to the longitudinal case, this overestimation of the effectiveness is tolerable for any lateral surface failure for several reasons. First, a single surface failure will not have a large impact on the estimated effectiveness, and secondly, all of the lateral surfaces have equivalent bandwidths. Finally, the nominal control law has some inherent robustness to unexpected roll rates. The net effect is that when a lateral surface experiences a hard-over failure, both the nominal and reconfiguration module will capture the failure disturbance. As the estimate of the control effectiveness converges, the proper control surface deflection required to attenuate the failure disturbance will be commanded; this effect eventually nulls the contribution from the nominal controller. This interaction between the RCM and the nominal controller will be highlighted in the simulation results that follow.

Directional Axis Reconfiguration Control Allocation The directional control allocation problem is particularly interesting for commercial transports. In general, commercial transports have a single rudder — though some new aircraft have a split rudder with independent actuators — and no other control surface has equivalent control power. Other secondary yaw effectors include differential ailerons, spoilers, and throttles. Table 6.10 compares the typical control power effectiveness and the position and rate limits for the yaw axis at a cruise condition of

35,000 feet and Mach 0.82.

Table 6.10: Directional Axis Achievable Accelerations, 0.82 M, 35,000 ft

Control Effector	N_δ $(rad/s^2/^o)$	Position $(^o)$ Limit	Rate $(^o/sec)$ Limit
$\delta_{rud_{dif}}$	-0.0130	$-6.0 \rightarrow 6.0$	80.0
$\delta_{ail_{dif}}$	-0.0013	$-20.0 \rightarrow 20.0$	80.0
$\delta_{spl_{dif}}$	0.0047	$-60.0 \rightarrow 0.0$	80.0
$\delta_{pla_{dif}}$	0.0016	$-10.0 \overset{trim}{\rightarrow} 10.0$	3.57

A few notes on this table. The dimensional derivatives are flight condition dependent, but the relative magnitudes of the rudder versus the rest are scalable throughout the flight envelope, because the rudder position limit is limited by empennage load considerations. Also, as in the longitudinal axis control allocation, the throttle position limits are artificially imposed to allow independent speed control. Finally, the spoilers are proverse yaw devices while the ailerons are adverse yaw devices; this trait will be utilized later.

Based on the directional achievable accelerations, a commercial transport is not well balanced from a reconfiguration perspective. The implications of this are twofold. For an aircraft with a single rudder, there is no other control surface that can cancel the yaw generated by a rudder hard-over failure. On the flip side, since the directional control powers of all the other surfaces are an order of magnitude less than the rudder, lateral reconfiguration is trivial for non-rudder failures. Thus, the only interesting directional failure is a failure of the rudder.

Even though some rudder fixed failure disturbance cannot be canceled by the other control surfaces, all is not lost; commercial aircraft generally have sizeable inherent static stability in the directional axis, which is realized in the C_{n_β} aerodynamic derivative. Due to this weathercocking ability, the rudder yawing moment will be canceled by flying with a non-zero β. This steady state β will then cause a rolling moment via the dihedral effect, which is realized in the C_{L_β} aerodynamic derivative. This induced rolling moment does present a possibility of departure however. But recall from the lateral axis discussion above, there is an abundant amount of lateral control power.

Under the current architecture of the reconfiguration module, the desired trim for the lateral-directional axis is wings level and zero side slip angle. Thus, for fixed position failures, the secondary yaw devices will attempt to drive the aircraft to zero angle of sideslip. In addition to driving the steady state β to zero, the secondary yaw devices must assume the responsibility of the yaw damper. For moderate fixed position failure, the static requirement can easily drive the secondary yaw devices to position saturation since there is an order of magnitude difference in the control authority. When the secondary surfaces saturate, a significant amount of the lateral control authority is nulled.

It would be advantageous to devise a method where the secondary yaw devices just provide yaw damping without trying to zero the sideslip angle. Thus, the inherent weathercocking ability of the aircraft would be used to capture the failure disturbance in yaw. A method for implementing this approach in the adaptive control architecture is unknown at this time. Attempts were made to redefine the desired trim Cf_m using the linear nominal model for a given fixed position rudder failure. This attempt failed because the nominal model does not approximate the crabbed flight condition well.

Referring back to Tables 6.10 and 6.9, ailerons or spoilers used as independent yaw devices will produce unacceptable lateral coupling. In effect, the reconfiguration module would have to generate a lateral reconfiguration command to cancel the unwanted coupling, and this would potentially result in a reduction of yaw acceleration. The net result is that a lot of control authority will be wasted.

To eliminate this potential 'force fighting', a split flap is devised using the spoilers and ailerons. A split flap is achieved so that the combined deflections of the spoilers and ailerons produce nearly zero rolling moment. Due to the fact that the spoilers are proverse yaw devices and the ailerons are adverse yaw devices, the yaw effectiveness terms are additive. An estimate of the spoiler deflection required to produce a zero rolling moment for a given aileron deflection is as follows:

$$\delta_{spl} = -\frac{L_{\delta_{ail_{dif}}}}{L_{\delta_{spl_{dif}}}} \delta_{ail_{spl}} \tag{6.12}$$

Using this gearing, the control effectiveness of the split flap is as follows:

$$\begin{aligned} L_{\delta_{spf}} &\approx 0.0 \\ N_{\delta_{spf}} &= N_{\delta_{ail_{dif}}} - N_{\delta_{spl_{dif}}} \frac{L_{\delta_{ail_{dif}}}}{L_{\delta_{spl_{dif}}}} \end{aligned} \tag{6.13}$$

Table 6.10 is repeated below for the cruise flight condition with split flap data inserted. The split flap position limits correspond to the aileron position limits. This is artificially enforced because if this limit is exceeded, the split flap will generate a significant roll acceleration that must be canceled by the opposite spoilers. Thus, spoilers will be deflecting on both wings which will not produce any additional yawing moment.

Table 6.11: Directional Axis Achievable Accelerations, 0.82 M, 35,000 ft (Repeated)

Control Effector	N_δ $(rad/s^2/^o)$	Position $(^o)$ Limit	Rate $(^o/sec)$ Limit
$\delta_{rud_{dif}}$	-0.0130	$-6.0 \rightarrow 6.0$	80.0
$\delta_{spf_{dif}}$	-0.0035	$-20.0 \rightarrow 20.0$	80.0
$\delta_{pla_{dif}}$	0.0017	$-10.0 \overset{trim}{\rightarrow} 10.0$	3.57

Using the numbers in Table 6.11, a static analysis was performed which concluded that the secondary yaw effectors can accommodate a failure disturbance from a rudder fixed failure of approximately:

$$|\delta_{rud_{failed}}| < 5.4^o$$

which is slightly less than the rudder's maximum deflection. However, this static analysis neglects the required control power needed to stabilize the Dutch Roll mode, which is certainly unstable for commercial transports at this flight point.

The yaw axis reconfiguration control allocation blends the control effectors in Table 6.11. However, due to the rudder deflection limit dependence on dynamic pressure, the control allocation gearing is dependent on dynamic pressure. As in the lateral axis, the control effector are geared in such a way that they all position saturate uniformly. Thus, a one degree pseudo yaw command translates into a one degree split flap deflection, half degree differential throttle command, and a dynamic pressure dependent rudder command.

$$1.0^o\ \delta_{drp} \Rightarrow \begin{array}{l} \frac{\delta_{rud_{max}}(\bar{q})}{20.0^o}\delta_{rud} \\ 1.0^o\ \delta_{spf} \\ -0.5^o\ \delta_{pla_{dif}} \end{array} \tag{6.14}$$

Based on this control allocation scheme, the pseudo yaw command position limits are $\pm 20.0^o$. Also, a negative sign on the differential throttles accounts for the fact that the rudder and split flap are adverse yaw devices where the differential throttle is a proverse yaw device.

The pseudo roll and yaw control power derivatives for the pseudo yaw command are:

$$\begin{aligned} L_{\delta_{drp}} &= L_{rud}\frac{\delta_{rud_{max}}(\bar{q})}{20.0^o} + L_{\delta_{spf}} - L_{\delta_{pla_{dif}}}0.5 \\ N_{\delta_{drp}} &= N_{rud}\frac{\delta_{rud_{max}}(\bar{q})}{20.0^o} + N_{\delta_{spf}} - N_{\delta_{pla_{dif}}}0.5 \end{aligned} \tag{6.15}$$

For a rudder failure, the yaw control effectiveness will be over-estimated and the resulting pseudo yaw command will be under-estimated. But given the high bandwidth of these surfaces and the fact that the aircraft will not depart in yaw, there is adequate time for the estimates to converge. For other failures that generate a yaw disturbance, the estimate of the yaw control effectiveness will be close to the true value.

Integration of the Reconfiguration Parameters

The second core function of the adaptive augmentor component is the integration of the reconfiguration parameters. In order to generate the pseudo reconfiguration commands for the control allocation above, the reconfiguration parameters need to be updated to reflect the current status of the aircraft. The reconfiguration parameters include the estimates of the reconfiguration control gains — $\hat{K}_x$, $\hat{K}_r$, and $\hat{K}_f$ — and the estimate $\hat{J}$ of the inverse control effectiveness matrix $(CB_u)^{-1}$.

Recall that the pseudo reconfiguration command, which is defined in Equation 2.14, is as follows:

$$u_{rcm_c}(t) = \hat{K}_x(t)\, x(t) + \hat{K}_r(t)\, r(t-\tau) + \hat{K}_f(t) \tag{6.16}$$

where $\hat{K}_x \in \mathbb{R}^{m\times n}$, $\hat{K}_r \in \mathbb{R}^{m\times m}$ and $\hat{K}_f \in \mathbb{R}^m$.

The estimate of the inverse of the control effectiveness matrix is required to calculate the input error. Recall that the input error, which is defined in Equation 2.17, is as follows:

$$e_i = \hat{J}[\dot{z} - (CA_m x + CB_m r + Cf_m + CB_p \Delta u)] \tag{6.17}$$

which is just the scaled non-saturating output error.

For the ACFS implementation, $m = 3$ and $n = 8$. Thus, according to Equation (2.14), there are a total of 24 state feed back gains, 9 pilot feed-forward gains, and 3 feed-forward disturbance gains. According to Equation (2.17), there are 9 elements of the estimated inverse. Thus, there are a total of 45 reconfiguration parameters that need to be updated.

Recall the robust adaptation law, which is defined in Equation 2.20:

$$\dot{\Phi}(j,:) = \begin{cases} -e_i(j) Z^T \Gamma & \text{if } |e_i(j)| \geq V_o(j) \\ 0 & \text{else} \end{cases} \tag{6.18}$$

for $j = 1 : m$ and where $|V| \leq V_o$ is the upper bound on the regressor disturbance, $Z = [x, r, 1, \dot{z} -$

$(CA_m x + CB_m r + Cf_m + CB_p \Delta u)]^T \in \mathbb{R}^{n+m+1+m}$, $\Phi = [\tilde{K}_r \ \tilde{K}_x \ \tilde{K}_f \ \tilde{J} \ ] \in \mathbb{R}^{m \times (n+m+1+m)}$, and the adaptation gain Γ is symmetric positive definite. As noted in Section 4.5, $CB_p \Delta u(t)$ may be uncertain for given failures. If a control surface fails, its contribution to $CB_p \Delta u(t)$ should be zero. Feeding an erroneous $CB_p \Delta u(t)$, especially for a low bandwidth surface, into the output error regressor will result in poor reconfiguration results. Section 4.5 suggested two solutions. For this implementation, given that control surface positions are measured, the failed surface's saturation level was set to zero, i.e. it is assumed that the surface position is measured and therefore the source of the failure is known.

At this point, all one needs to know to implement this adaptation law is an initial estimate of the reconfiguration parameters $\Phi(t_o)$, an adaptation gain Γ, a value for the upper bound of the regressor disturbance V_o, and a discrete integration algorithm. Each of these will be discussed in the following paragraphs.

The initial estimates of the reconfiguration control gains are zero —$\hat{K}_x(t_o) = 0$, $\hat{K}_r(t_o) = 0$, and $\hat{K}_f(t_o) = 0$ — to reflect the nominal condition, and the initial estimate of $\hat{J}(t_o)$ is set to the inverse of the estimated pseudo control effectiveness matrix $\hat{CB}_u^{-1}$ defined in the control allocation section. An estimate of the pseudo control effectiveness matrix is as follows:

$$\hat{CB}_u = \begin{bmatrix} M_{\delta_{dep}} & 0 & 0 \\ 0 & L_{\delta_{dap}} & L_{\delta_{drp}} \\ 0 & N_{\delta_{dap}} & N_{\delta_{drp}} \end{bmatrix} \tag{6.19}$$

where $M_{\delta_{dep}}$ come from (6.9), $L_{\delta_{dap}}$ and $L_{\delta_{drp}}$ come from (6.11), and $N_{\delta_{dap}}$ and $N_{\delta_{drp}}$ come from (6.13) and (6.15). The inverse of the block diagonal $\hat{CB}_u$ is then used as the initial estimate of the $\hat{J}(t_o)$.

Next, the adaptation gain Γ is selected. In general, the adaptation gain Γ is a diagonal matrix with each diagonal element unique and greater than zero, i.e. $\Gamma(i,i) > 0 \ i = 1 : 15$. We will refer to the set of diagonal elements $\Gamma(i,i)$ as the adaptation law gains. The magnitudes of the adaptation laws gains affect the convergence characteristics of the reconfiguration parameters $\Phi(t)$. Ideally, the adaptation law gains should be as large as possible to ensure rapid convergence;

however, if the gains are too large, significant overshoot in the optimal values can produce chattering in the control surfaces. If the gains are too small, the convergence rise time can be too long and can result in the aircraft entering an unrecoverable attitude. Clearly some of the adaptation parameters $\Phi(t)$ are more important than others. As highlighted in the control allocation discussion, the failure disturbance reconfiguration control gains $\hat{K}_f(t)$ are extremely important to the reconfiguration task. In addition, the estimate of the inverse control effectiveness matrix $\hat{J}(t)$ is important, so the reconfiguration deflections are not under or over-estimated. Finally, the reconfiguration control gains that augment the dynamic stability — $\hat{K}_x(t)$ — are important. Varying the magnitudes of the adaptation law gains $\Gamma(i,i)$ based on the relative importance of different reconfiguration parameters $\Phi(t)$ can permit fast convergence of the important reconfiguration control gains while avoiding chattering in others.

As in all design trades, something must be lost by varying the relative magnitudes of the adaptation law gains $\Gamma(i,i)$. Given that the adaptation law gain $\Gamma(12,12)$ for the reconfiguration control gain $\hat{K}_f(t)$ is the highest, $\hat{K}_f(t)$ will fluctuate about the optimal value to accommodate for the lack of convergence of the other reconfiguration parameters $\hat{K}_x(t)$, $\hat{K}_r(t)$, and $\hat{J}(t)$. To compound this problem, the fluctuating $\hat{K}_f(t)$ reduces the input error, which then results in a slower convergence rate of $\hat{K}_x(t)$, $\hat{K}_r(t)$, and $\hat{J}(t)$.

Next, the upper bound of the regressor noise V_o needs to be selected. V_o defines the deadband for adaptation law in (2.20). Recall from Section 2.5 that the deadbands robustify the adaptation algorithm to destabilizing regressor noise that includes higher-order nonlinearities, atmospheric disturbances, sensor noise, and model reduction errors. Clearly, the upper bound of the regressor noise V_o is flight condition and atmospheric disturbance dependent. The proposed approach for determining this upper bound is to group the higher-order nonlinearities, sensor noise, and model reduction errors into the general category of modeling error. The other category is atmospheric disturbance.

During normal operations, the computed input error is the regressor noise; thus, a measure of the regressor noise $V(t)$ is available throughout the flight envelope. Also, a rough measurement of the atmospheric disturbance is available on most commercial transports. The atmospheric

disturbance can be computed by measuring the noise content in the rotational rates, or it can be estimated using a Light Detection And Ranging (LIDAR) system. Thus, the atmospheric disturbance contribution to the regressor noise $V(t)$ can be subtracted out of the computed input error, and the resultant modeling error can be stored. The upper bound of the regressor noise can then be obtained on-line by using the measured atmospheric turbulence and the stored modeling error.

The final topic of the reconfiguration parameter integration is the selection of a integration technique. The discrete integration technique used is second-order Adams-Bashford predictor which is given as below. This technique is also used to integrate the aircraft equations of motion.

$$\Phi(k) = \Phi(k-1) + \frac{1}{2}[\dot{\Phi}(k) + \dot{\Phi}(k-1)]\Delta t + [\dot{\Phi}(k) - \dot{\Phi}(k-1)]\Delta t \tag{6.20}$$

6.3 Simulation Results

This section presents the simulation results of the RCM architecture implemented in the ACFS. All simulation results that follow are for a single flight point, which corresponds to a cruise condition: 35,000 feet and M=0.82. The discussion begins with a presentation of the aircraft flying qualities specifications as they pertain the ACFS. The specifications will be used to compare post-failure performance to nominal performance. Then the results of the reference model identification will be presented. Finally, the bulk of this section is devoted to illustrating the RCM potential for accommodating fixed-position failures to control surfaces.

6.3.1 Aircraft Flying Qualities

Aircraft flying qualities are the standard way of judging an aircraft's performance. The Federal Airworthiness Regulations (FAR) state the standards for US commercial aircraft. FAR Part 25 is the primary document for jet commercial transports; the ACFS falls into this category. Given that FAR-25 gives sporadic guidance in the area of stability and control handling qualities, the Military Specification for the Flying Qualities of Piloted Airplanes, MIL-F-8785C, is the default

specifications.

The military specifications define different aircraft classes, flight phases, and flying qualities levels. Based on these definitions, the ACFS is a Class III aircraft, and as stated above, the evaluation is to be performed at cruise conditions which correspond to a Category B flight phase. The different levels of flying qualities are outlined in Table 6.12.

Table 6.12: Flying Qualities Levels: Flying-Qualities Specification

Level	Definition
1	Flying qualities clearly adequate for the mission flight phase.
2	Flying qualities adequate to accomplish the mission flight phase but some increase in pilot workload or degradation in mission effectiveness exists.
3	Flying qualities such that the airplane can be controlled safely, but pilot workload is excessive or mission effectiveness is inadequate or both.

To quantify the aircraft's flying qualities, MIL-8758C defines the flying qualities levels in terms of linear aircraft modes. The typical metrics used to quantify these modes are undamped natural frequencies and damping ratios or time constants. The requirements for dynamic longitudinal stability are illustrated in Table 6.13 where $N_\alpha = -Z_\alpha v_t/g$ is the load factor gradient and

Table 6.13: Longitudinal Stability Flying-Qualities Specifications: Class III, Category B

Mode	Metric	Limit	Level 1	Level 2	Level 3
Short Period	ζ_{sp} (-)	Min	0.30	0.20	0.15
		Max	2.00	2.00	-
	$\omega_{n_{sp}}$ (rad/sec)	Min	$\sqrt{0.085N_\alpha}$	$\sqrt{0.038N_\alpha}$	$\sqrt{0.038N_\alpha}$
		Max	$\sqrt{3.60N_\alpha}$	$\sqrt{10.0N_\alpha}$	-
Phugoid	ζ_p (-)	Min	0.04	0.0	$T_{2_p} \geq 55.0sec$

$T_{2_p} = -log_e 2/\zeta_p \omega_{n_p}$ is the Phugoid time to double. Note, Z_α is defined in Equation 6.6. The requirements for dynamic lateral-directional stability are illustrated in Table 6.14 where $T_{2_s} = log_e 2/(1/T_s)$ is the spiral time to double and T_s is the spiral mode time constant. Also, Class

Table 6.14: Lateral-Directional Stability Flying-Qualities Specifications: Class III, Category B

Mode	Metric	Limit	Level 1	Level 2	Level 3
Spiral	T_{2_s} (sec)	Min	20.0	8.0	4.0
Dutch Roll	ζ_d (-)	Min	0.08	0.02	0.02
	$\zeta_d\omega_{n_d}$ $(rad\,\mathrm{sec})$	Min	0.15	0.05	-
	ω_{n_d} $(rad\,\mathrm{sec})$	Min	0.4	0.4	0.04
Roll	T_r (sec)	Min	1.4	3.0	10.0

III aircraft may be exempt from the minimum ω_{n_d} requirements.

6.3.2 Reference Model Identification

For the test flight condition, the reference model was estimated as outlined in Section 6.2.2. The reference model and pertinent performance issues will be presented below. For this identification, atmospheric disturbances were included. As noted in the implementation section, the reference model was generated off-line in Matlab using a data record. The on-line identification was simulated by feeding the algorithm new data at each new sample time. The data record included a pitch stick doublet followed by a roll stick and yaw pedal doublets. After these doublets, excitation signals were injected into the spoilers, throttle levers, and the horizontal tail so that the control effectiveness CB_p could be estimated. The time record used for the identification was 180 seconds in length (2400 samples) and the complete record was fed to the sequential identification algorithm while first and second stage identification was performed in parallel.

The ordering of the states, pilot inputs and the control surface deflections are defined in Section

6.2.1. The control effectiveness matrix CB_p was identified in the first stage and is as follows:

$$CB_p = \begin{bmatrix} -0.0282 & -0.0771 & 0.0004 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0297 & 0.0052 & 0.0502 & -0.0000 \\ 0 & 0 & 0 & -0.0013 & -0.0130 & 0.0047 & 0.0016 \end{bmatrix} \tag{6.21}$$

The estimated desired dynamics A_m and B_m were identified in the second stage and are as follows:

$$A_m = \begin{bmatrix} -0.0205 & 0.1000 & -31.5395 & 0.0581 & 0 & 0 & 0 & 0 \\ -0.0002 & -0.8626 & -0.0022 & 1.0111 & 0 & 0 & 0 & 0 \\ 0.0000 & 0.0107 & -0.0208 & 0.9932 & 0 & 0 & 0 & 0 \\ 0.0004 & -1.4115 & 0.0528 & -1.4444 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.1282 & 0.0400 & -0.0024 & -0.9882 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & 0 & -3.6475 & 0 & -2.1222 & 0.8192 \\ 0 & 0 & 0 & 0 & 3.2333 & 0 & -0.1037 & -1.0003 \end{bmatrix} \tag{6.22}$$

$$B_m = \begin{bmatrix} 0 & 0.0060 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0 \\ 0 & 0.0085 & 0 \\ 0.0000 & 0 & -0.0070 \\ 0 & 0 & 0 \\ 0.0126 & 0 & -0.3316 \\ -0.0010 & 0 & 0.4043 \end{bmatrix} \tag{6.23}$$

Based on these estimates, the desired trim f_m was determined such that the desired trim of wings level, zero sideslip, and zero accelerations was satisfied:

$$f_m = \begin{bmatrix} 17.1477 \\ 0.2194 \\ -0.0010 \\ -0.3202 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.24}$$

Figures 6-3 and 6-4 illustrate the linear model response when compared to the nonlinear simulation. For this simulation, the pilot commands from the nonlinear simulator were fed to the Simulink model of the linear identification model. Note, the linear simulation does not contain a turbulence model.

It is seen from these figures that longitudinal response does match fairly well. The Phugoid mode can be observed in the linear response because the true versus equivalent pilot inputs are used in the linear simulation. Recall from Section 6.2.1 that the equivalent stick inputs mimic the autopilot attitude hold mode. The only deficiency in the identification model is the roll response for rudder pedal inputs, which corresponds to the time interval from 50 seconds to 80 seconds in Figure 6-4. Investigation of the cause of this deficiency revealed that the roll stability derivatives fluctuate when rudder pedals are present, i.e. the roll dimensional derivatives change when the roll stick and rudder pedals are deflected. Given this anomaly and the fact that good roll response was not possible for both lateral stick and rudder pedals, the lateral stick inputs were emphasized and the rudder pedal inputs were deemphasized in the identification algorithm.

Based on this linear reference model, the nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-5. Also, the nominal aircraft flying qualities were assessed. Table 6.15 presents the flying qualities for the nominal aircraft. It is seen that all of the modes have Level 1 flying qualities. The nominal frequency response

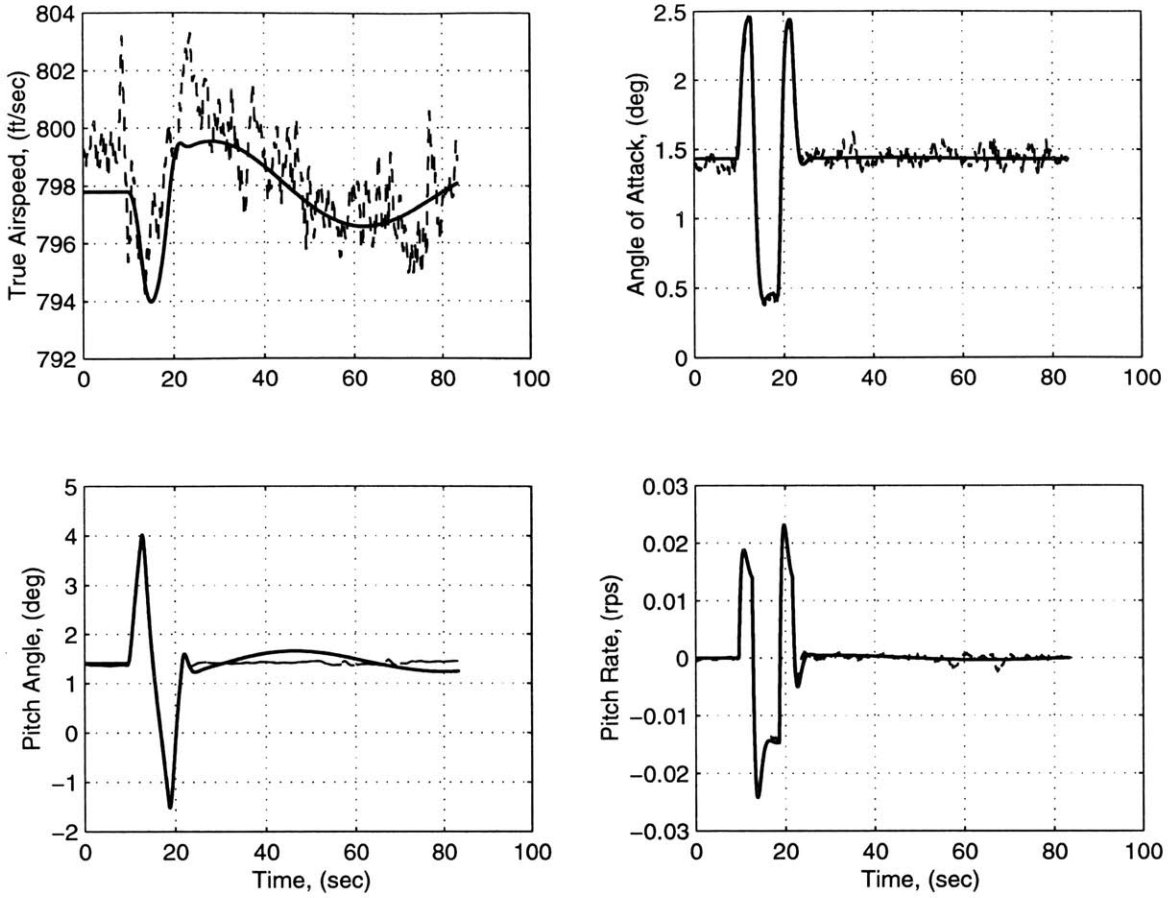

Figure 6-3: Longitudinal Response of Linear Model and Nonlinear Simulator, 0.82 M, 35,000 ft (— Linear - - Nonlinear)

and aircraft flying qualities will be used later in this chapter to compare the post-failure with RCM to nominal performance. In doing this comparison, it is important to note that these comparisons will be based on identified post-failure and nominal models. The linear models will be generated using the same pilot inputs that produce the responses in Figures 6-3 and 6-4. It is seen in these figures that low frequency modes, which include the Phugoid and Spiral modes, are not excited; thus, the identification process cannot generate reliable estimates of these modes. As a result, the comparison of the Phugoid damping ζ_p and the the Spiral time constant T_{2_s} should not be viewed as reliable data; equivalently, the frequency response below $\omega < 0.1 rad/sec$ is not reliable.

Finally, the objective of the system identification is to minimize the output performance error

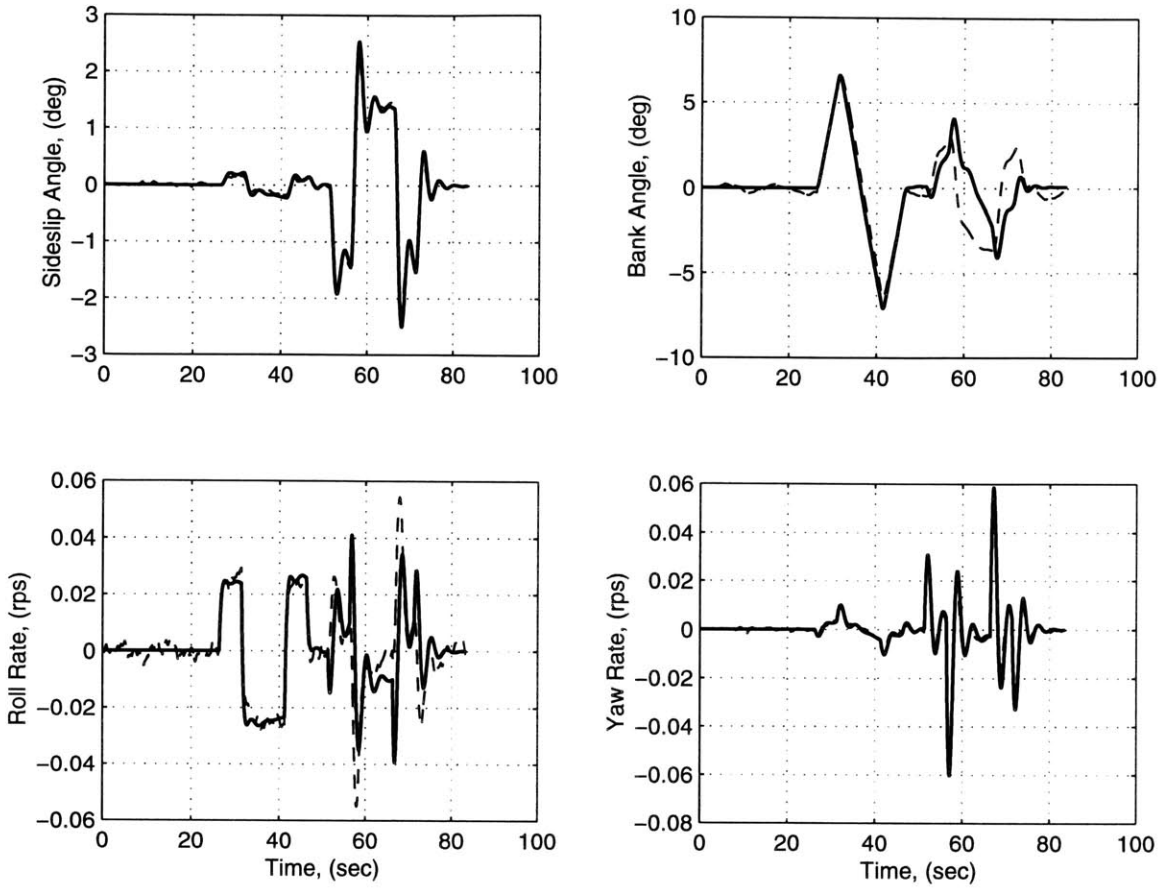

Figure 6-4: Lateral-Directional Response of Linear Model and Nonlinear Simulator, 0.82 M, 35,000 ft (— Linear - - Nonlinear)

$e_o(t)$ which is a scaled version of the input error $e_i(t)$. Recall that the input performance error $e_i(t)$ is the primary metric for updating the reconfiguration gains and is also the key measure for determining whether a failure has occurred. In order to use the input error as a measure for failure detection, the input error or equivalently, the output error needs to be near zero for nominal operations. As presented in Section 2.4, the input error is a noisy signal due to the higher-order nonlinearities, atmospheric disturbances, sensor noise, and model reduction errors. Due to robustness issues associated with the adaptation algorithm, this noise limits the achievable performance of the RCM; recall the discussion in Section 2.4 about the upper bound on the regressor disturbance V_o.

Figure 6-6 presents the nominal output performance error for the generated reference model.

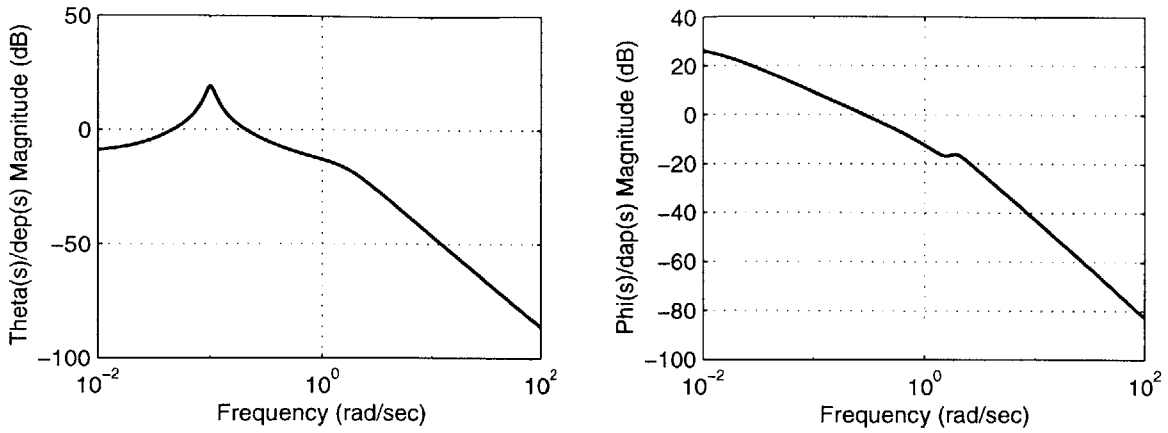

Figure 6-5: Nominal Frequency Response, 0.82 M, 35,000 ft

Table 6.15: Nominal Aircraft Flying-Qualities: Class III, Category B

Mode	Metric	Value	Level
Short	ζ_{sp}	0.71	1
Period	$\omega_{n_{sp}}$	1.63 rad/sec	1
Phugoid	ζ_p	0.11	1
Spiral	T_{2_s}	Stable	1
Dutch Roll	ζ_d	0.28	1
	$\zeta_d\omega_{n_d}$	0.52 rad/sec	1
	ω_{n_d}	1.84 rad/sec	1
Roll	T_r	0.46 sec	1

The scales in these three plots are chosen based on the maximum acceleration magnitudes experienced due to the pilot inputs. Based on these scales, it can be deduced from the figures that the output error noise is about an order of magnitude less than the maximum accelerations. Furthermore, in the pitch acceleration output error, at about 20 seconds, a bias appears which is attributed to the attitude hold mode turning on. This is a nonlinear effect that is grouped into modeling error. Finally, the roll acceleration output error has a blip between 50 and 80 seconds due to the anomaly of the rudder pedal inputs, but given that the ACFS does not have a rudder pedal interface, this modeling error should not have a major effect on the reconfiguration results that follow.

As promised in Section 6.2.1, the impact of using state derivative estimates $\hat{\dot{x}}(t)$ in the context

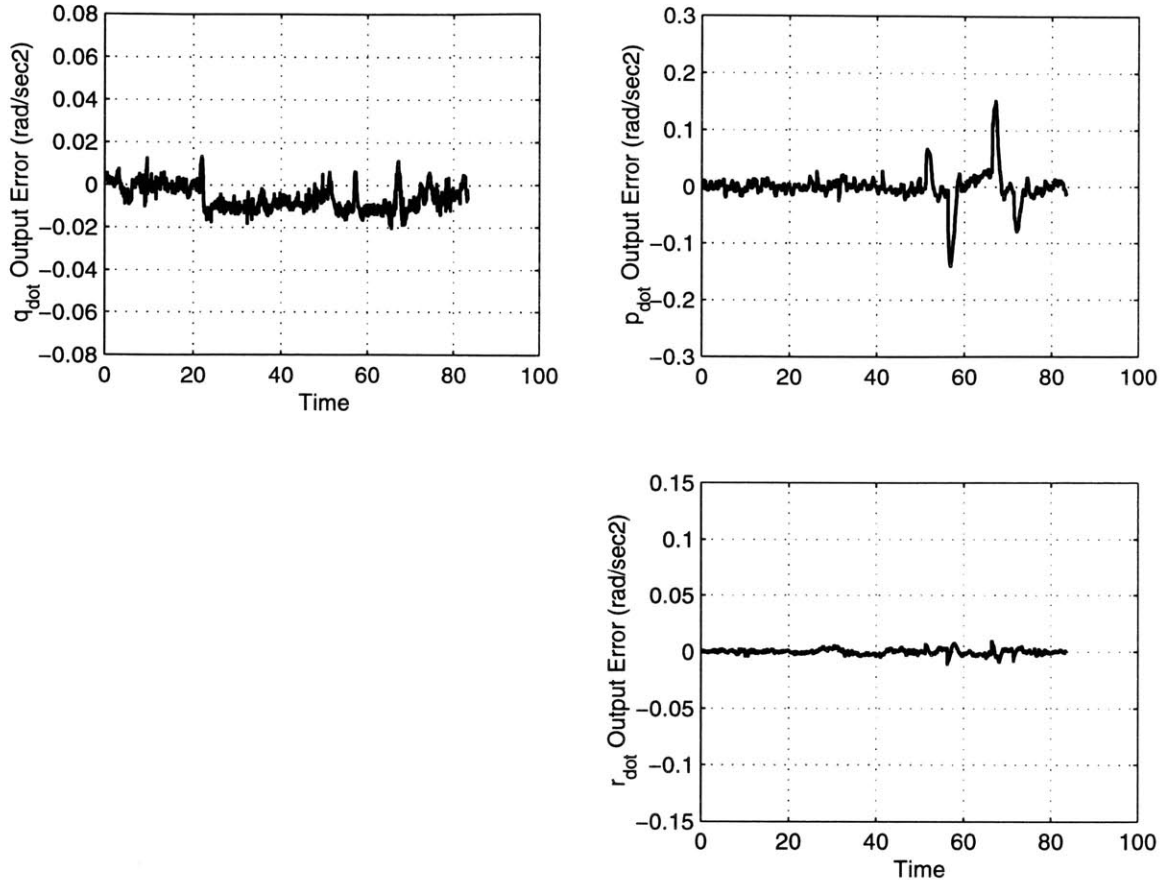

Figure 6-6: Output Performance Error $e_o(t)$ for Nominal Aircraft, 0.82 M, 35,000 ft

of system identification and computation of the output performance error $e_o(t)$ is explored in this section. Using the same procedures as above, the Kalman filters were simulated off-line in Matlab by feeding the recursive algorithm new data at each new sample time. State derivative estimates $\hat{\dot{x}}(t)$ where obtained for the same data used in the above results. These estimates were then used to generate the reference model using the same procedures as previously noted. It will be shown next that the results of these simulations demonstrated that the state derivative estimates were capable of generating a reference model suitable for the reconfiguration task.

Figures 6-7 and 6-8 illustrate the response of the linear model generated with $\hat{\dot{x}}(t)$ data when compared to the nonlinear simulation. Again, for this simulation, the pilot commands from the nonlinear simulator were fed to the Simulink model of the linear identification model, and the linear simulation does not contain a turbulence model.

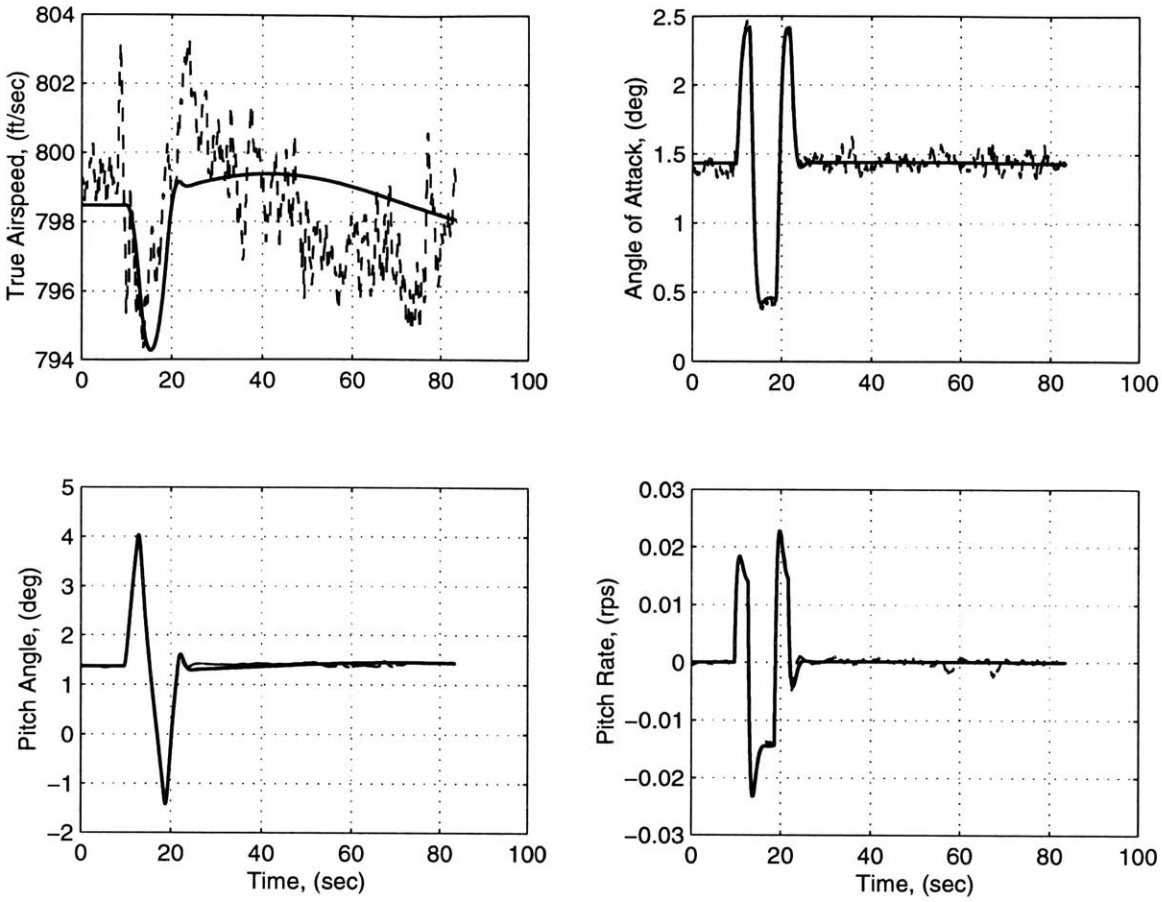

Figure 6-7: Longitudinal Response of Linear Model ($\hat{\dot{x}}(t)$ Data) and Nonlinear Simulator, 0.82 M, 35,000 ft (— Linear - - Nonlinear)

It is seen from these figures that the response of the linear model generated with $\hat{\dot{x}}(t)$ data resembles the linear model response illustrated in Figures 6-3 and 6-4. Based on the new linear reference model, the nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-9.

Also, the nominal aircraft flying qualities were assessed. Table 6.16 presents the flying qualities for the nominal aircraft. It is seen that all of the modes have Level 1 flying qualities. Both the frequency response functions and the flying qualities for the new reference model are close to the former results.

Finally, the impact of using the $\hat{\dot{x}}(t)$ on the ultimate reconfiguration performance can be viewed

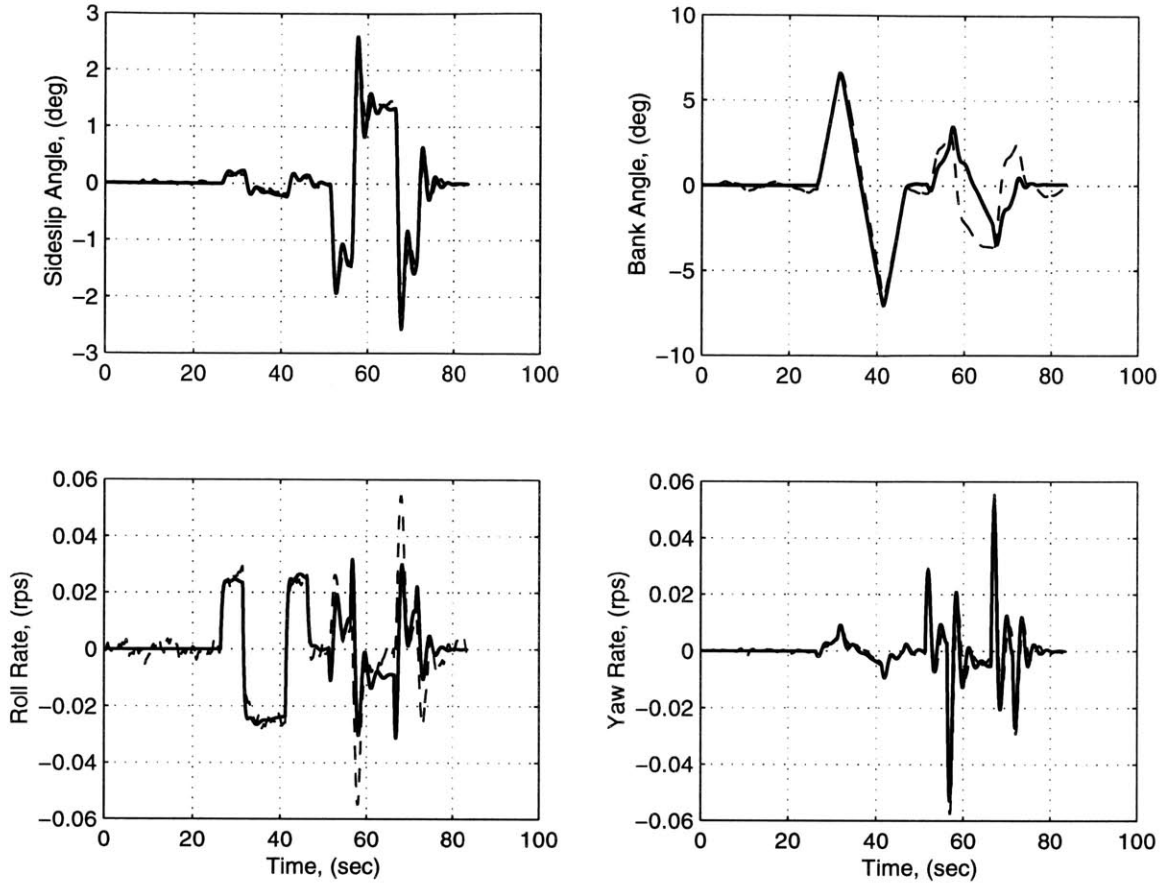

Figure 6-8: Lateral-Directional Response of Linear Model ($\hat{\dot{x}}(t)$ Data) and Nonlinear Simulator, 0.82 M, 35,000 ft (— Linear - - Nonlinear)

in the increased noise content of the output performance error $e_o(t)$. Figure 6-10 presents the nominal output performance error for the reference model generated using $\hat{\dot{x}}(t)$ data. Again, the scales in these three plots are chosen based on the maximum acceleration magnitudes experienced due to the pilot inputs. It is seen from these figures that the output error noise is about an order of magnitude less than the maximum accelerations. Comparing this figure to Figure 6-6, it is seen that Figure 6-10 has similar noise characteristics to the previous results. In the pitch acceleration output error, at about 20 seconds, a bias appears which is attributed to the attitude hold mode turning on. This is a nonlinear effect that is grouped into modeling error. The roll acceleration output error becomes large between 50 and 80 seconds due to the anomaly of the rudder pedal inputs, with a similar magnitude as in Figure 6-6. However, the yaw acceleration output error exhibits larger errors in this region, but as noted before, given that the ACFS

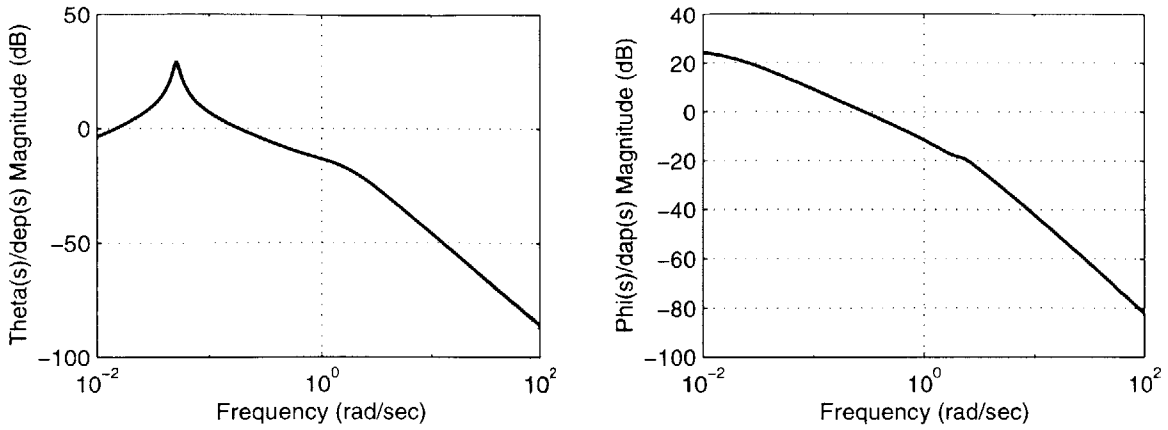

Figure 6-9: Nominal Frequency Response ($\hat{\dot{x}}(t)$ Data), 0.82 M, 35,000 ft

Table 6.16: Nominal Aircraft Flying-Qualities ($\hat{\dot{x}}(t)$ Data): Class III, Category B

Mode	Metric	Value	Level
Short	ζ_{sp}	0.73	1
Period	$\omega_{n_{sp}}$	1.78 rad/sec	1
Phugoid	ζ_p	0.06	1
Spiral	T_{2_s}	Stable	1
Dutch Roll	ζ_d	0.25	1
	$\zeta_d\omega_{n_d}$	0.55 rad/sec	1
	ω_{n_d}	2.21 rad/sec	1
Roll	T_r	0.40 sec	1

does not have a rudder pedal interface, this modeling error should not have a major effect on the reconfiguration results. Thus, the estimate of the state derivative $\hat{\dot{x}}(t)$ is suitable for the reconfiguration task.

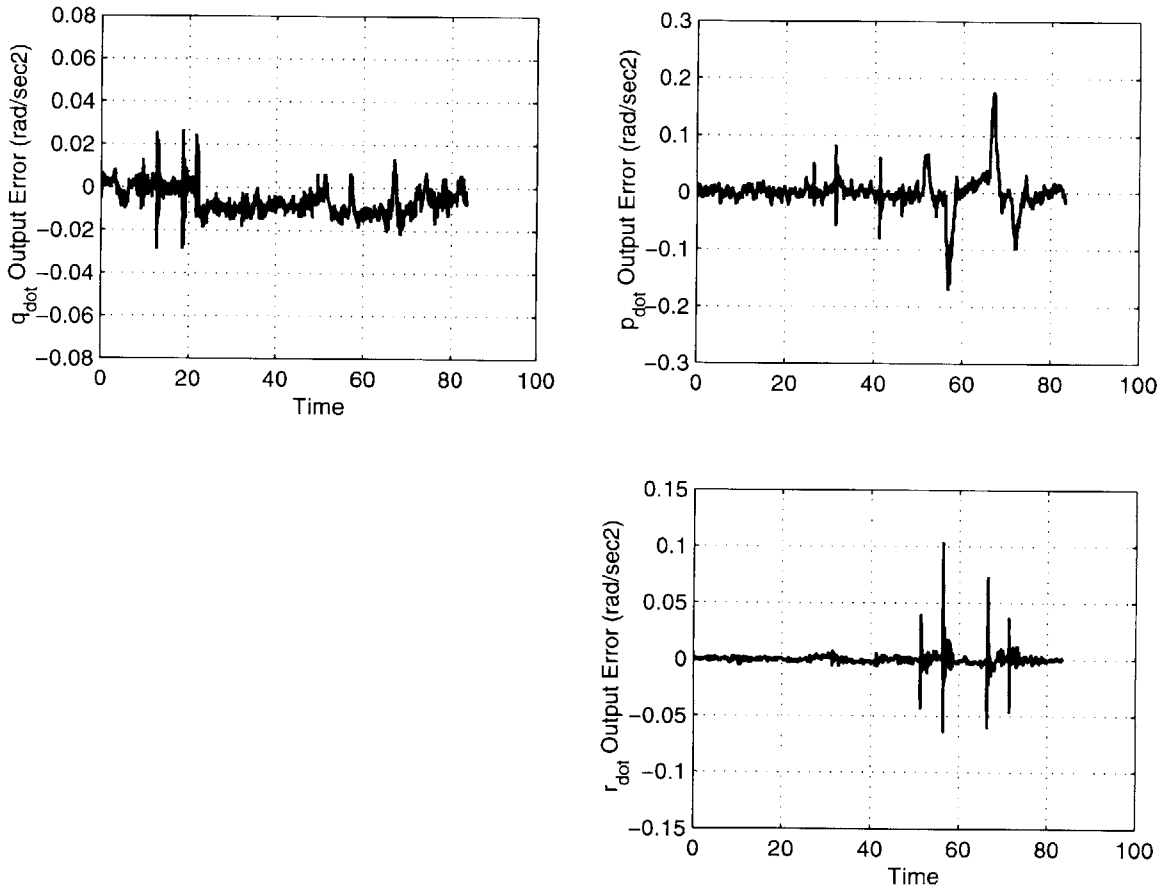

Figure 6-10: Output Performance Error $e_o(t)$ for Nominal Aircraft ($\hat{\dot{x}}(t)$ Data), 0.82 M, 35,000 ft

6.3.3 Longitudinal Failures

In this subsection, three longitudinal failure scenarios will be presented: horizontal tail fixed position failure, elevator fixed position failure, and an empennage hydraulic failure. Simulations of each of these failure scenarios will be shown at the selected cruise condition, and a comparison between the nominal and post-failure performance, with and without reconfiguration, will be presented. Additionally, a comparison of pitch stick to pitch angle and roll stick to bank angle frequency response functions will be presented. The frequency response functions are generated using the identification procedures outlined in the previous sections. Finally, the post-failure aircraft flying qualities will be compared to the nominal aircraft flying qualities.

Atmospheric disturbances are included in these simulations. Given the nominal output performance errors illustrated in Figure 6-6, the regressor disturbance upperbound for the robust adaptation algorithm, in Equation (2.20), was set to $V_o = \hat{J}(t)[0.01 \quad 0.01 \quad 0.002]^T$. Finally, the failure-logic switch is closed and remains closed, i.e. a failure is declared and adaptation is turned on, when the output performance error $e_o(t)$ exceeds $[0.05 \quad 0.03 \quad 0.03]^T \; rad/sec^2$. These values were selected through inspection of Figure 6-6 and noting that these bounds are not exceed for pitch and roll stick inputs. Recall from the previous section that the rudder pedal anomaly causes errors exceeding these bounds. However, since the ACFS does not have a rudder pedal interface for pilot inputs, no false alarm are expected.

Horizontal Tail Fixed Position Failure

The first failure scenario is a horizontal tail fixed position failure. Based on static analysis at the given flight condition, the elevator can cancel horizontal tail fixed failures that range from -9.1^o to 4.0^o, which is the upper limit. As a worst case scenario, the horizontal tail is moved instantaneously from the trim setting to the failed position. Because of this instantaneously failure disturbance and the time required to integrate the reconfiguration gains, it was determined by simulations that the aircraft was able to recover from -7.0^o to 4.0^o fixed failures. For failures that exceed this range, the aircraft stalls. Stalling the aircraft is not necessarily fatal; however,

this region clearly violates the model formulation, and adaptation characteristics are uncertain.

Simulation results for a horizontal tail fixed failure with and without the RCM are illustrated in Figures 6-11 and 6-12. At 5.0 seconds into the simulation, the horizontal tail is instantaneously

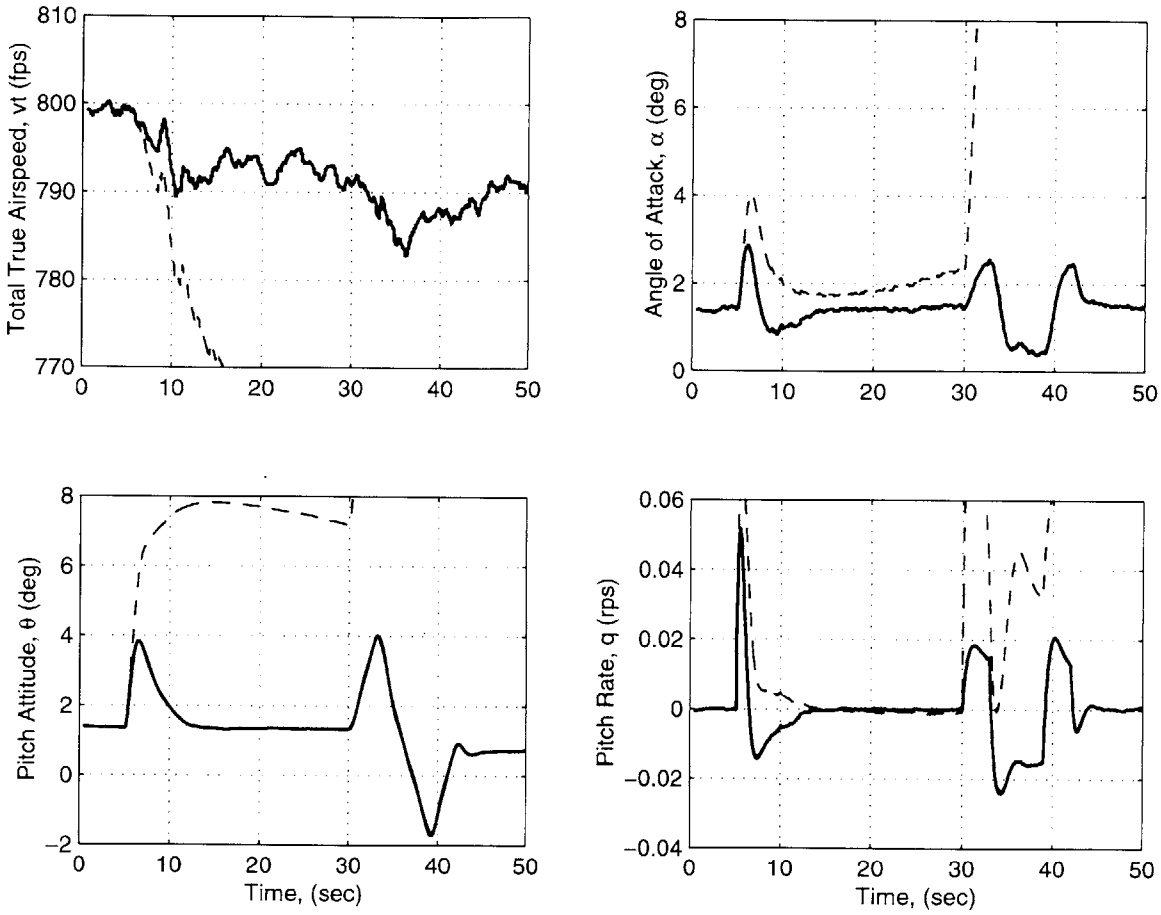

Figure 6-11: Horizontal Tail Failure, $\delta_{stb} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o RCM)

moved from the trim setting of -1.12^o to -5.0^o, and at approximately 30.0 seconds into the simulation, a pitch doublet is performed. It is seen from this Figure 6-11 that without reconfiguration, the aircraft departs. It is seen from Figure 6-12 that initially, the nominal control system integrates up the elevator deflection to zero the pitch rate. However, when a stick input is introduced to pitch the aircraft down, the aircraft pitches up and departs. This departure is caused by a discontinuity in the nominal control systems elevator command. The pitch stick command is not added as a perturbation to the elevator deflection required to zero the pitch

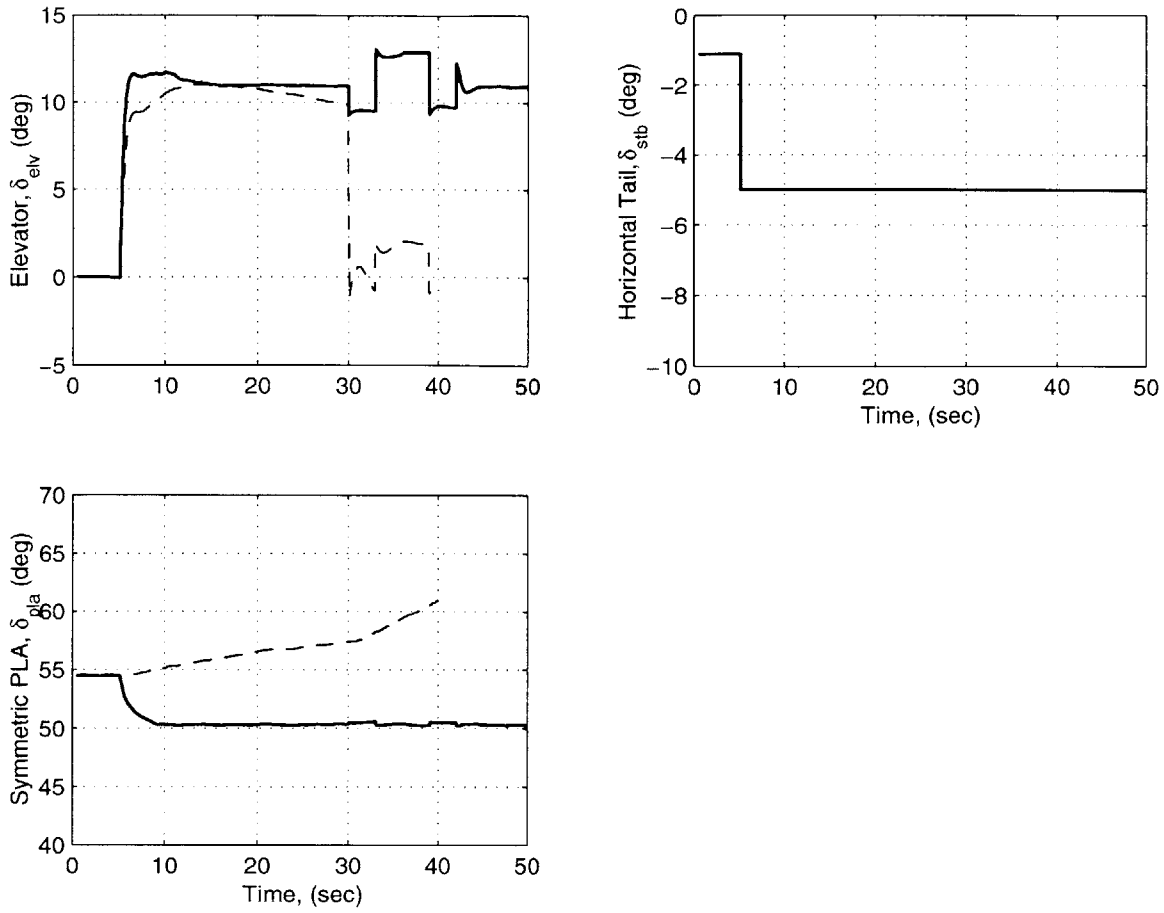

Figure 6-12: Surface Deflections for Horizontal Tail Failure, $\delta_{stb} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o RCM)

rate. Instead, as soon as the stick is moved, the pitch rate error integrator is zeroed.

With reconfiguration, the failure disturbance is captured through a combination of the nominal control system and the adaptive augmentor. Since both the adaptive augmentor and the nominal controller are working to cancel the disturbance, the pitch angle excursion is reduced by half. Additionally, the adaptive augmentor forces the aircraft down to the desired trim attitude, conveyed to the adaptation algorithm via Cf_m, without pilot inputs.

After the aircraft returns to its initial attitude, open-loop pilot commands are given to provide a comparison between post-failure performance and nominal performance. Figures 6-13 and 6-14 illustrate the comparison. The only significant difference between the nominal and post-

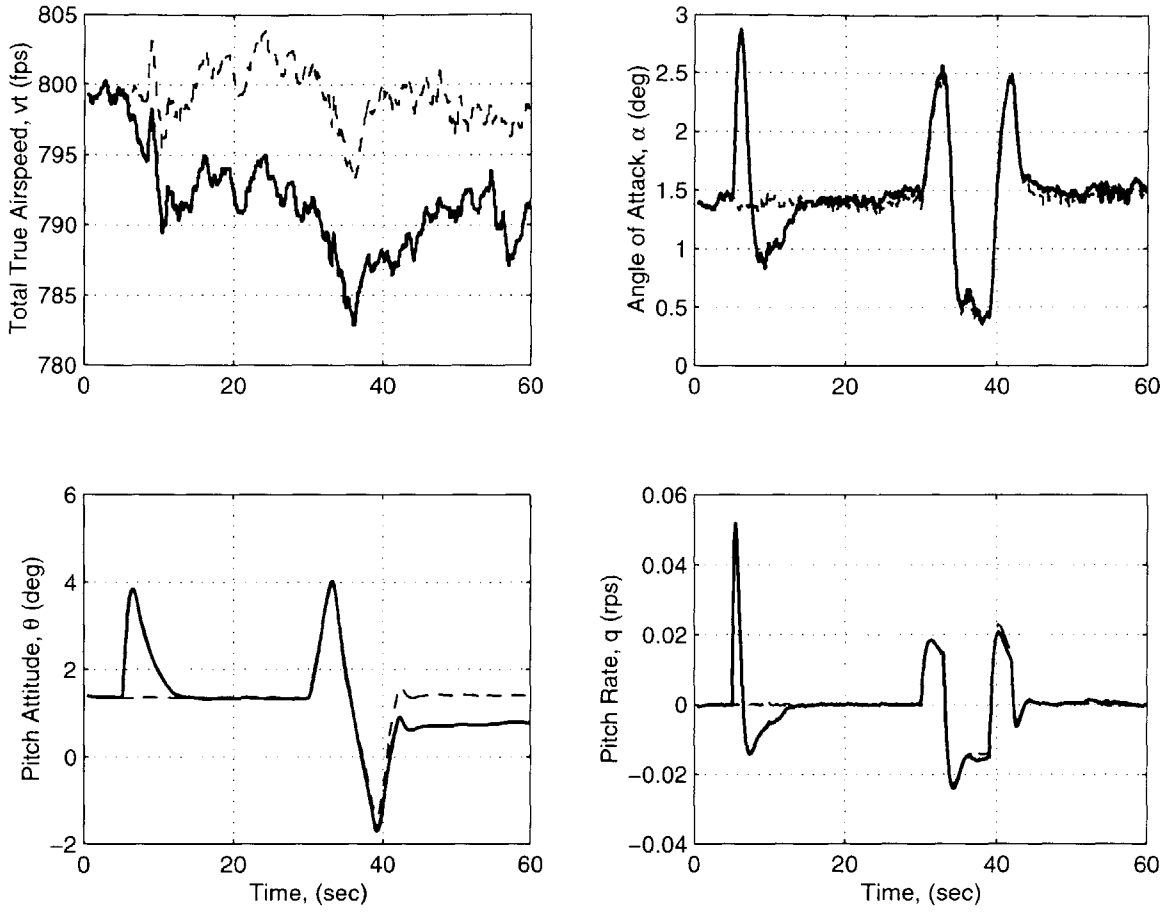

Figure 6-13: Comparison of Horizontal Tail Failure with RCM to Nominal Performance, $\delta_{stb} = -5.0^o$, 0.82 M, 35,000 ft (– RCM, - - Nominal)

failure disturbance is seen in the true airspeed $v_t(t)$ and the pitch angle $\Theta(t)$. Naturally, the horizontal tail position post-failure produces a greater trim drag; thus, the true airspeed is reduced. Likewise, small errors in the pitch rate get integrated into the pitch angle. Small differences can be observed in the peaks of the pitch rate trajectories at approximately 35 and 40 seconds.

A comparison between the post-failure and nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-15. It is seen from this figure that the Phugoid mode has a higher natural frequency and damping. Otherwise, the transfer functions match.

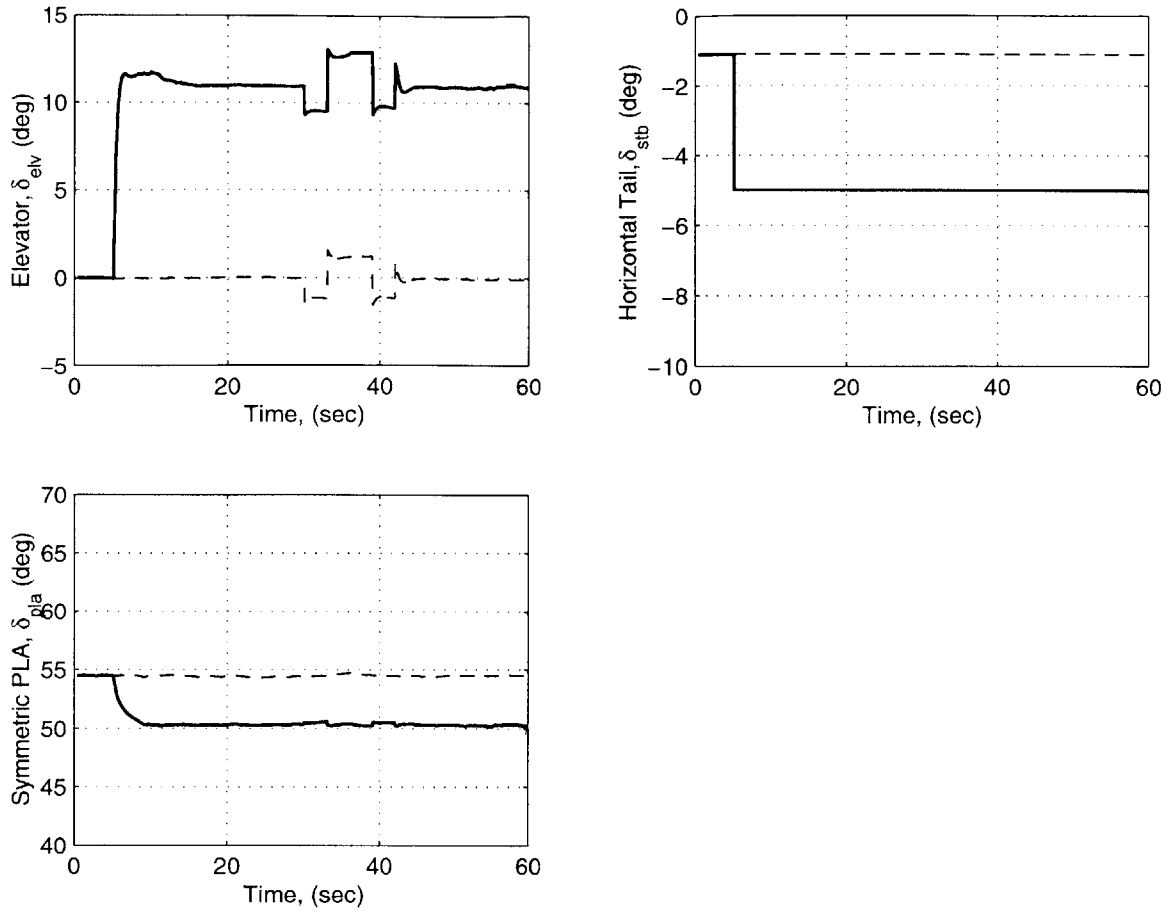

Figure 6-14: Surface Deflection Comparison of Horizontal Tail Failure with RCM to Nominal Performance, $\delta_{stb} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - Nominal)

Finally, the post-failure flying qualities are presented in Table 6.17. It is seen from this table that the post-failure aircraft with RCM has Level 1 flying qualities. Additionally, the the post-failure modes are close to the nominal modes.

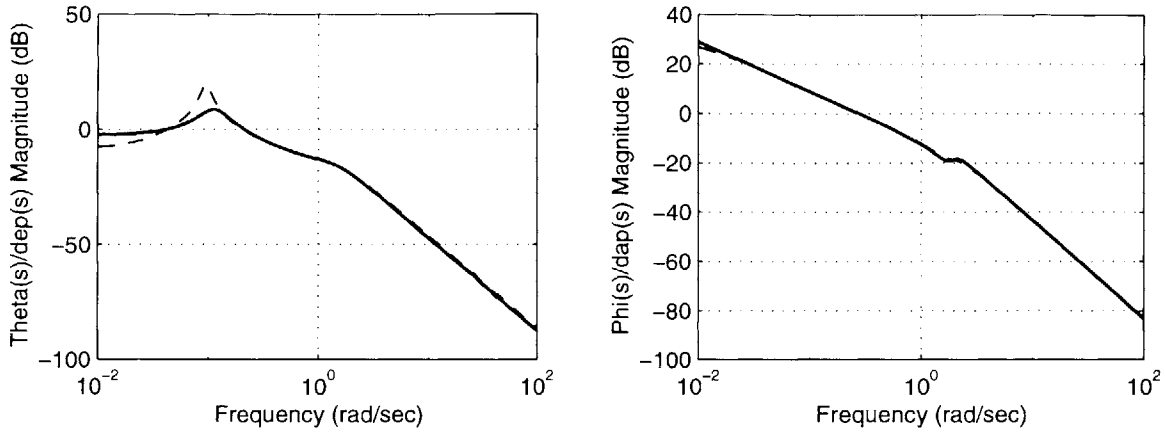

Figure 6-15: Comparison of Horizontal Tail Failure with RCM to Nominal Performance, $\delta_{stb} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - Nominal)

Table 6.17: Post-Failure Versus Nominal Aircraft Flying-Qualities for Horizontal Tail Failure: $\delta_{stb} = -5.0^o$, 0.82 M, 35,000 ft, Class III, Category B

Mode	Metric	Nom Value	Fail Value	Level
Short Period	ζ_{sp}	0.71	0.66	1
	$\omega_{n_{sp}}$	1.63 rad/sec	1.85	1
Phugoid	ζ_p	0.11	0.28	1
Spiral	T_{2_s}	Stable	Stable	1
Dutch Roll	ζ_d	0.28	0.22	1
	$\zeta_d\omega_{n_d}$	0.52 rad/sec	0.45	1
	ω_{n_d}	1.84 rad/sec	2.04	1
Roll	T_r	0.46 sec	0.51	1

Elevator Fixed Position Failure

The next failure scenario is a failure to the elevator. Based on the control authority at 35,000 feet and 280 knots, the stabilizer can cancel a static elevator deflection ranging from -25.0^o to 10.9^o. However, due to the bandwidth differences, it was determined by simulation that the horizontal tail and collective engine combination was only able to recover from instantaneous hard-over elevator deflection failures ranging from -8.0^o to 4.0^o. For failures that exceed this range, the aircraft either stalls or exceeds maximum dynamic pressure. Simulation data for an elevator failure with and without RCM are illustrated in Figures 6-16 and 6-17. At 5 seconds into the time history, the elevator is instantaneously moved to -5.0^o.

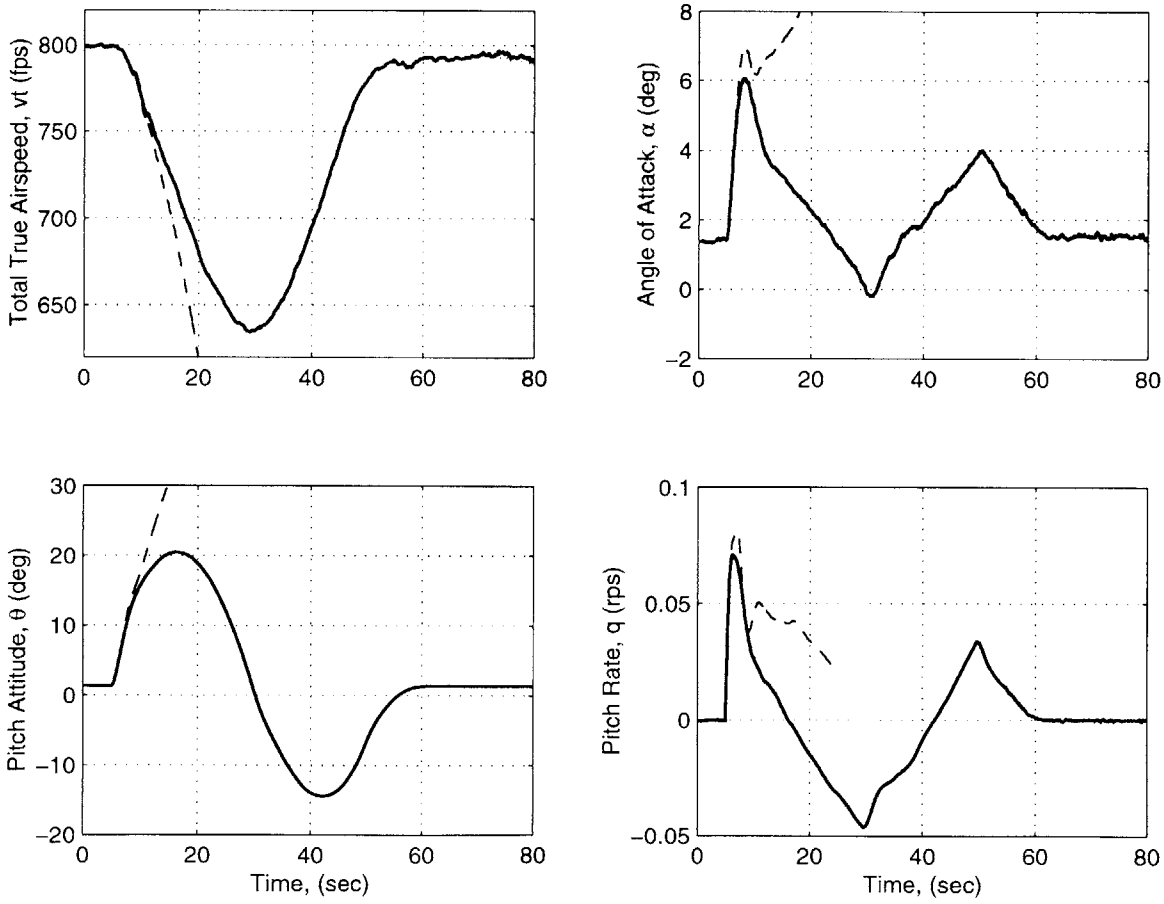

Figure 6-16: Elevator Failure, $\delta_{elv} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o RCM)

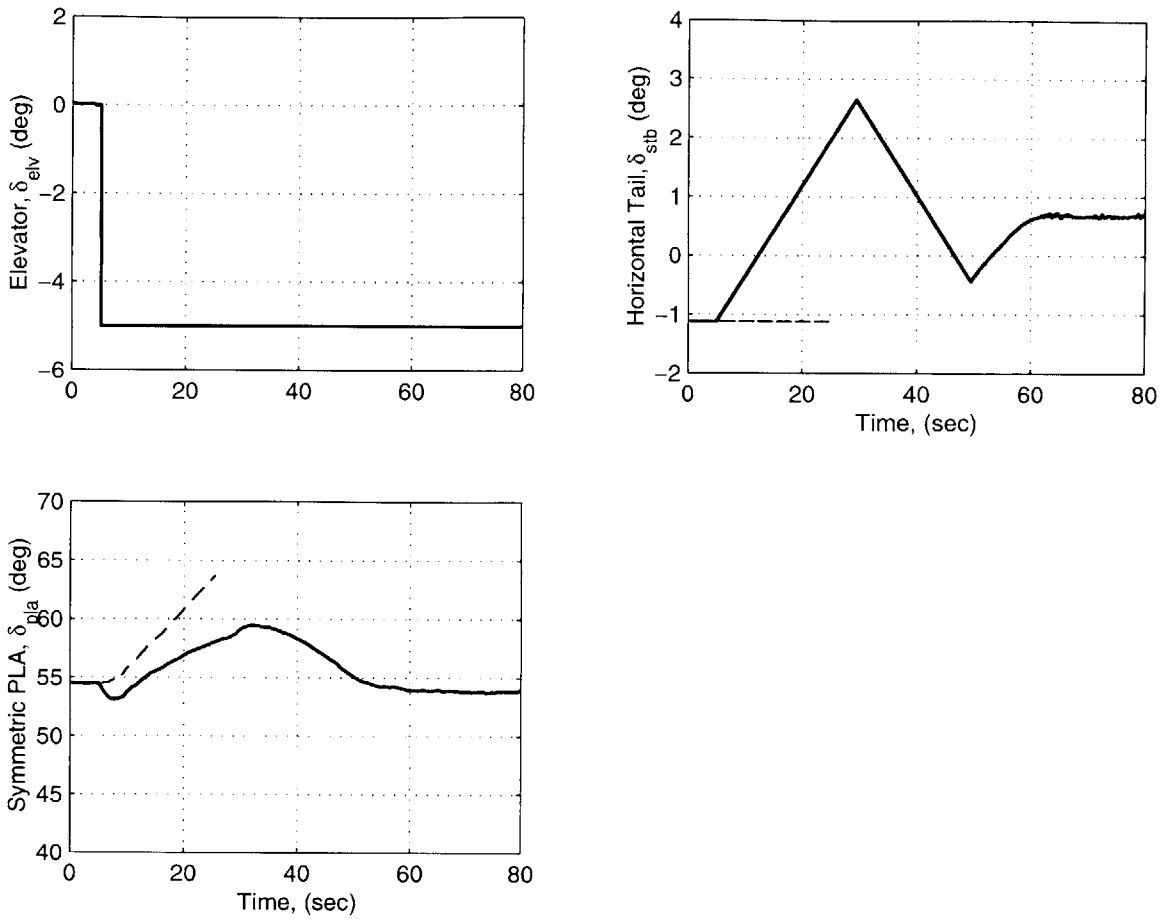

Figure 6-17: Surface Deflections for Elevator Failure, $\delta_{elv} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o RCM)

It is seen from this Figure 6-16 that without reconfiguration, the aircraft departs (note that the trajectories without RCM were stopped due to the departure at $t = 25.4$ seconds). With reconfiguration, the failure disturbance is attenuated and the aircraft returns to its initial trim conditions 55 seconds after the failure without pilot inputs. After 80 seconds, a pitch doublet is performed to demonstrate that the low bandwidth horizontal tail and collective throttles can provide pitch control during post-failure operations with normal stick inputs. Figures 6-18 and 6-19 illustrate the comparison. It is seen from this Figure 6-18 that there is a considerable loss of pitch control bandwidth. This result is expected since the horizontal tail's rate limit is $0.15^o/sec$ compared to the elevator's $80.0^o/sec$. Note, although there is considerable saturation experienced due to the reduction in pitch control bandwidth, the adaptive algorithm remains

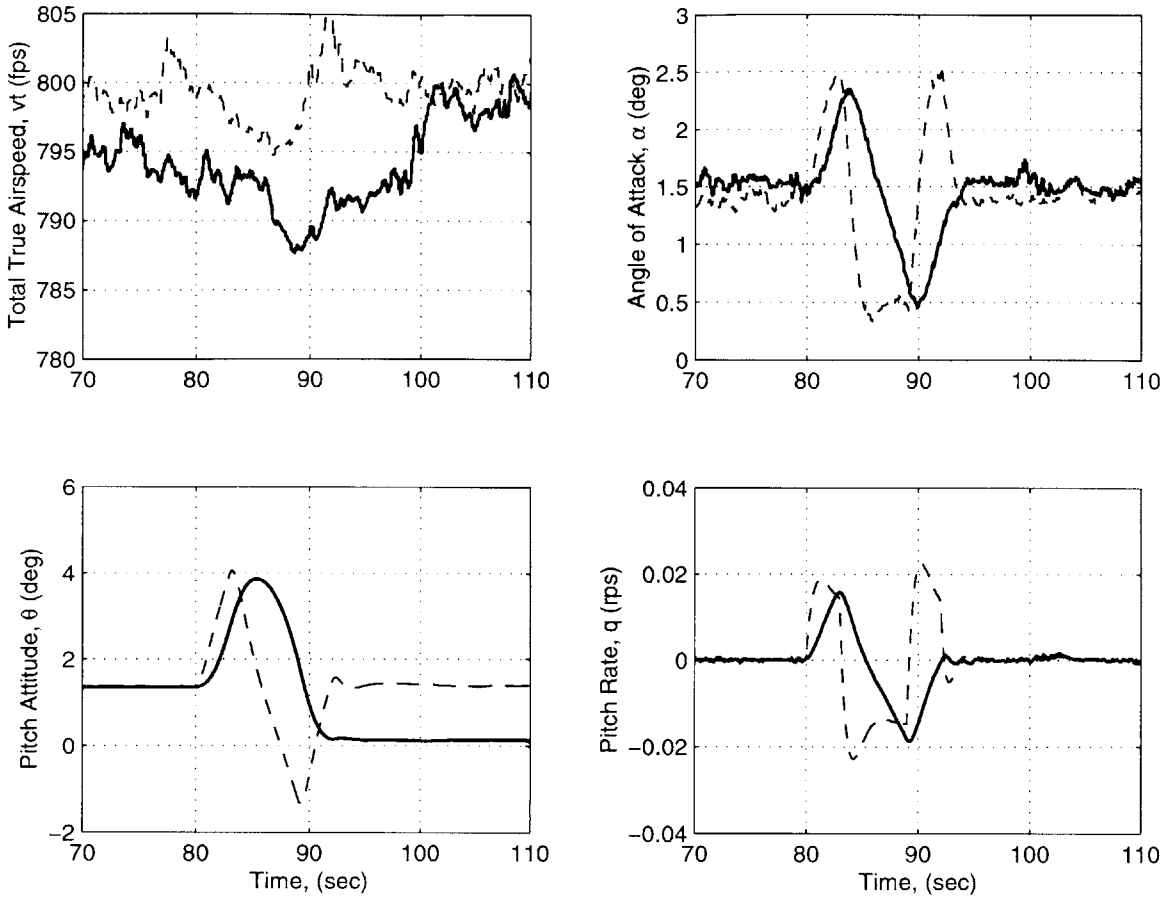

Figure 6-18: Comparison of Elevator Failure with RCM to Nominal Performance, $\delta_{elv} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o Nominal)

stable.

In addition to demonstrating the post-failure pitch authority, Figure 6-18 illustrates that the attitude hold mode, which attenuates the Phugoid mode, is still functioning. During normal operation, the attitude hold commands are given to the elevator only. As discussed in Section 6.2.1, these autopilot commands are converted to equivalent pitch stick commands by inverting the stick logic. Since these equivalent pitch stick commands are fed to the RCM, the attitude hold autopilot mode is replicated by the horizontal tail via the RCM.

A comparison between the post-failure and nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-20. This figure illustrates the

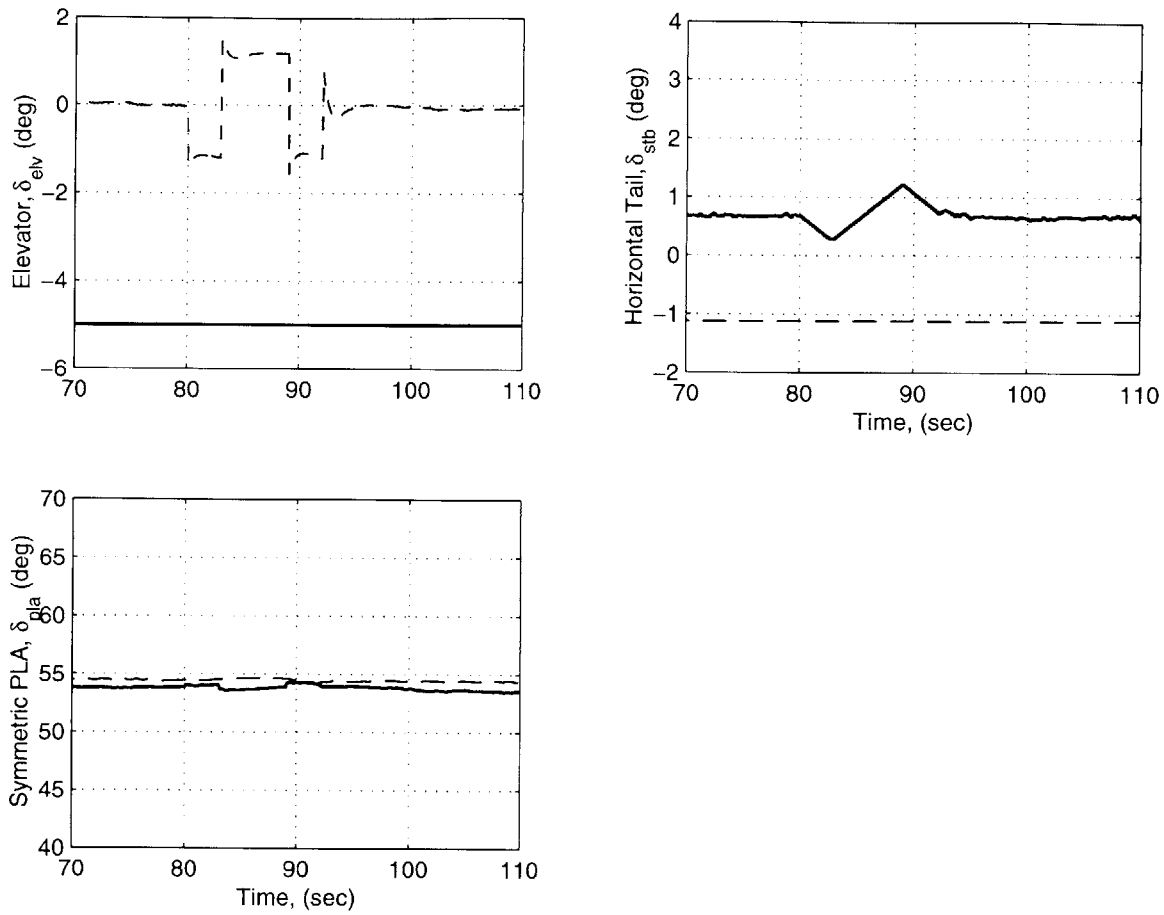

Figure 6-19: Surface Deflection Comparison of Elevator Failure with RCM to Nominal Performance, $\delta_{elv} = -5.0^o$, 0.82 M, 35,000 ft (RCM, - - w/o Nominal)

bandwidth reduction in the longitudinal axis. As expected, there is little change in the lateral response.

The post-failure flying qualities are presented in Table 6.18. It is seen from this table that the post-failure aircraft with RCM has below Level 3 flying qualities for the short period undamped natural frequency due to the bandwidth limitations. As expected, the lateral-directional post-failure modes are close to the nominal modes.

A final comment on this failure scenario is necessary. The post-failure performance deficiencies are due to the lack of pitch control bandwidth. If the elevator is split, right and left, with independent actuation, there would be adequate bandwidth post-failure if either one fails. Thus,

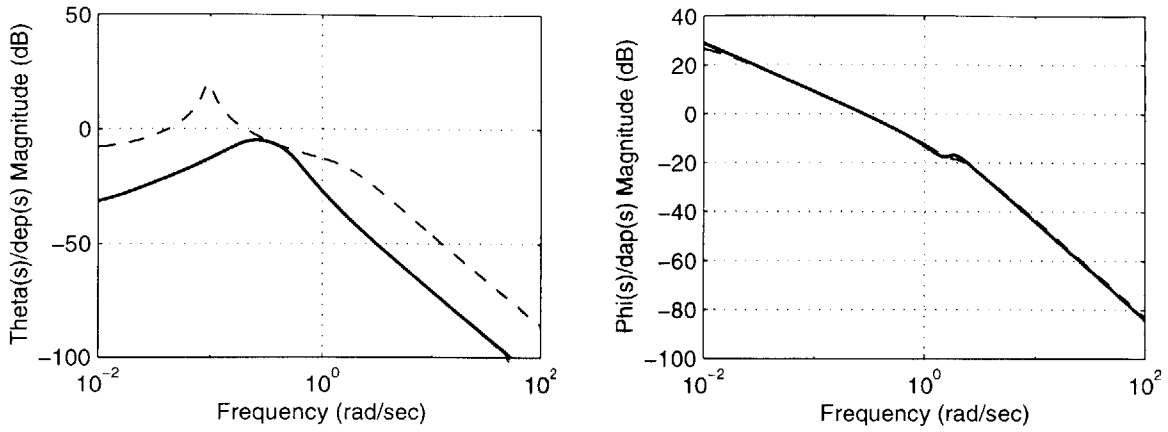

Figure 6-20: Comparison of Elevator Failure with RCM to Nominal Performance, $\delta_{elv} = -5.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o Nominal)

Table 6.18: Post-Failure Versus Nominal Aircraft Flying-Qualities for Elevator Failure: $\delta_{elv} = -5.0^o$, 0.82 M, 35,000 ft, Class III, Category B

Mode	Metric	Nom Value	Fail Value	Level
Short Period	ζ_{sp}	0.71	0.42	1
	$\omega_{n_{sp}}$	1.63 rad/sec	0.47	< 3
Phugoid	ζ_p	0.11	0.51	1
Spiral	T_{2_s}	Stable	Stable	1
Dutch Roll	ζ_d	0.28	0.24	1
	$\zeta_d\omega_{n_d}$	0.52 rad/sec	0.43	1
	ω_{n_d}	1.84 rad/sec	1.79	1
Roll	T_r	0.46 sec	0.58	1

the post-failure performance would resemble nominal performance. To illustrate this bandwidth deficiency for the current failure scenario, Figure 6-21 illustrates the pitch stick to pitch angle frequency response for various simulated horizontal tail rate limits. The two solid lines in this figure represent the results illustrated in Figure 6-20, and the dashed lines are the intermediate rate limits. It is seen from this figure that as the rate limit increases, the bandwidth approaches the nominal response. As noted previously, the frequency data below 0.1 rad/sec should not be viewed as reliable data.

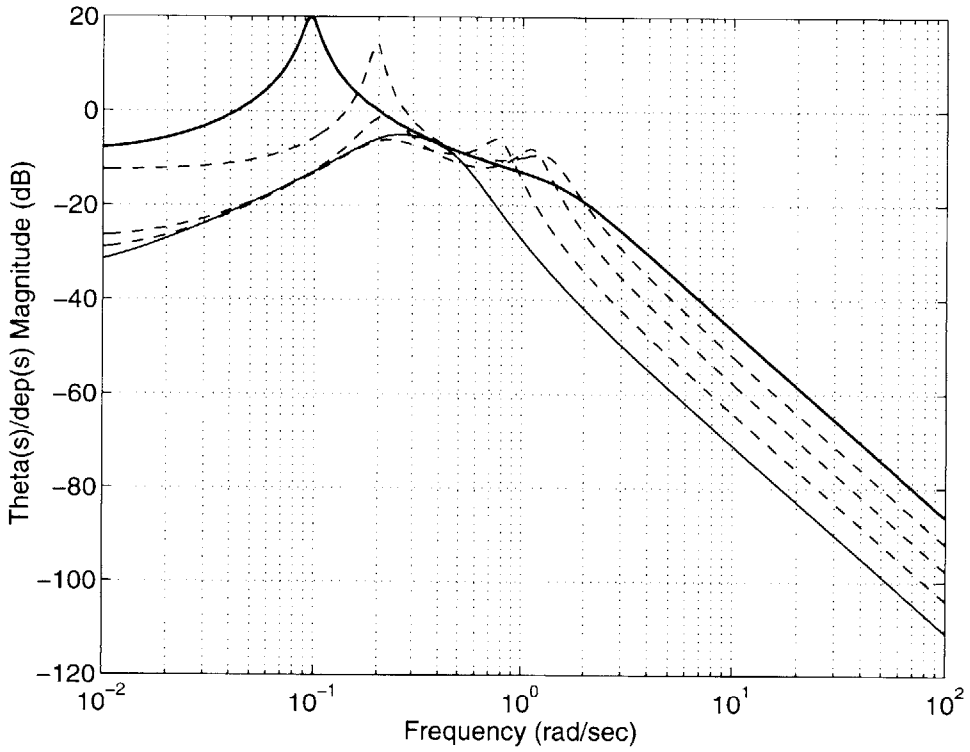

Figure 6-21: Comparison of Pitch Stick to Pitch Angle Performance for Various Horizontal Tail Rate Limits with an Elevator Failure, $\delta_{elv} = -5.0^o$, 0.82 M, 35,000 ft (from left to right in figure — $0.15^o/sec$, - - $0.15x2.5^o/sec$, - - $0.15x5.0^o/sec$, - - $0.15x20^o/sec$, —Nominal)

Empennage Hydraulic Failure

The final longitudinal failure scenario demonstrated is the loss of hydraulics to both the elevator and stabilizer. For this failure, the two surfaces are failed at there trim setting and the engines are used to pitch the aircraft. Figures 6-22 and 6-23 present a comparison of post-failure and nominal performance for a pitch doublet. As expected, the moment generated by the collective

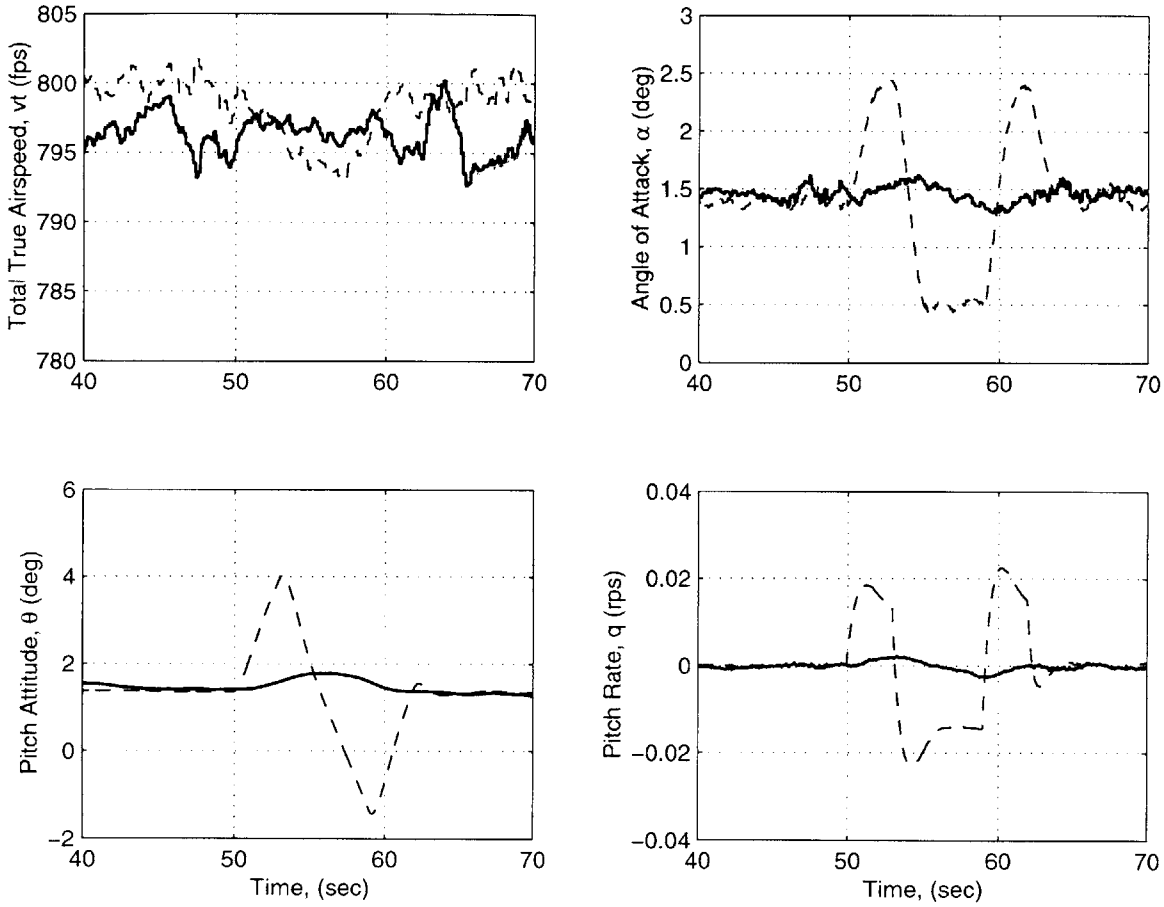

Figure 6-22: Comparison of Empennage Hydraulic Failure with RCM to Nominal Performance, $\delta_{elv} = 0.0^o, \delta_{stb} = -1.12^o$, 0.82 M, 35,000 ft (— RCM, - - w/o Nominal)

engine throttle is small, but the integral of this small moment does produce a low-bandwidth pitch capability.

Figures 6-24 and 6-25 illustrate the pitch authority for longer pitch stick inputs. For these trajectories, full pitch stick inputs were held until the aircraft attains a climb rate of $\pm 1,000$

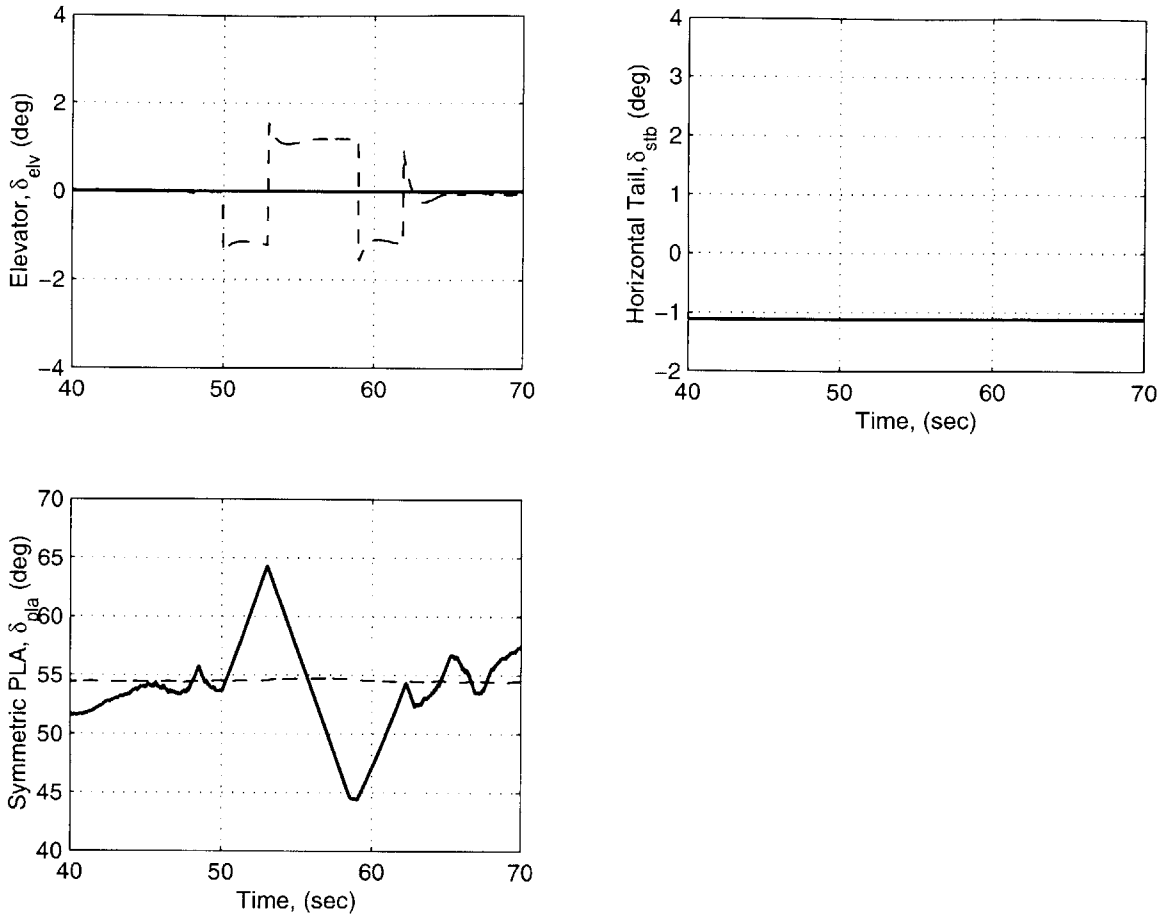

Figure 6-23: Surface Deflection Comparison of Empennage Hydraulic Failure with RCM to Nominal Performance, $\delta_{elv} = 0.0^o, \delta_{stb} = -1.12^o$, 0.82 M, 35,000 ft (— RCM, - - w/o Nominal)

feet-per-minute (fpm). It is seen that it takes about 15 seconds to change the climb rate from $0 \rightarrow 1,000$ fpm. Also note from these figures that the engines are attenuating the Phugoid mode via the equivalent pitch stick input from the autopilot attitude hold mode. Furthermore, although there is considerable saturation experience during this slow maneuver, the adaptive algorithm remains stable.

A comparison between the post-failure and nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-26. This figure illustrates the bandwidth reduction in the longitudinal axis. As expected, there is no change in the lateral response.

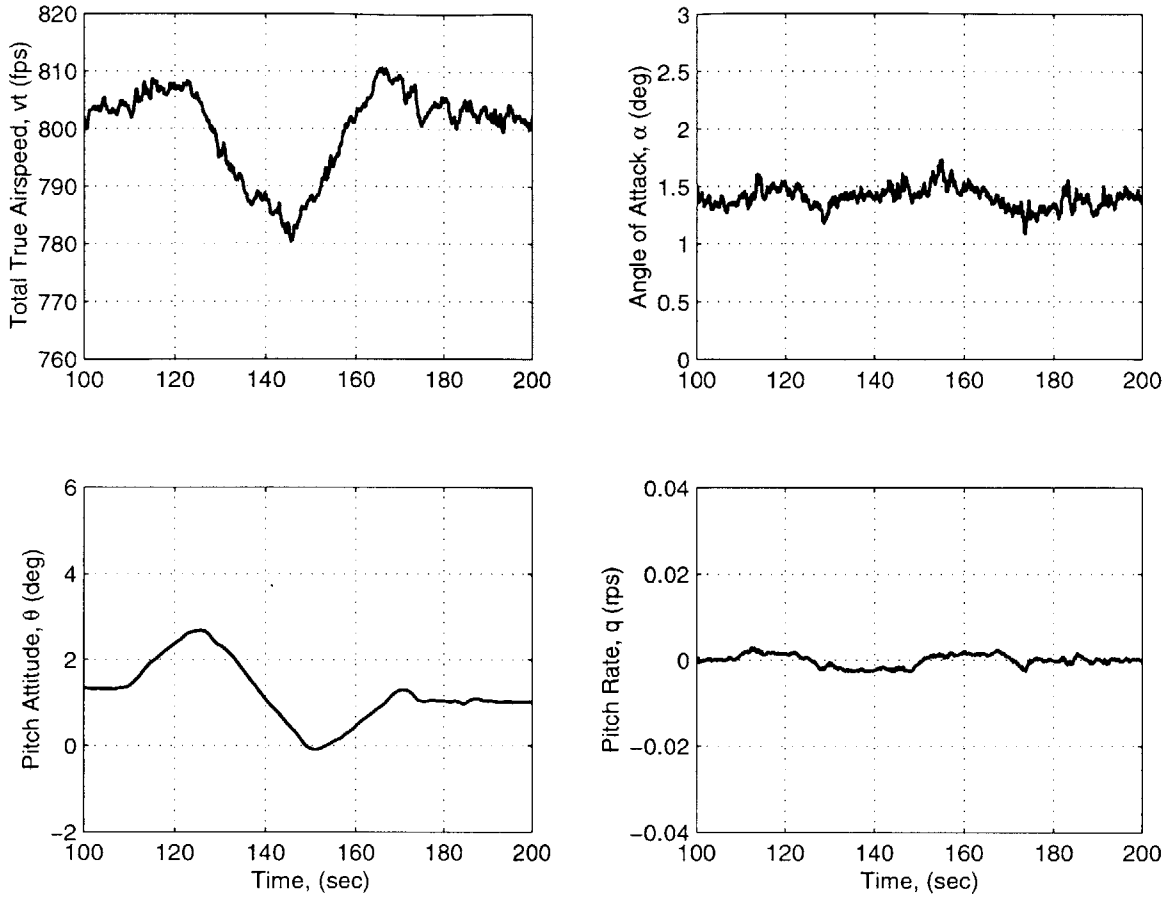

Figure 6-24: Empennage Hydraulic Failure, $\delta_{elv} = 0.0^o, \delta_{stb} = -1.12^o$, 0.82 M, 35,000 ft (RCM)

Finally, the post-failure flying qualities are presented in Table 6.19. It is seen from this table that the post-failure aircraft with RCM has below Level 3 flying qualities for the short period mode due to the bandwidth limitations. As expected, the lateral-directional post-failure modes are close to the nominal modes.

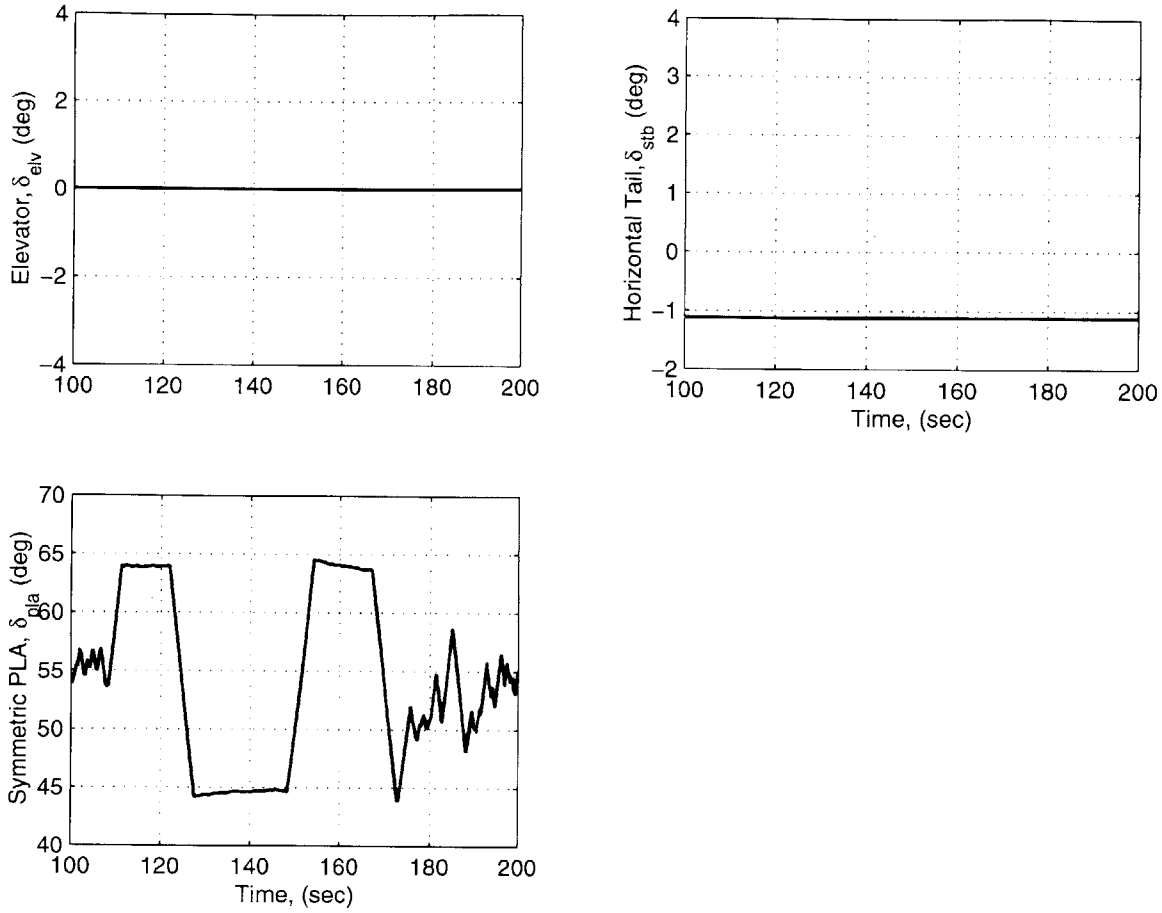

Figure 6-25: Surface Deflections for Empennage Hydraulic Failure, $\delta_{elv} = 0.0^o, \delta_{stb} = -1.12^o$, 0.82 M, 35,000 ft (– RCM)

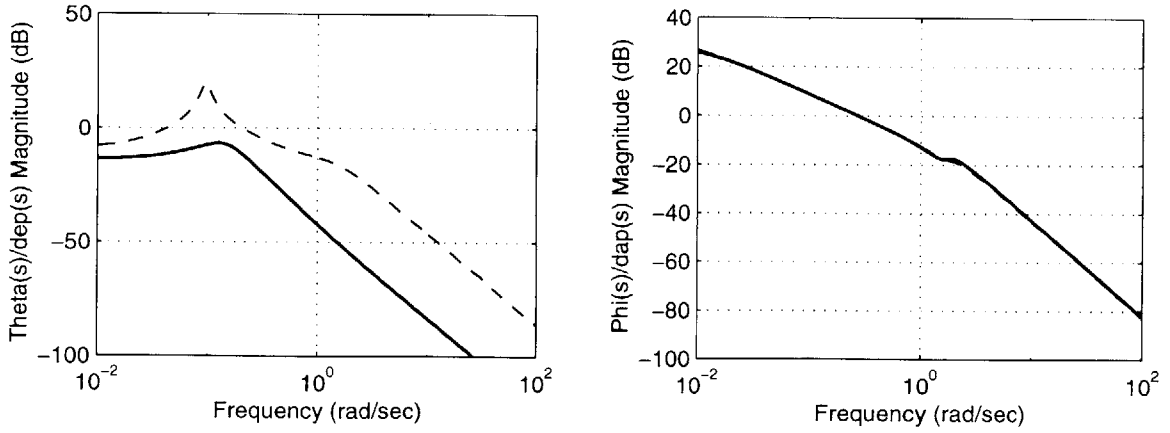

Figure 6-26: Empennage Hydraulic Failure with RCM to Nominal Performance, $\delta_{elv} = 0.0^o, \delta_{stb} = -1.12^o$, 0.82 M, 35,000 ft (— RCM, - - w/o Nominal)

Table 6.19: Post-Failure Versus Nominal Aircraft Flying-Qualities for Empennage Hydraulic Failure, $\delta_{elv} = 0.0^o, \delta_{stb} = -1.12^o$, 0.82 M, 35,000 ft, Class III, Category B

Mode	Metric	Nom Value	Fail Value	Level
Short	ζ_{sp}	0.71	Degenerate	< 3
Period	$\omega_{n_{sp}}$	1.63 rad/sec	Degenerate	< 3
Phugoid	ζ_p	0.11	0.45	1
Spiral	T_{2_s}	Stable	Stable	1
Dutch Roll	ζ_d	0.28	0.27	1
	$\zeta_d\omega_{n_d}$	0.52 rad/sec	0.50	1
	ω_{n_d}	1.84 rad/sec	1.81	1
Roll	T_r	0.46 sec	0.47	1

6.3.4 Lateral-Directional Failures

In this subsection, three lateral-directional failure scenarios will be presented: spoiler fixed position failure, aileron fixed position failure, and rudder fixed position failure. Each of these failure scenarios are simulated at the selected cruise condition and a comparison between the nominal and post-failure performance, with and without reconfiguration, will be presented. Furthermore, nominal and post-failure frequency responses and aircraft handling qualities will be compared.

As in the longitudinal failure simulations, atmospheric disturbances are included in these simulations, and the regressor disturbance upperbound for the robust adaptation algorithm is set to $V_o = \hat{J}(t)[0.01 \quad 0.01 \quad 0.002]^T$. Also, the failure-logic switch is closed and remains closed when the output performance error $e_o(t)$ exceeds $[0.05 \quad 0.03 \quad 0.03]^T \; rad/sec^2$.

Spoiler Fixed Position Failure

The first lateral-directional failure scenario that will be presented is a hard-over failure to the left furthest outboard spoiler. Because of the redundancy in the lateral axis, any single surface fixed position failure can be accommodated by the RCM. Simulation data for the left outboard spoiler failure with and without RCM is illustrated in Figures 6-27 and 6-28. At 5.0 seconds into the simulation, a left outboard spoiler is deflected to -60.0^o instantaneously. It is seen from this Figure 6-27 that without reconfiguration, the aircraft rapidly departs. It is seen from Figure 6-28 that the nominal controller does have a level of robust stability. The nominal controller integrates the roll rate error, and commands an aileron deflection to cancel the roll rate error. However, at this flight condition, the ailerons alone cannot generate a moment great enough to cancel a spoiler hard-over failure; thus the ailerons magnitude saturate, and the aircraft departs. With reconfiguration, the roll spoilers on the right wing along with the ailerons are able to capture the failure disturbance. For the adaptation gain used, the maximum bank angle produce by the failure is approximately -9.0^o. After the failure disturbance is captured, i.e. $p(t) \rightarrow 0$, the bank angle slowly decreases between the time interval $10 \rightarrow 20$ seconds. The spiral

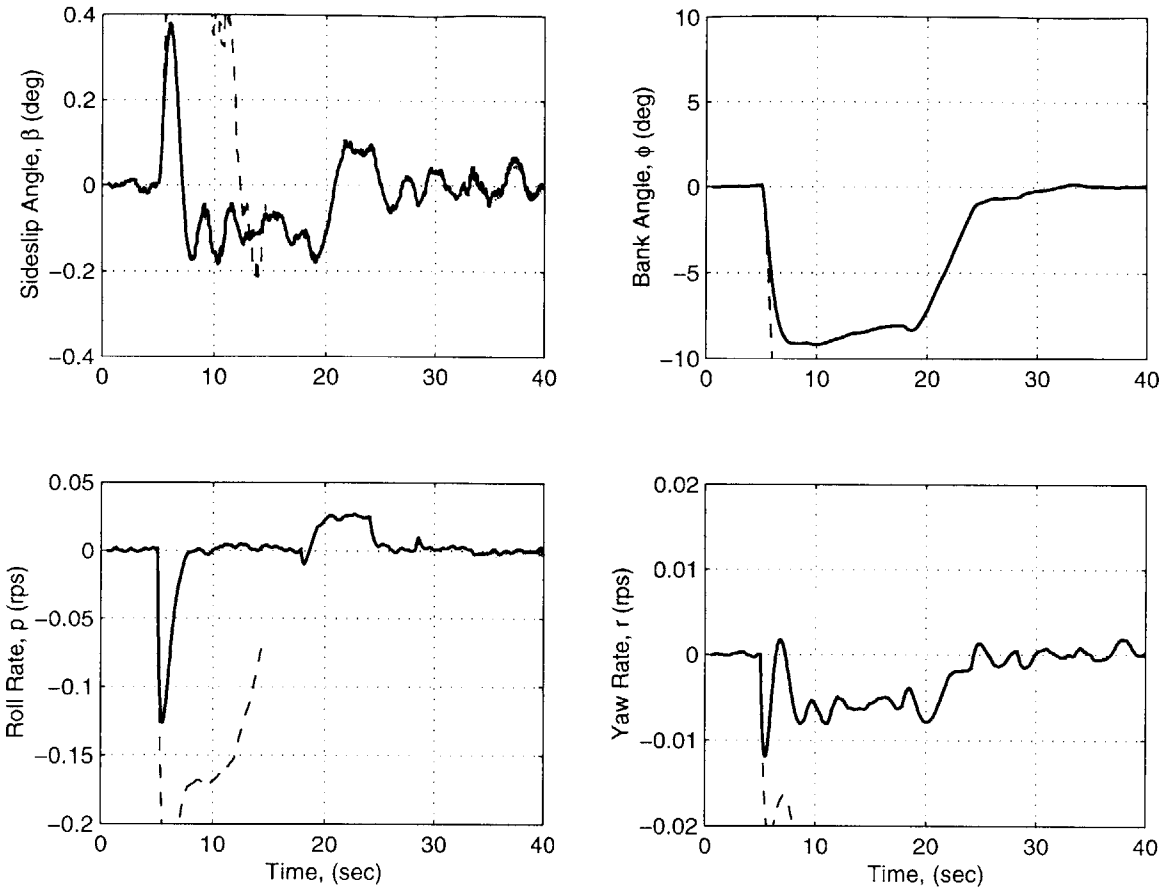

Figure 6-27: Left Outboard Spoiler Failure, $\delta_{spl} = -60.0^o$, 0.82 M, 35,000 ft(— RCM, - - w/o RCM)

time constant is on the order of 100 seconds. At 20 seconds, the pilot levels the wings.

After the pilots levels the wings, open-loop pilot commands are inputed to provide a comparison between post-failure performance and nominal performance. Figures 6-29 and 6-30 illustrate the comparison. It is seen from this Figure 6-29 that for the roll stick doublet, nominal performance is completely recovered post-failure.

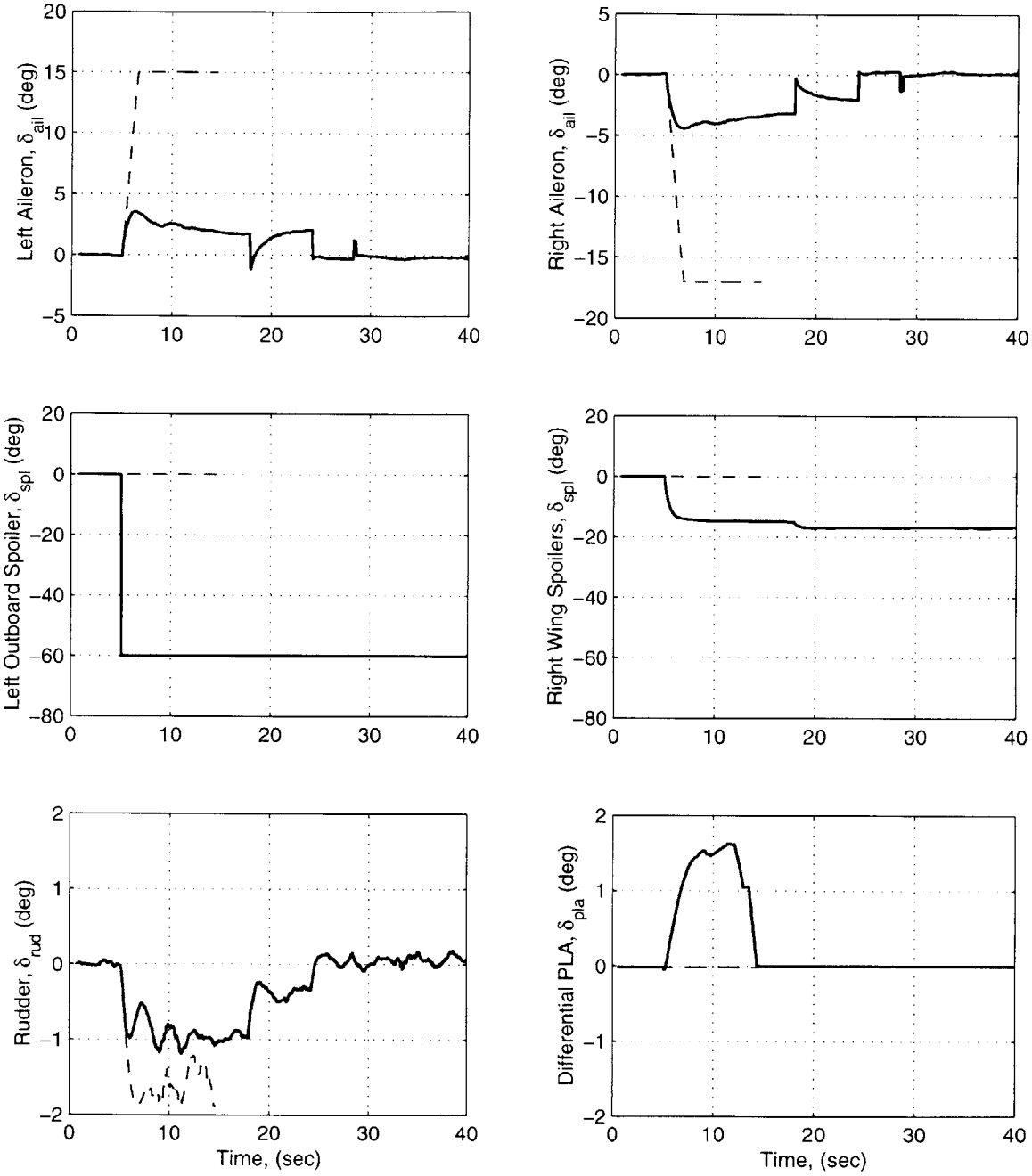

Figure 6-28: Surface Deflections for Left Outboard Spoiler Failure, $\delta_{spl} = -60.0^o$, 0.82 M, 35,000 ft(--- RCM, - - w/o RCM)

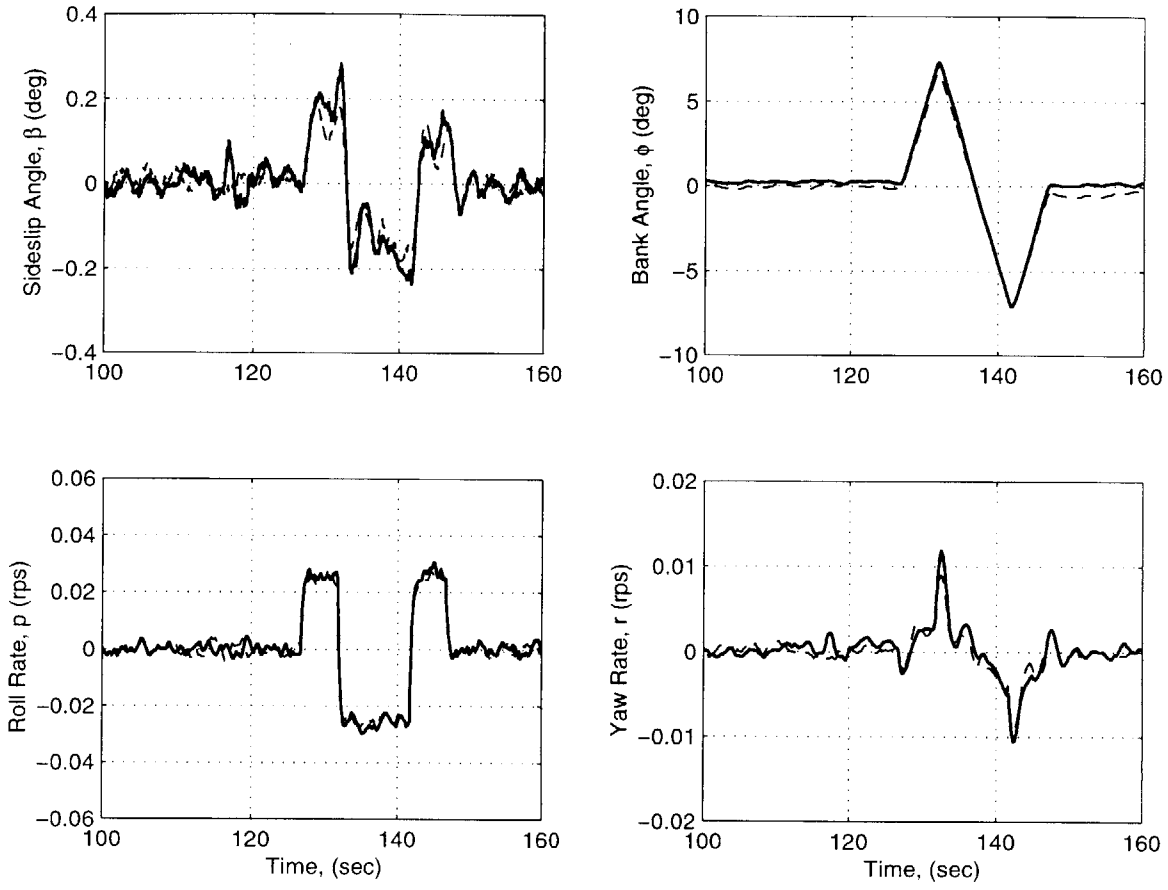

Figure 6-29: Comparison of Spoiler Failure with RCM to Nominal Performance, $\delta_{spl} = -60.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

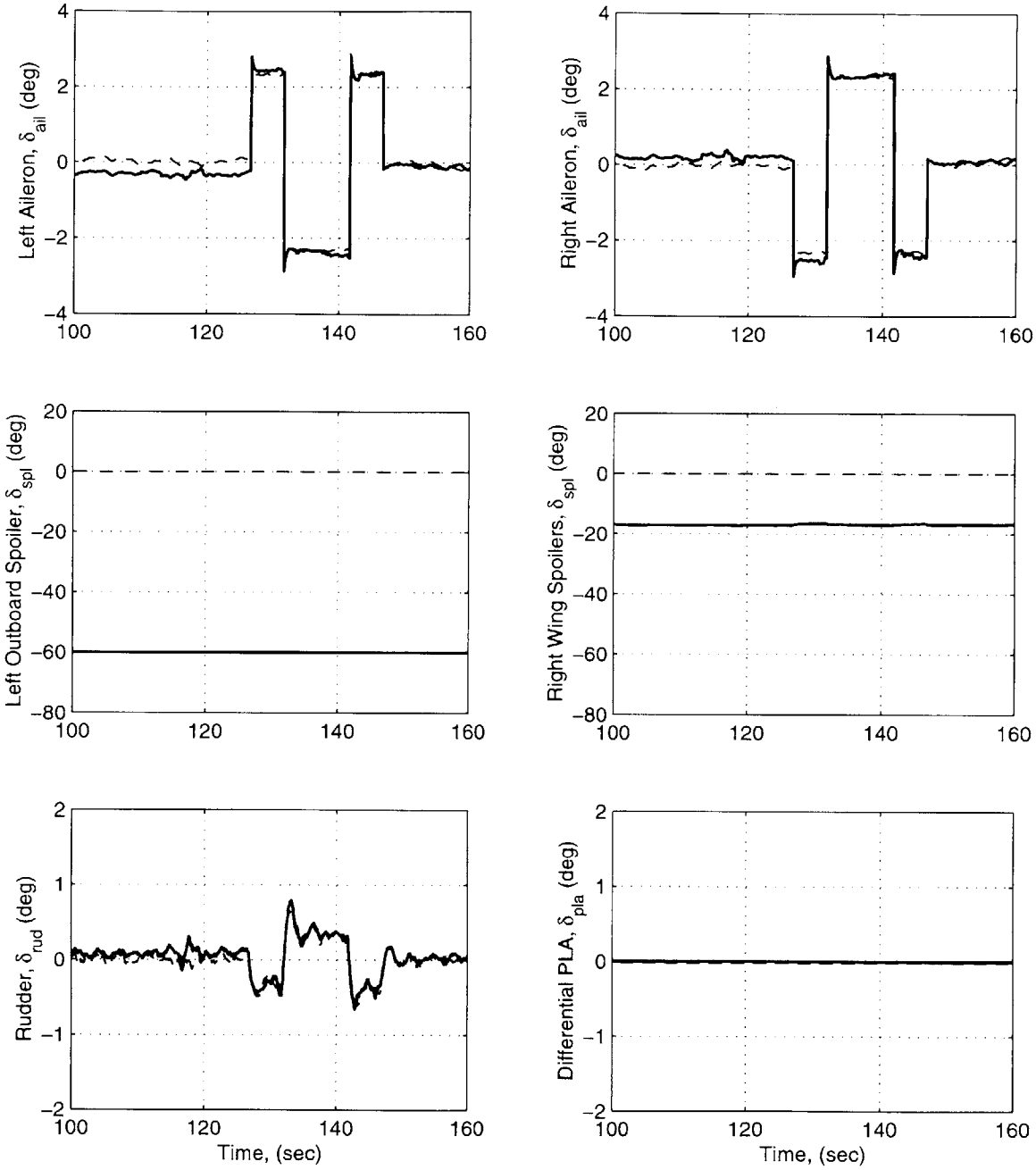

Figure 6-30: Surface Deflection Comparison of Spoiler Failure with RCM to Nominal Performance, $\delta_{spl} = -60.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

A comparison between the post-failure and nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-31. It is seen from this figure that within the viable frequency range of the input signals, the frequency responses match.

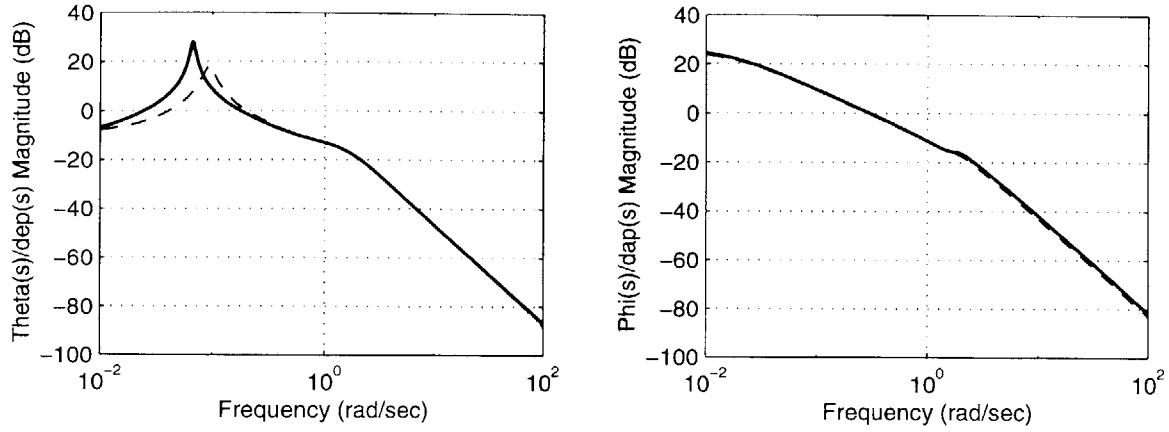

Figure 6-31: Comparison of Spoiler Failure with RCM to Nominal Frequency Response, $\delta_{spl} = -60.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

Next, the post-failure flying qualities are presented in Table 6.20. It is seen from this table that the post-failure aircraft with RCM has Level 1 flying qualities. Additionally, the post-failure modes are close to the nominal modes.

Table 6.20: Post-Failure Versus Nominal Aircraft Flying-Qualities for Spoiler Failure: $\delta_{spl} = -60.0^o$, 0.82 M, 35,000 ft, Class III, Category B

Mode	Metric	Nom Value	Fail Value	Level
Short Period	ζ_{sp}	0.71	0.69	1
	$\omega_{n_{sp}}$	1.63 rad/sec	1.61	1
Phugoid	ζ_p	0.11	0.04	1
Spiral	T_{2_s}	Stable	Stable	1
Dutch Roll	ζ_d	0.28	0.26	1
	$\zeta_d\omega_{n_d}$	0.52 rad/sec	0.50	1
	ω_{n_d}	1.84 rad/sec	1.89	1
Roll	T_r	0.45 sec	0.43	1

As mentioned above, the nominal control system does have a certain level of robustness to roll rate errors. The nominal controller integrates the roll rate error, and commands an aileron

deflection in an attempt to cancel the roll rate error caused by the spoiler failure. Though this inherent robustness is not adequate alone to accommodate this spoiler failure, it does present the opportunity to illustrate how the RCM interacts with a nominal controller to accommodate failures. Figure 6-32 illustrates the ailerons and spoiler deflections required to attenuate the spoiler failure. In this figure, the aileron deflections correspond to the differential aileron deflec-

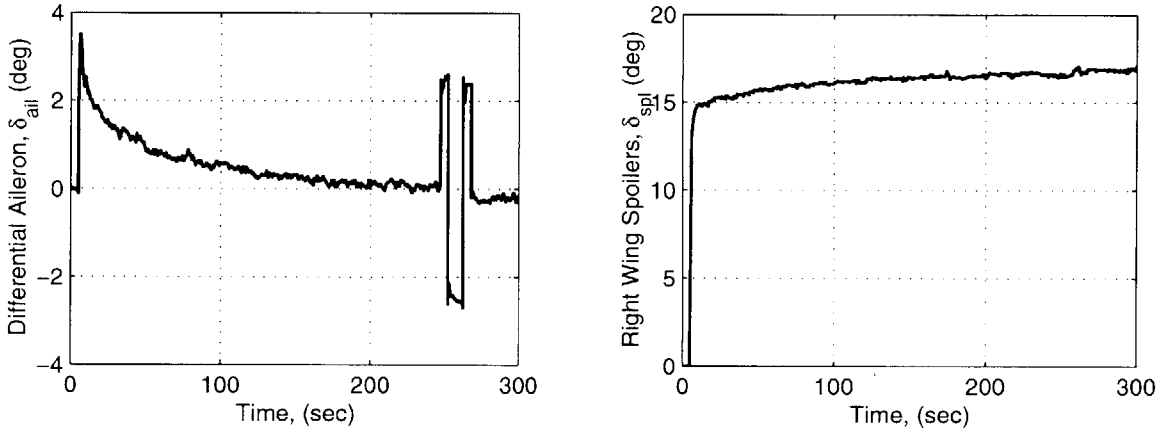

Figure 6-32: Differential Aileron and Spoiler Deflections for Spoiler Failure, $\delta_{spl} = -60.0^o$, 0.82 M, 35,000 ft

tions $\delta_{ail_{dif}}(t)$ commanded by the nominal control system, and the spoiler deflections correspond to the differential spoiler deflections $\delta_{spl_{dif}}(t)$ commanded by the RCM. For this simulation, no pilot inputs were used to level the wings, instead the spiral mode was allowed to attenuate. After the failure is introduced, the nominal control system commands 3.5^o of differential aileron deflection while the RCM commands 14.0^o of differential spoiler deflection. The combination of these two capture the roll rate error. As the spiral mode decays, the nominal control roll rate integrators integrate down which drives the ailerons to there trim settings. This process results in the nominal controller releasing trim control authority to the reconfiguration module. Around 230 seconds into this simulation, the ailerons have converged to 0.0^o and the spoilers have converged to 16.5^o; thus, the RCM has full authority over the spoiler failure disturbance. During the time interval from 240 to 270 seconds, a roll doublet is performed. It is seen that the ailerons alone produce the desired rolling moment and the RCM does not contribute any additional commands to perform this maneuver. Since the spoilers are not used for roll control at this flight condition, the spoiler failure does not affect the nominal control system's capability

to generate the desired roll moment.

In summary, this simulation illustrates how the RCM interacts with the nominal control system to accommodate failure. If the nominal control system has some inherent robustness, the RCM will only inject whatever is required to obtain the desired performance. Thus, if a nominal control system has a high-degree of inherent robustness, the RCM will do very little; if the nominal control system has a low-degree of robustness, the RCM will do all of the reconfiguration task.

Aileron Fixed Position Failure

The next failure disturbance scenario that is presented is a hard-over failure to the left aileron. At 5.0 seconds into the simulation, the left aileron is moved to -25.0^o instantaneously. Recall that the aileron deflections range from -25.0^o to 15.0^o; so the nominal controller should be able to accommodate this failure if it integrates the right aileron to -25.0^o. Figures 6-33 and 6-34 illustrate the nominal performance and failure performance with and without reconfiguration.

As illustrated in Figure 6-33, without reconfiguration, the aircraft departs rapidly in roll. It is seen from Figure 6-34 that the nominal control system integrates the right aileron to -17.0^o instead of the required -25.0^o. With reconfiguration, the failure disturbance is captured through the combination of right aileron and left wing spoilers. The maximum bank angle produced by the failure is approximately -4.6^o. After the failure disturbance is captured, the bank angle slowly decreases during the time interval between 10 and 20 seconds. At 20 seconds, the pilot levels the wings.

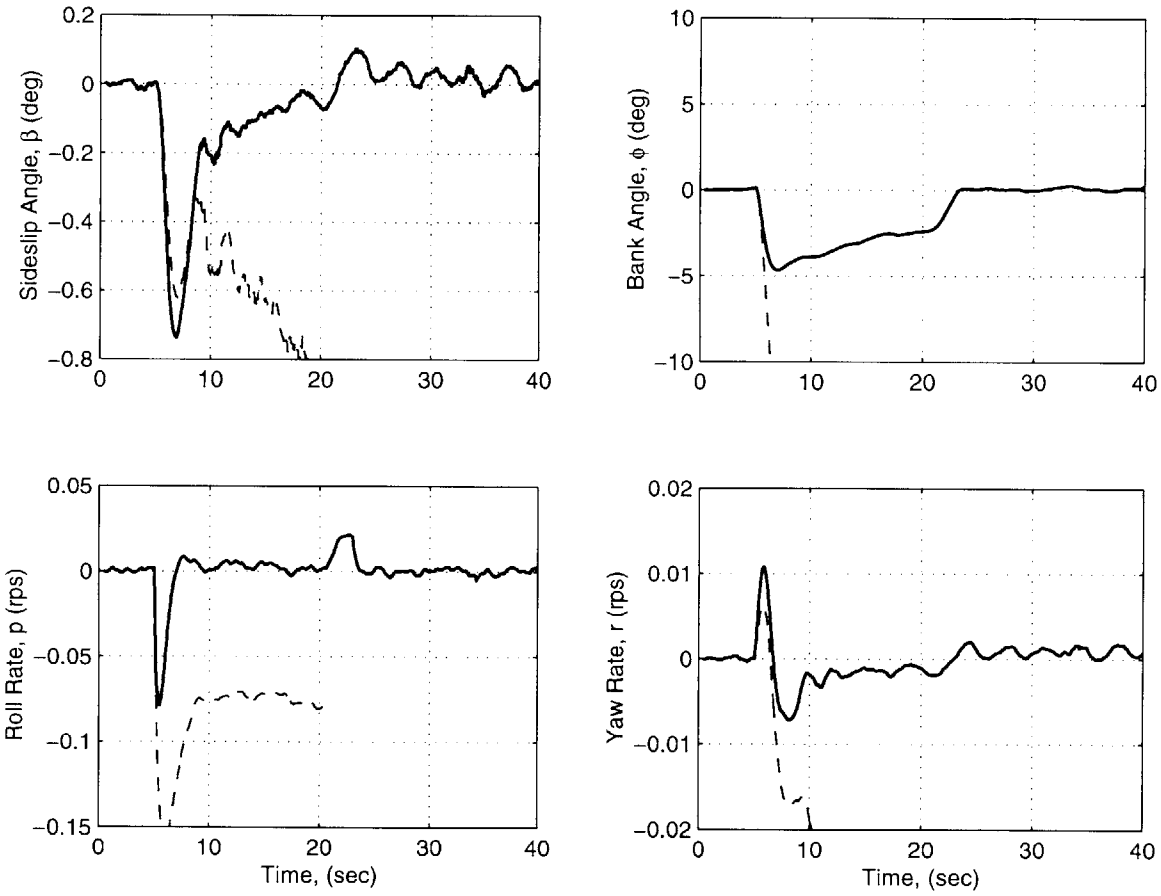

Figure 6-33: Left Aileron Failure, $\delta_{ail_l} = -25.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o RCM)

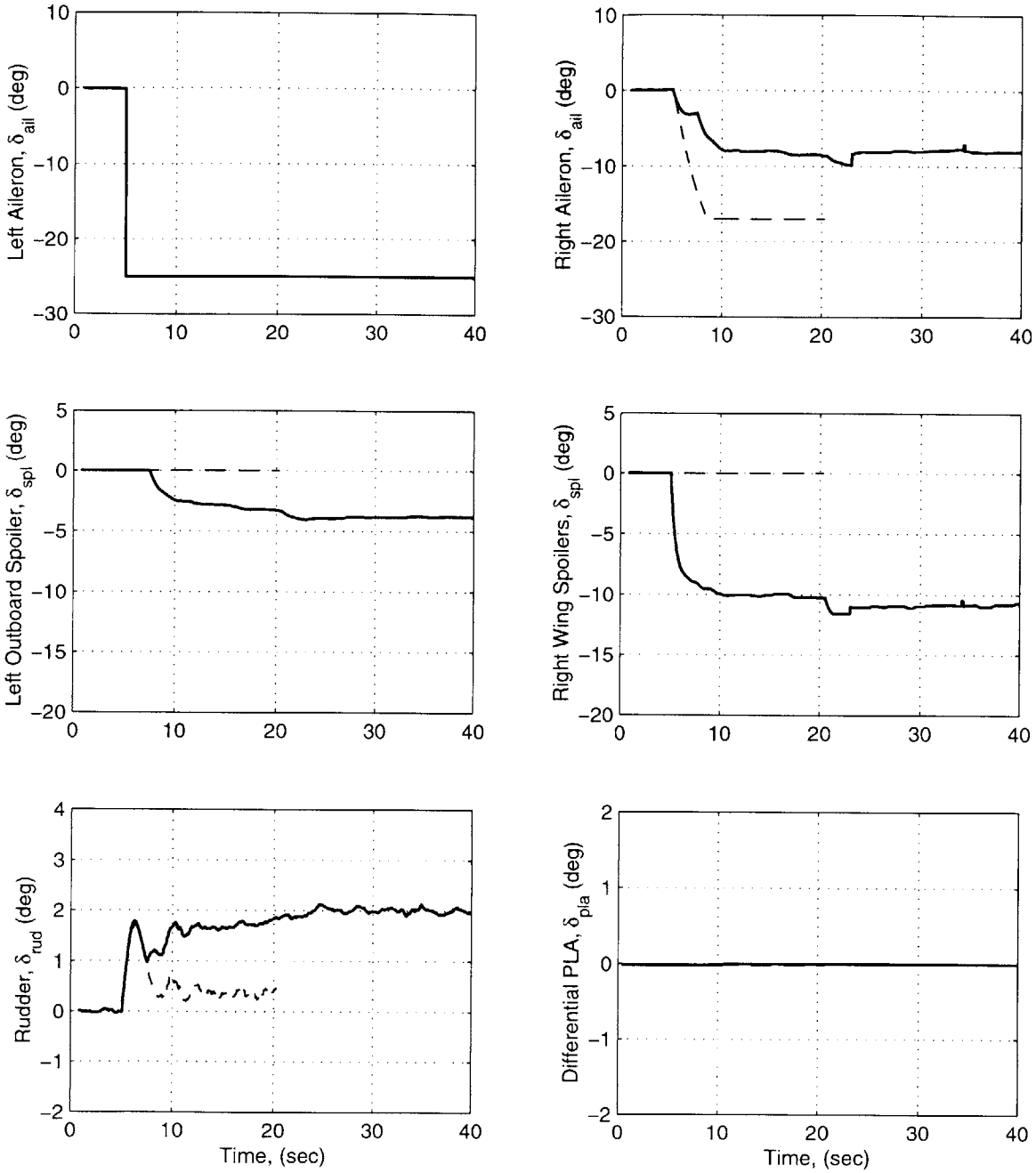

Figure 6-34: Surface Deflections for Left Aileron Failure, $\delta_{ail_l} = -25.0^o$, 0.82 M, 35,000 ft (- RCM, - - w/o RCM)

After the pilot levels the wings, open-loop pilot commands are given to provide a comparison between post-failure performance and nominal performance. Figures 6-35 and 6-35 illustrate the comparison. It is seen from Figure 6-35 that for the roll stick doublet, nominal performance is

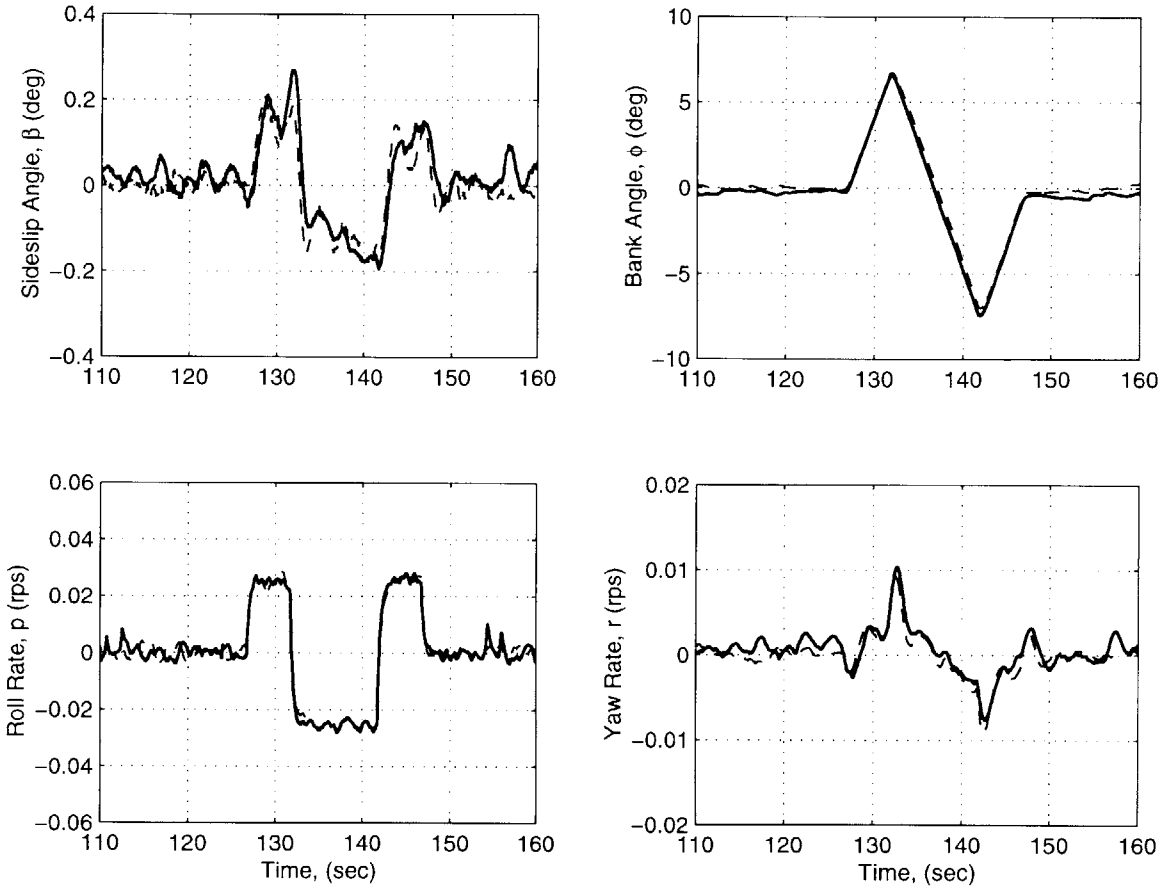

Figure 6-35: Comparison of Aileron Failure with RCM to Nominal Performance, $\delta_{ail_l} = -25.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

completely recovered post-failure. There is one primary distinction between the aileron failure and the spoiler failure. In both cases, the failures introduce a disturbance that must be attenuated. In both cases, the aircraft's stability derivatives are not changed by the failures. For the spoiler failure, the nominal control power is not affected. However, for the aileron failure, the aircraft's roll control derivative is reduced by half; thus, the RCM must provide roll control to achieve the desired performance. This additional roll control command is illustrated in Figure 6-36 by viewing the larger right aileron deflections and the right wing spoiler deflections.

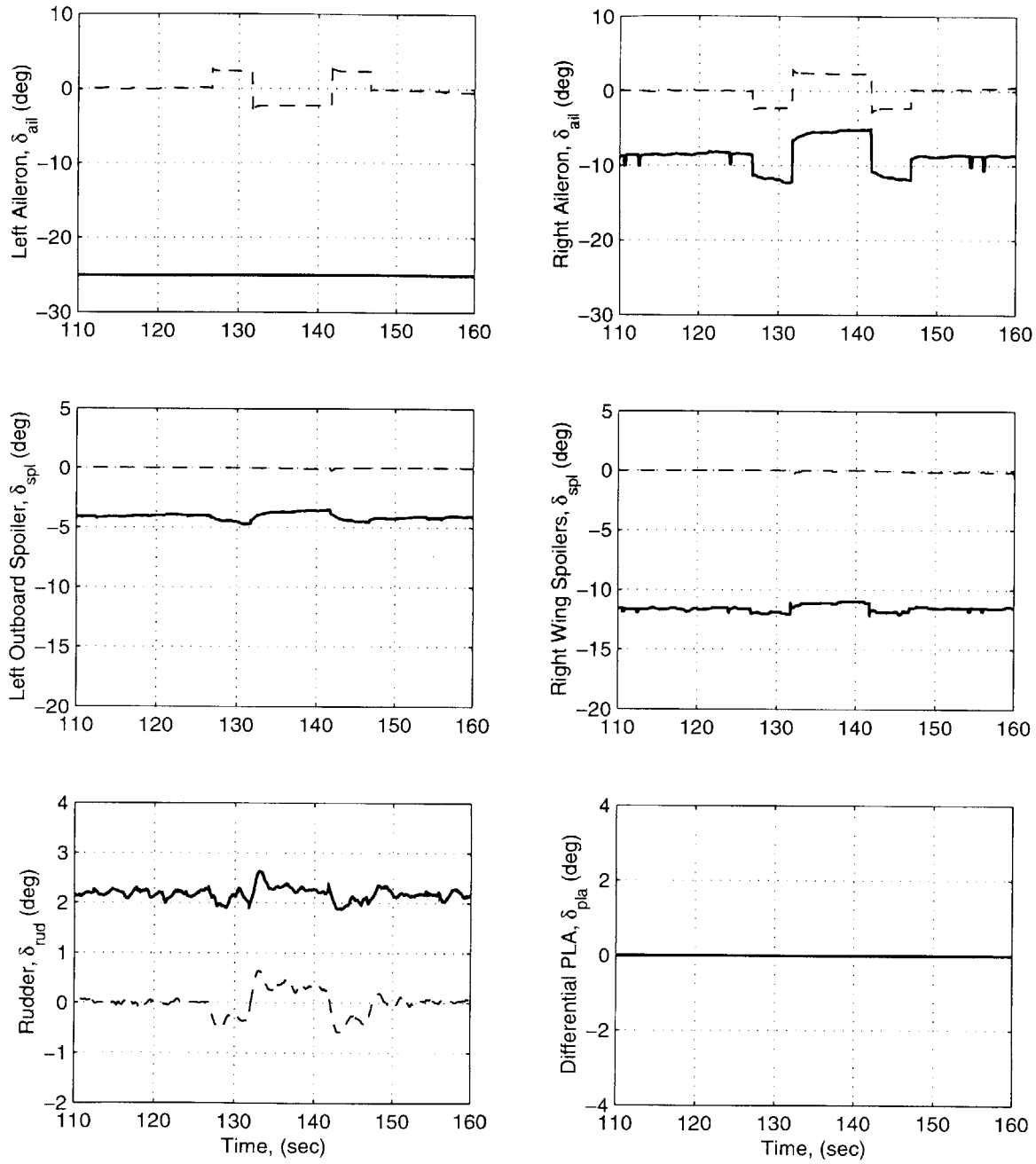

Figure 6-36: Surface Deflection Comparison of Aileron Failure with RCM to Nominal Performance, $\delta_{ail_l} = -25.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

A comparison between the post-failure and nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-37. It is seen from this figure that within the viable frequency range of the input signals, the frequency responses match.

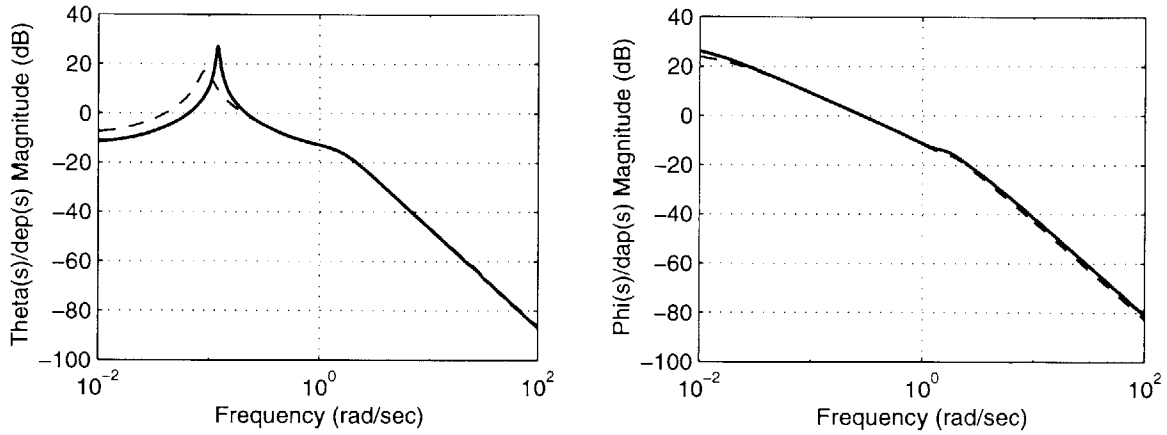

Figure 6-37: Comparison of Aileron Failure with RCM to Nominal Frequency Response, $\delta_{ail_l} = -25.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

Next, the post-failure flying qualities are presented in Table 6.21. It is seen from this table that the post-failure aircraft with RCM has Level 1 flying qualities except for the Phugoid damping which has a Level 2 rating. Additionally, the post-failure modes are close to the nominal modes.

Table 6.21: Post-Failure Versus Nominal Aircraft Flying-Qualities for Aileron Failure: $\delta_{ail_l} = -25.0^o$, 0.82 M, 35,000 ft, Class III, Category B

Mode	Metric	Nom Value	Fail Value	Level
Short	ζ_{sp}	0.71	0.63	1
Period	$\omega_{n_{sp}}$	1.63 rad/sec	1.63	1
Phugoid	ζ_p	0.11	0.04	1
Spiral	T_{2_s}	Stable	Stable	1
Dutch Roll	ζ_d	0.28	0.33	1
	$\zeta_d\omega_{n_d}$	0.52 rad/sec	0.54	1
	ω_{n_d}	1.84 rad/sec	1.63	1
Roll	T_r	0.45 sec	0.48	1

Rudder Fixed Position Failure

The final failure scenario that will be presented is a fixed rudder failure. At 5.0 seconds into the simulation, the rudder is move instantaneously from the trim setting — which is approximately 0.0^o — to 4.0^o. Figures 6-38 and 6-39 illustrate the nominal performance and failure performance with and without reconfiguration. It is seen from Figure 6-38 that without reconfiguration, the

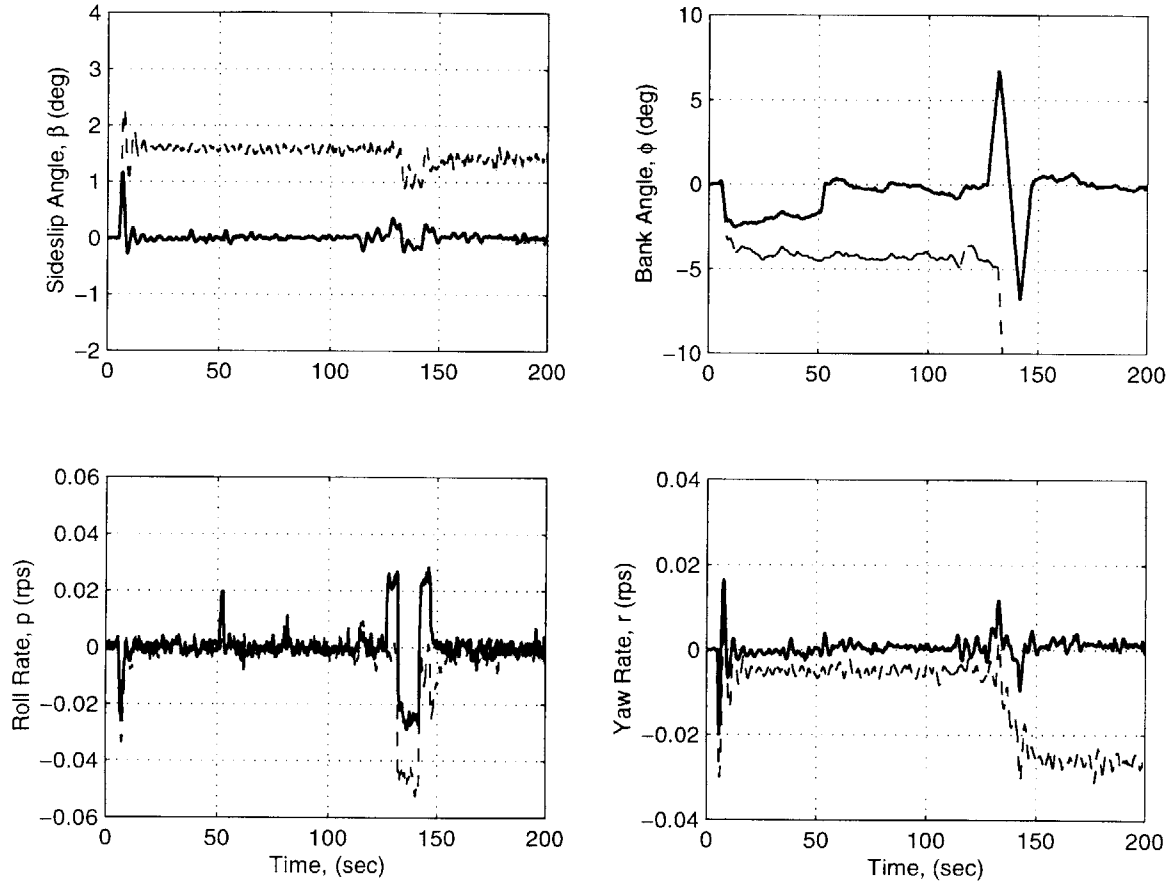

Figure 6-38: Rudder Failure, $\delta_{rud} = 4.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o RCM)

aircraft departs in roll attitude. Initially, the inherent robustness in the lateral axis captures the induced rolling moment generated by the rudder failure, i.e. $p(t) \rightarrow 0$. However, the nominal control system does not have inherent robustness in the directional axis, and the yaw rate is not attenuated. Then, when the roll doublet is attempted, the aircraft departs in roll.

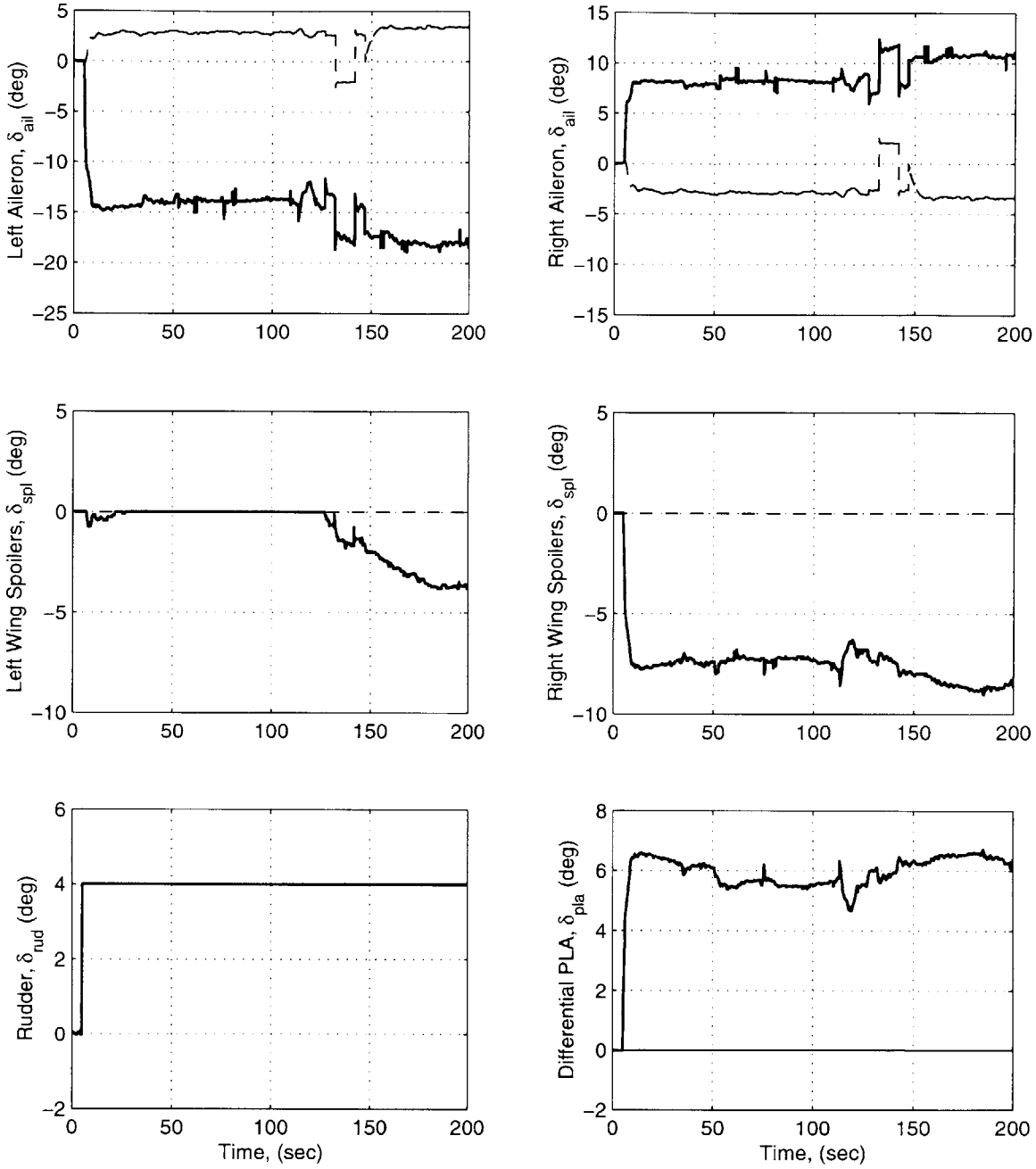

Figure 6-39: Surface Deflections for Rudder Failure, $\delta_{rud} = 4.0^o$, 0.82 M, 35,000 ft (— RCM, - - w/o RCM)

With reconfiguration, the failure disturbance is captured by using the split flap configuration and differential engines to attenuate the yaw rate error and spoilers to attenuate the roll rate error. Note, these secondary yaw devises provide the yaw damping since the entire rudder is inoperative. The maximum bank angle produce by the failure is approximately 3.0^o, and the maximum sideslip angle is 1.1^o. After the failure disturbance is captured, the bank angle slow decreases between the time interval $20 \rightarrow 50$ seconds. At 50 seconds, the pilot levels the wings.

After the pilot levels the wings, open-loop pilot commands are given to provide a comparison between post-failure performance and nominal performance. Figures 6-40 and 6-41 illustrate the comparison.

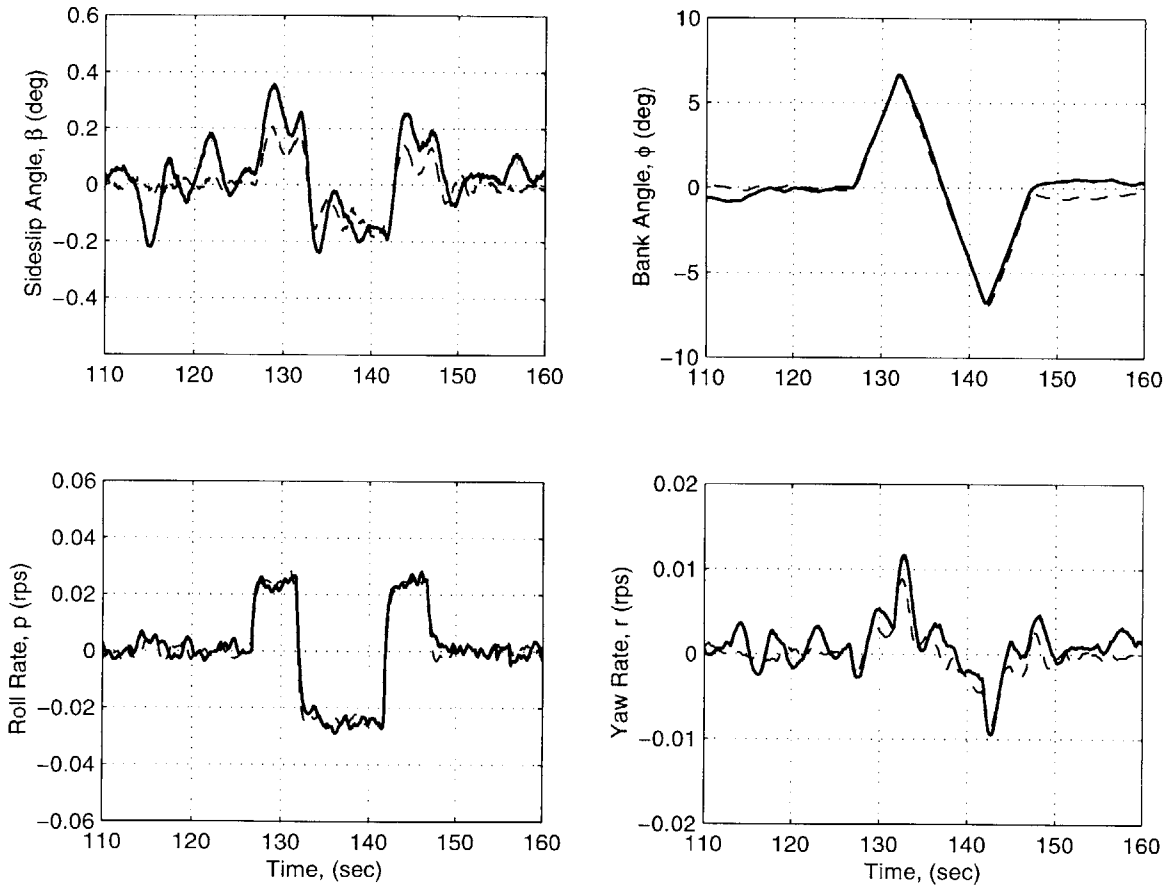

Figure 6-40: Comparison of Rudder Failure with RCM to Nominal Performance, $\delta_{rud} = 4.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

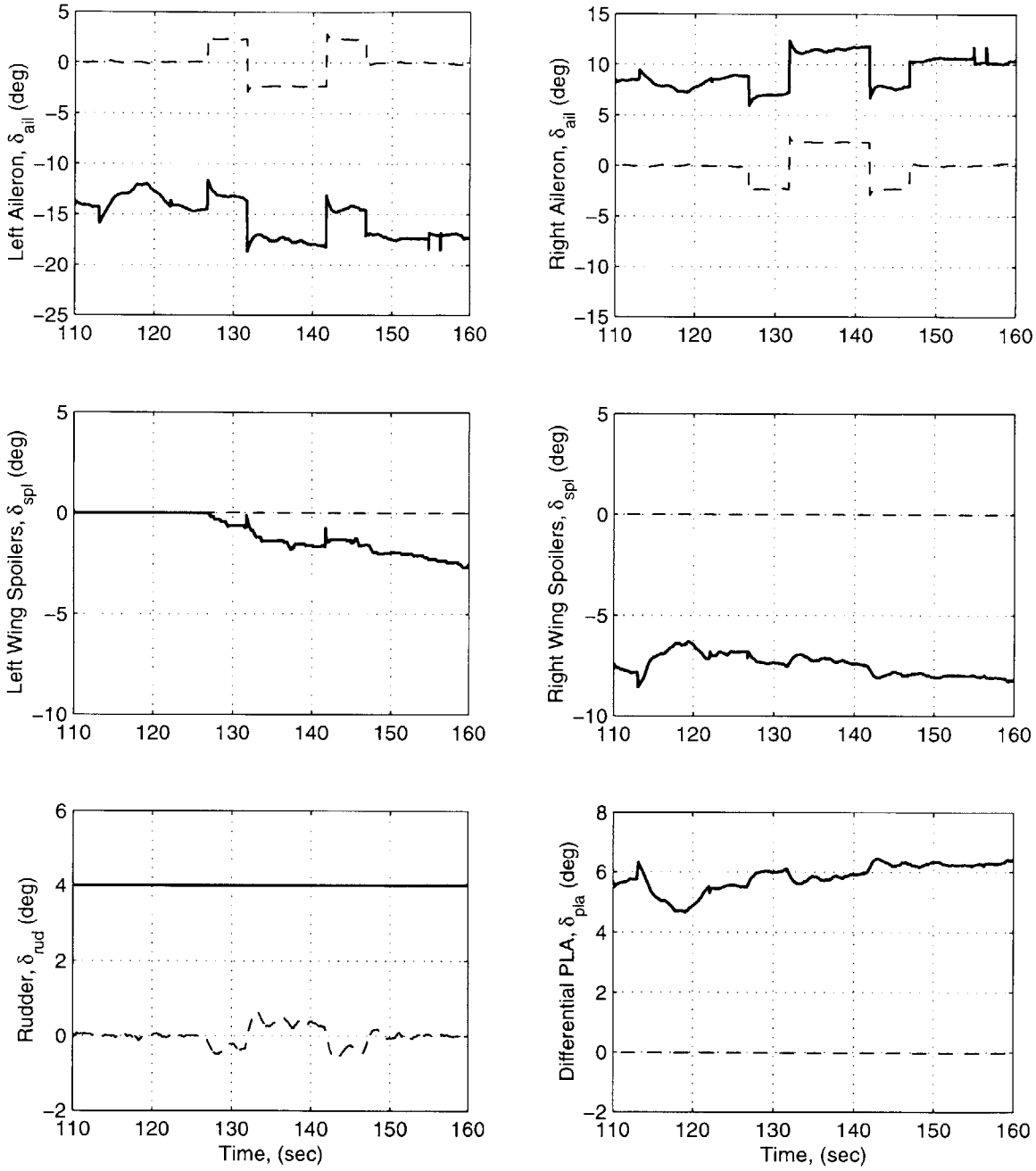

Figure 6-41: Surface Deflection Comparison of Rudder Failure with RCM to Nominal Performance, $\delta_{rud} = 4.0^o$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

It is seen from Figure 6-40 that for the roll stick doublet, near nominal performance is achieved post-failure. The only deficiency is that the Dutch Roll mode has slightly lower damping. This lower damping can be view by observing the yaw rate response.

A comparison between the post-failure and nominal frequency responses from pitch and roll stick to pitch and bank angle respectively are illustrated in Figure 6-42. It is seen from this figure that within the viable frequency range of the input signals, the frequency responses match.

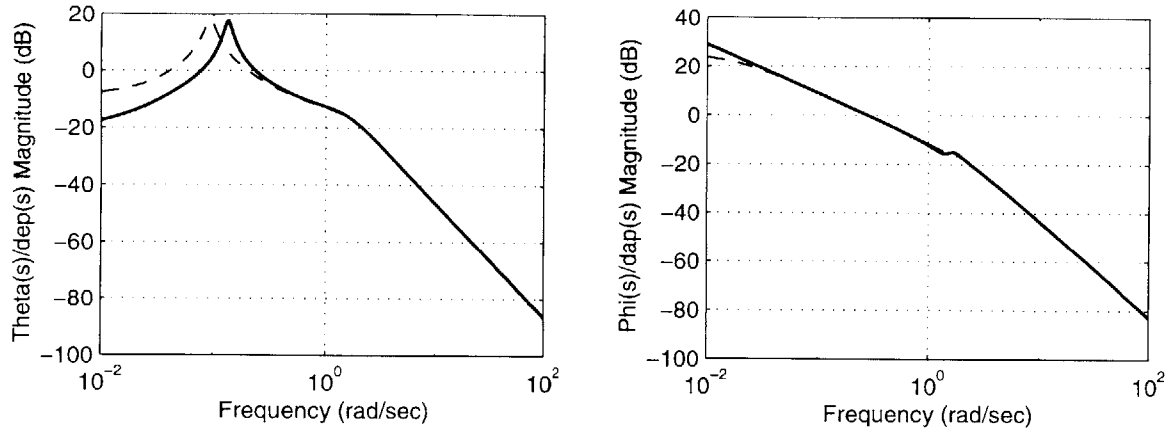

Figure 6-42: Comparison of Rudder Failure with RCM to Nominal Frequency Response, $\delta_{rud} = 4.0^{o}$, 0.82 M, 35,000 ft(— RCM, - - Nominal)

Next, the post-failure flying qualities are presented in Table 6.22. It is seen from this table that the post-failure aircraft with RCM has Level 1 flying qualities. Additionally, most of the post-failure modes are close to the nominal modes with the exception of the Dutch Roll mode which has a lower frequency and damping.

Table 6.22: Post-Failure Versus Nominal Aircraft Flying-Qualities for Rudder Failure: $\delta_{rud} = 4.0^o$, 0.82 M, 35,000 ft, Class III, Category B

Mode	Metric	Nom Value	Fail Value	Level
Short Period	ζ_{sp}	0.71	0.68	1
	$\omega_{n_{sp}}$	1.63 rad/sec	1.60	1
Phugoid	ζ_p	0.11	0.08	1
Spiral	T_{2_s}	Stable	Stable	1
Dutch Roll	ζ_d	0.28	0.16	1
	$\zeta_d\omega_{n_d}$	0.52 rad/sec	0.26	1
	ω_{n_d}	1.84 rad/sec	1.61	1
Roll	T_r	0.45 sec	0.45	1

6.4 Conclusions

This chapter presented a detailed description of the implementation of the RCM architecture in the Advanced Concept Flight Simulator (ACFS). The ACFS is a high fidelity, six degree-of-freedom flight simulator that represents a generic narrow-body commercial transport aircraft. This implementation and the accompanying results demonstrate the RCMs potential for accommodating a wide variety of control surface failures. Key issues associated with commercial transport reconfiguration were highlighted. Furthermore, simulation results illustrated that the RCM was able to provide nominal performance post-failure, without corrective pilot inputs, provided that sufficient control effectiveness existed post-failure. Without the RCM architecture, most failures presented resulted in aircraft departures.

The implementation and simulation results of the recursive reference model identification were presented. It was shown that the two-stage on-line identification procedure was able to generate a reference model representative of the aircraft's nominal performance. The implementation highlighted issues associated with identification of commercial transports. The identification algorithm was tuned so that unbiased estimates were obtained for various excitation conditions. Furthermore, the implementation also illustrated that during normal operation, some additional control surface excitation is required to generate a complete estimate of the reference model.

The implementation of the adaptive augmentor component illustrates the challenge that commercial transport reconfiguration presents. In both the longitudinal and directional axis, the control surfaces are not well balanced from a reconfiguration viewpoint, i.e. in the longitudinal axis, there is no surface with equivalent bandwidth to the elevator, and in the direction axis, there is no surface with equivalent control authority to the rudder. Thus, for failure to either the elevator or rudder, the reconfiguration task is difficult. As a result, a novel reconfiguration control allocation scheme was devised that blends in all the control effectors in a given axis to perform the reconfiguration task. In the longitudinal axis, the reconfiguration task was accomplished by combining the elevator, horizontal tail, and symmetrical throttles. In the lateral axis, differential spoiler and aileron deflections were used, and in the directional axis, a combination of rudder, split flap (spoilers and ailerons geared to produce pure yawing moment), and differential

throttles were used to accommodate failures.

Simulation results that compared the nominal aircraft to a failed aircraft with and without reconfiguration at a cruise flight condition were presented. These results illustrated the RCMs potential for commercial transport reconfiguration. A variety of single event, fixed position control surface failures were illustrated: elevator, horizontal tail, spoiler, aileron, and rudder. The results demonstrate that when the aircraft had adequate control authority post-failure, nominal performance was recovered. Without the RCM, the aircraft departed for most of the failure scenarios presented. Post-failure performance was assessed by comparing time histories, frequency response functions, and aircraft handling qualities post-failure to nominal. For failures where nominal performance is infeasible due to lack of control authority, the RCM architecture still provided a means of controlling the aircraft with traditional pilot stick inputs. For these situations, the reconfiguration task was performed in a stable manner while in the presence of significant control surface saturation. The primary reason for the lack of control authority post-failure is that the ACFS has a single actuator for the elevator and a single actuator for the rudder. If this aircraft had a split elevator and rudder, the potential for nominal performance for a failure to one of these actuators exists.

Simulation results also illustrated how the RCM interacts with the nominal control system during post-failure operations. If the nominal control has inherent robustness to certain failures, the RCM will only inject signals that are required to obtain nominal performance. Thus, if the nominal control system has a high level of robustness to a particular failure, the RCM will do nothing. On the other hand, if the nominal control system has a low level of robustness, the RCM will perform most of the reconfiguration task. Furthermore, simulation results also illustrated how the RCM interacts with the autopilot. The nominal control system defaults to an attitude hold mode for pitch stick-free operations, and the control system generates elevator commands to perform this function. For failures to the elevator, the elevator attitude hold commands are translated into equivalent pitch stick inputs which are then fed to the RCM. As a result, the RCM performs the attitude hold function.

In all of the failure scenarios presented, the reconfiguration was performed rapidly without

requiring the pilot to assess the failure status and or take corrective action before the aircraft enters an unrecoverable attitude. Even if the pilot was able assess the failure status rapidly, the pilot controls do not permit the appropriate interface to accommodate the failures since the pilot cannot command differential spoiler or split flap deflections.

THIS PAGE LEFT BLANK INTENTIONALLY

Chapter 7

Summary, Contribution, and Recommendations

7.1 Summary

The objective of this research was to investigate reconfiguration strategies that would improve aviation safety while adhering to real world constraints. This objective was accomplished. The motivation for this research originates from the NASA safety objective of reducing the fatal accident rate. These simple words were translated into a reconfiguration architecture that is composed of several algorithms. Both analytical and numerical analysis were performed to demonstrate the reconfiguration potential in the event of a control surface failure. The results revealed that nominal performance was achievable for a wide variety of control surface failures if there was adequate control authority post-failure.

In 1997, NASA stated the objective to reduce the fatal accident rate in commercial aviation by 80% in 10 years, and 90% in 20 years. Since the fatal accident rate is already small, many aspects of the U.S. flight transportation fleet must be improved to achieve this objective. One important aspect is reconfiguration: an estimated 14% of fatal incidents could have been prevented through accommodating damaged or inoperative control surfaces. Therefore, reconfigurable flight control

is an integral aspect of the overall safety plan. However, there are many obstacles to adding reconfiguration logic to the commercial fleet. A primary obstacle is that a significant fraction of the fleet in 10-20 years will consist of current aircraft; complex, all-encompassing changes to the flight control laws of these vehicles is a costly proposition. Furthermore, any new technology requiring significant changes to flight control design methodology is unlikely to be accepted by industry. Considering these obstacles, the focus of the research presented here was to develop reconfiguration strategies that can be added to existing flight control systems, keeping cost and complexity low.

The types of failures that were considered were failures to the control surfaces. The failure set included unanticipated, single or multiple, simultaneous or sequential failures that affect the control power of the aircraft, that may affect the baseline aerodynamics, and that may produce large disturbances. The failure set excludes those unsolvable areas where the aircraft cannot be saved, i.e, a static equilibrium must exist and there must be enough control power to stabilize the aircraft.

The reconfiguration framework proposed is completely separate from the control law and takes input information about the state of the vehicle, command inputs and control law outputs; it then generates an augmentation signal that compensates for deficiencies in the basic control laws. Given this framework, an extensive literature survey was conducted to identify potential technologies. Given the advantages and disadvantages of the available technologies, a low-cost retrofit reconfiguration module for commercial transports must meet several requirements. There are architectural requirements: independent of existing control structure, minimize off-line analysis, exhibit nominal performance without failures, applicable to a variety of aircraft, and provide adequate command following post failure. There are also algorithm requirements: ensure stability post failure, accommodate unforeseen failures, adapt quickly to destabilizing forces, and account for control surface constraints.

Given the requirements and the available technologies, a direct adaptive approach using a model reference framework was selected for the retrofit framework. The architecture consisted of an adaptive augmentor, flight condition dependent reference model, and signal conditioning

and estimation components. The theoretical development of each of these components are detailed in the methodology chapters of the thesis. This development was then followed by the implementation chapters that illustrates the implementation of the retrofit architecture on two separate aircraft simulators: the F/A-18 and the Advanced Concept Flight Simulator (ACFS), which is a generic narrow-body commercial transport.

The adaptive augmentor component is the primary algorithm that determines the reconfiguration strategy. A direct adaptive, input error approach that incorporates input saturation in the adaptive control design was adopted for the adaptive augmentor component. This formulation permits rapid adaptation and can accommodate hard-over failures. Additionally, a proof of stability existed for a SISO, open-loop plant with input saturation. Several extensions were required. The algorithm was formulated for close-loop, retrofit architecture that tracked the desired trim condition. Additionally, methods were included to handle control surface position and rate saturation. Finally, an existing proof of stability for SISO, open-loop plants was extended to include a MIMO, closed loop formulation with rate and position saturation while accounting for modeling error, sensor, and process noise.

Given that this is a model reference control framework, a reference model is required. The reference model component provides the desired dynamics of the healthy aircraft to the adaptive augmentor component. The accuracy of the reference model is the key issue associated with the operational performance. Inaccuracies in the reference model will lead to false alarms and ultimately limit the achievable post-failure performance. As a result, the reference model must be flight condition dependent. Given the additional requirements for low-cost implementation and life cycle maintenance, the reference model is to be generated on-line using system identification. To achieve these goals, a two stage identification process is outlined which is well suited for on-line identification of the desired dynamics. The foundation of this technique is based on the minimum variance estimator and is composed of both batch and sequential algorithms. This algorithm is the main part of the overall architecture that takes a mature military technology and transitions it to commercial applications.

The signal conditioning and estimation module is the final component of the reconfiguration

architecture. Its primary function is to provide the required signals to the adaptive augmentor and reference model components. Several assumptions were made in the previous components about the availability of signals. Given that some of this signals are not available on existing commercial transports, these previous simplifications result in greater complexity for the signal conditioning and estimation component. Depending on the available sensors suites, Kalman filters and control surface models may be required to generate all of the signals required to perform the reconfiguration task.

With these components, the retrofit architecture was implemented on two separate aircraft. The first aircraft was the F/A-18 nonlinear simulator that resides at MIT. The purpose of this implementation was to provide a proof of concept of the modular nature of the architecture and to provide initial simulation results. A wide variety of control surface failures were performed and reported: rudder fixed-position failure, stabilator fixed position failure, stabilator floating failure. Comparisons were made between the failed aircraft with and without reconfiguration and between the failed-with-reconfiguration and the healthy aircraft. Simulation revealed that the retrofit module was able to stabilize the aircraft in the presence of large failure disturbances and control surface saturation. Furthermore, nominal performance was achieved if there was adequate control authority post-failure. Without reconfiguration, the aircraft departs for the failures considered.

The second aircraft used was the Advanced Concept Flight Simulator (ACFS) that resides at NASA Dryden Flight Research Center. The ACFS is a high fidelity, six degree-of-freedom flight simulator that represents a generic narrow-body commercial transport aircraft. This implementation and the accompanying results demonstrated the retrofit module's potential for accommodating a wide variety of control surface failures. Key issues associated with commercial transport reconfiguration were highlighted. Furthermore, simulation results illustrated that the retrofit module was able to provide nominal performance post-failure, without corrective pilots inputs, provided that sufficient control effectiveness existed post-failure. Without the reconfiguration architecture, most failures presented resulted in aircraft departure.

The implementation and simulation results of the recursive reference model identification were

presented. It was shown that the two-stage on-line identification procedure was able to generate a reference model representative of the aircraft's nominal performance. The implementation highlighted issues associated with identification of commercial transports. The identification algorithm was tuned so that unbiased estimates were obtained for various excitation conditions. Furthermore, the implementation illustrated that during normal operation, some additional control surface excitation is required to generate a complete estimate of the reference model.

The implementation of the adaptive augmentor component illustrates the challenge that commercial transport reconfiguration presents. In both the longitudinal and directional axis, the control surfaces are not well balanced from a reconfiguration viewpoint, i.e. in the longitudinal axis, there is no surface with equivalent bandwidth to the elevator, and in the directional axis, there is no surface with equivalent control authority to the rudder. Thus, for failure to either the elevator or rudder, the reconfiguration task is difficult. As a result, a novel reconfiguration control allocation scheme was devised that blends in all the control effectors in a given axis to perform the reconfiguration task. In the longitudinal axis, the reconfiguration task was accomplished by combining the elevator, horizontal tail, and symmetrical throttles. In the lateral axis, differential spoiler and aileron deflections were used, and in the directional axis, a combination of rudder, split flap (spoilers and ailerons geared to produce pure yawing moment), and differential throttles were used to accommodate failures.

Simulation results that compared the nominal aircraft to a failed aircraft with and without reconfiguration at a cruise flight condition were presented. These results illustrated the retrofit module's potential for commercial transport reconfiguration. A variety of single event, fixed position control surface failures were illustrated: elevator, horizontal tail, spoiler, aileron, and rudder. The results demonstrated that when the post failure aircraft had adequate control authority post-failure, nominal performance was recovered. Without the retrofit module, the aircraft departed for most of the failure scenarios presented. Post-Failure performance was assessed by comparing time histories, frequency response functions, and aircraft handling qualities post-failure to nominal. For failures where nominal performance is infeasible due to the lack control authority, the reconfiguration architecture still provided a means of controlling the aircraft with traditional pilot stick inputs. For these situations, the reconfiguration task was performed

in a stable manner in the presence of significant control surface saturation. The primary reason for the lack of control authority post-failure is that the ACFS has a single actuator for the elevator and a single actuator for the rudder. If this aircraft had a split elevator and rudder, the potential for nominal performance for a failure to one of these actuators exists.

Simulation results also illustrated how the retrofit module interacts with the nominal control system during post-failure operations. If the nominal control has inherent robustness to certain failures, the retrofit module will only inject signals that are required to obtain nominal performance. Thus, if the nominal control system has a high level of robustness to a particular failure, the retrofit module will do nothing. On the other hand, if the nominal control system has a low level of robustness, the retrofit module will perform most of the reconfiguration task. Furthermore, simulation results also illustrated how the retrofit module interacts with the autopilot. The nominal control system defaults to an attitude hold mode for pitch stick-free operations, and the control system generates elevator commands to perform this function. For failures to the elevator, the elevator attitude hold commands are translated into equivalent pitch stick inputs which are then fed to the retrofit module. As a result, the retrofit module performs the attitude hold function.

In all of the failure scenarios presented, the reconfiguration was performed rapidly without requiring the pilot to assess the failure status and then hopefully take the corrective action before the aircraft entered an unrecoverable attitude. Even if the pilot was able assess the failure status rapidly, the pilot controls do not permit the appropriate interface to accommodate the failures since the pilot cannot command differential spoiler or split flap deflections.

To summarize the conclusions:

- Given the requirements and the available technologies, a direct adaptive approach using a model reference framework is well suited for the retrofit framework. The architecture consists of an adaptive augmentor, flight condition dependent reference model, and signal conditioning and estimation components.
- An on-line, two stage identification process can identify low-order equivalent desired dynamics of a healthy aircraft.
- Depending on the available sensors suites, Kalman filters and control surface models can generate all of the required signals for the reconfiguration task.
- The retrofit architecture can provide reconfiguration functionality to a variety of aircraft.

The module was implemented in a F/A-18 nonlinear simulator and in a generic commercial transport nonlinear simulator. Simulation results revealed that the retrofit module is able to stabilize the aircraft in the presence of large failure disturbances and control surface saturation using normal stick inputs. Furthermore, nominal performance is achieved if there is adequate control authority post-failure.

- Due to the lack of control surface redundancy, commercial transport reconfiguration is a difficult task. In both the longitudinal and directional axes, the control surfaces are not well balanced from a reconfiguration viewpoint, i.e. in the longitudinal axis, there is no surface with equivalent bandwidth to the elevator, and in the directional axis, there is no surface with equivalent control authority to the rudder. Thus, for failure to either the elevator or rudder, the reconfiguration task is difficult. Given this difficulty, a novel reconfiguration control allocation scheme was devised that blends in all the control effectors in a given axis to perform the reconfiguration task.
- The retrofit system does not cancel the nominal control system, but instead interacts minimally to perform the reconfiguration task. If the nominal control has inherent robustness to certain failures, the retrofit module will only inject signals that are required to obtain nominal performance.

7.2 Contributions

This research makes a series of contributions that range from architectural design to extensions of theoretical development in the current literature to the implementation of this reconfiguration approach on two aircraft. Thus, the research is well balanced from a theoretical and implementation perspectives.

The first contribution is that this research addresses the requirement to improve aviation safety. Simple requirements to reduce the fatal accident rate were translated into straight forward algorithms that achieve that task. In doing so, a retrofit architecture was developed that encompassed the objectives of low-cost implementation while providing rapid, automatic reconfiguration functionality to existing aircraft. This reconfiguration architecture satisfies the requirements of being independent of existing control structure, exhibit nominal performance without failures, applicable to a variety of aircraft, and provide adequate command following post failure.

The second contribution is that existing algorithms in the literature were extended for the retrofit architecture, and mature technology was incorporated where appropriate. The adaptive augmentor component uses a direct, input error adaptive approach that can accommodate

unknown failures. This formulation was extended for a closed-loop, non-intrusive format that tracks the desired trim conditions and accounts for control surface constraints. An existing proof of stability was then extended to include MIMO, closed loop model with both rate and position saturation. The reference model component consists of a flight condition dependent, low-order equivalent model of the desired dynamics. An innovative model structure was developed that attempts to maximize the achievable performance post failure while minimizing false alarms. An on-line system identification algorithm is used to generate the desired dynamics. This algorithm represents the part of the architecture where a mature military technology is transitioned for commercial applications.

The final contribution involves the implementation of the retrofit module in two separate high-fidelity simulators. The simulation results supplemented the theoretical development and provide graphical evidence of the reconfiguration potential. Demonstrating this on two different classes of aircraft verifies that this approach is applicable to a wide variety of aircraft. Additionally, the implementations uncovered paramount issues associated with commercial transport reconfiguration which are not available in the current literature. Furthermore, using realistic, high-fidelity, six degree-of-freedom simulators, the implementation demonstrated that this approach does work.

Obviously, we have not exhaustively tested all failure scenarios that one may pose, i.e. slow time varying failures, actuator polarity failures, high-lift device failures, or missing surface failures. Given that instantaneous hard-over failures are in most cases more severe than these examples, it is the author's opinion that this reconfiguration approach can accommodate the other failure types provided that a static equilibrium condition exists and that adequate control authority is available post-failure. It was for this reason that hard-over type failure were selected for evaluation. At most, minor modifications would be required to accommodate a substantial subset of all possible control surface failures. In other words, conceptually, there are no assumptions in the architecture that precludes the accommodation of these other failure types. More testing is required to substantiate this opinion.

7.3 Recommendations

It is the recommendation of the author that the retrofit architecture needs to be evaluated in a piloted simulator, preferably a FAA certified commercial transport simulator. Issues associated with algorithm computational requirements and storage requirements need to be resolved. Additionally, it would be advantageous to have an impaired aircraft make a controlled descent and landing. Issues associated with pilot evaluation, hardware integration, and software integration need to be addressed before proceeding with a flight test program.

Additionally, for aircraft without a split rudder, it would be advantageous to devise a method where the secondary yaw devices just provide yaw damping without trying to zero the sideslip angle. Thus, the inherent weathercocking ability of the aircraft would be used to capture the failure disturbance in the directional axis. A method for implementing this approach in the adaptive control architecture is unknown at this time. Attempts were made to redefine the desired trim Cf_m using the linear nominal model for a given fixed position rudder failure. This attempt failed because the nominal model does not approximate the crabbed flight condition well. Thus, it might be wise to remove the desired trim reconfiguration gain K_f in the directional axis.

Finally, it might be advantageous to extended the stability proof present herein for the case of an unstable aircraft with augmentation, i.e. a failure causes the augmented aircraft to be unstable. As background information, there are four cases that we are concerned about: stable aircraft without a nominal control system, unstable aircraft without a nominal control system, stable aircraft with a nominal control system, and unstable aircraft with a nominal control system. Stability proofs exist for the first three cases, and there is no apparent extension available for the fourth case. The engineering benefit of developing the proof is questionable for commercial transport reconfiguration since the Phugoid and spiral modes have time constants in the order of 200 seconds. However, developing this proof might provide a valuable extension to the existing stability analysis literature for closed-loop plants with input saturation.

THIS PAGE LEFT BLANK INTENTIONALLY

Appendix A

Derivation of the Stability Axis Nonlinear Moment Equations

The stability axis nonlinear moment equation will be derived from the body axis equations. The derivation follows the notation contained in Stevens and Lewis.The conversion of the body-axis equations to stability axis, one equation at a time, is time consuming and lengthy. As mentioned in Stevens and Lewis, an alternative approach is to use the vector-matrix representations using rotation matrices S. The coordinate transformation S will be presented below. We begin with the flat-earth, body axis moment equations in vector-matrix format (Equation 1.5-4 in [71]):

$$\dot{\omega}_B = -J^{-1}\Omega_B J\omega_B + J^{-1}T_B \tag{A.1}$$

where

$$\omega_B = \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \quad J = \begin{bmatrix} J_x & 0 & -J_{xz} \\ 0 & J_y & 0 \\ -J_{xz} & 0 & J_z \end{bmatrix} \quad J^{-1} = \frac{1}{\Gamma} \begin{bmatrix} J_z & 0 & J_{xz} \\ 0 & \frac{\Gamma}{J_y} & 0 \\ J_{xz} & 0 & J_x \end{bmatrix}$$

$$T_B = \begin{bmatrix} L_B \\ M_B \\ N_B \end{bmatrix} \quad \Omega_B = \begin{bmatrix} 0 & -R & Q \\ R & 0 & -P \\ -Q & P & 0 \end{bmatrix} \quad \Gamma = J_x J_z - J_{xz}^2$$

and the subscript implies body-axis.

Recalling the definitions of body and stability axis systems, the conversion from body to stability-axis is simply a rotation about the aircraft's y-axis by the angle of attack α. The coordinate transformation is thus defined as follows:

$$V_s = S_\alpha V_B \quad \text{where} \quad S_\alpha = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{A.2}$$

where the subscript S refers to stability-axis coordinate system. Also, for the relative transformation between two coordinate frames with the same origin, the strapdown equation is useful:

$$\Omega_R = S_\alpha \dot{S}_\alpha^T = \begin{bmatrix} 0 & 0 & -\dot{\alpha} \\ 0 & 0 & 0 \\ \dot{\alpha} & 0 & 0 \end{bmatrix} \tag{A.3}$$

This relationship can be obtained by taking the derivative of (A.2), performing the matrix multiplication, and using trigonometric identities.

We begin the derivation with the body-axis moment equation expressed as follows, which is

(A.1) with the cross product expanded:

$$J\frac{d}{dt}(\omega_B) + \omega_B \times J\omega_B = T_B \tag{A.4}$$

Premultiply (A.4) through by S_α in (A.2):

$$S_\alpha J\frac{d}{dt}(\omega_B) + S_\alpha(\omega_B \times J\omega_B) = S_\alpha T_B \tag{A.5}$$

Now, the following substitutions can be made:

$$\begin{aligned}
T_B &= S_\alpha^T T_S \\
\omega_B &= S_\alpha^T \omega_S \\
S_\alpha(\omega_B \times J\omega_B) &= S_\alpha\omega_B \times S_\alpha\omega_B J\omega_B = \omega_S \times S_\alpha J S_\alpha^T \omega_S
\end{aligned}$$

Where in the last equation, we used the fact that the coordinate transformations are distributive with the cross-product. Making these substitutions, the moment equation (A.5) becomes:

$$S_\alpha J\frac{d}{dt}(S_\alpha^T\omega_S) + \omega_S \times (S_\alpha J S_\alpha^T)\omega_S = T_S \tag{A.6}$$

The derivative term in the above equation can be simplified using the strapdown equation (A.3).

$$\begin{aligned}
S_\alpha J\tfrac{d}{dt}(S_\alpha^T\omega_S) &= S_\alpha J(S_\alpha^T\dot{\omega}_S + \dot{S}_\alpha^T\omega_S) = S_\alpha J S_\alpha^T(S_\alpha S_\alpha^T\dot{\omega}_S + S_\alpha\dot{S}_\alpha^T\omega_S) \\
&= (S_\alpha J S_\alpha^T)(\dot{\omega}_S + \Omega_R\omega_S)
\end{aligned}$$

Thus,

$$(S_\alpha J S_\alpha^T)(\dot{\omega}_S + \Omega_R\omega_S) + \omega_S \times (S_\alpha J S_\alpha^T)\omega_S = T_S \tag{A.7}$$

Define the stability axis inertial matrix as follows:

$$J_s = S_\alpha J S_\alpha^T \quad J_s = \begin{bmatrix} J_{x_s} & 0 & -J_{xz_s} \\ 0 & J_{y_s} & 0 \\ -J_{xz_s} & 0 & J_{z_s} \end{bmatrix} \quad J_s^{-1} = \frac{1}{\Gamma_s} \begin{bmatrix} J_{z_s} & 0 & J_{xz_s} \\ 0 & \frac{\Gamma - s}{J_{y_s}} & 0 \\ J_{xz_s} & 0 & J_{x_s} \end{bmatrix},$$

where $\Gamma_s = J_{x_s} J_{z_s} - J_{xz_s}^2$. The body-stability axis rotation of the inertias can be expanded to produce the following transformation matrix:

$$\begin{bmatrix} J_{x_s} \\ J_{z_s} \\ J_{xz_s} \\ J_{y_s} \end{bmatrix} = \begin{bmatrix} \cos^2\alpha & \sin^2\alpha & -\sin 2\alpha & 0 \\ \sin^2\alpha & \cos^2\alpha & \sin 2\alpha & 0 \\ \frac{1}{2}\sin 2\alpha & -\frac{1}{2}\sin 2\alpha & \cos 2\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} J_x \\ J_z \\ J_{xz} \\ J_y \end{bmatrix}$$

Substituting the stability axis inertias into the moment equations (A.7) we have:

$$J_S(\dot{\omega}_S + \Omega_R \omega_S) + \omega_S \times J_S \omega_S = T_S. \tag{A.8}$$

Now, the cross product can be replace with a matrix multiplication as follows:

$$\omega_S \times J_S \omega_S = \Omega_S J_S \omega_S \quad \text{where,} \quad \Omega_S = \begin{bmatrix} 0 & -R_s & Q_s \\ R_s & 0 & -P_s \\ -Q_s & P_s & 0 \end{bmatrix}.$$

Premultiplying through by the inverse of the stability-axis inertias and incorporating the simplification above, the moment equation is as follows:

$$\dot{\omega}_S + \Omega_R \omega_S + J_S^{-1} \Omega_S J_S \omega_S = J_S^{-1} T_S. \tag{A.9}$$

Knowing that the aerodynamic moments are a function of stability-axis states and the control surface deflections, the stability axis moments are generally modeled as a first order expansion of these dependencies:

$$J_S^{-1} T_S = CA_p x + CB_p u + Cf_p$$

where the elements of CA_p and CB_p are the flight condition dependent, dimensional stability derivatives defined in the stability axis system, Cf_p represents the aerodynamic intercept, x consist of the longitudinal and lateral-direction stability-axis states, and u is a vector of control surface deflections. Note, since the translation and kinematic equations are not included, the aerodynamics above represent a subset of the total open-loop aerodynamics. Since the above expression is the linear expansion of the aerodynamic moments, the aerodynamics are the same as the aerodynamics of the linearized equations of motion. Thus, combining the non-linear dynamics with the linear moment buildup, the non-linear, stability axis moment equations of motion are:

$$\dot{\omega}_S + \Omega_R \omega_S + J_S^{-1} \Omega_S J_S \omega_S = CA_p x + CB_p u + Cf_p \tag{A.10}$$

THIS PAGE LEFT BLANK INTENTIONALLY

Bibliography

[1] Shifrin, C. A., "Aviation Safety Takes Center Stage Worldwide," *Aviation Week & Sapce Technology*, pp. 46-48, November 4, 1996.

[2] Zemlyakov, S. D., Rutkovskii, V. Y., and Silaev, A. V., "Reconfiguring Aircraft Control Systems in Case of Failures," *Automation and Remote Control*, Vol. 57, No. 1, pp. 1-13, 1996.

[3] Eslinger, R. A., and Chandler, P. R., "Self-Repairing flight Control System Program Overview," *IEEE National Aerospace and Electronics Conference*, Vol. 2, pp. 504-511, 1988.

[4] Looze, D. P., Weiss, J. L., Eterno, J. S., and Barrett, N. M., "An Automatic Redesign Approach for Restructurable Control Systems," *IEEE Control Systems Magazine*, Vol. 5, No. 2, pp. 16-22, 1985.

[5] Bonnice, W. F., Wagner, E., Motyka, P., and Hall, S. R, "The Application of the Detection Filter to Aircraft Control Surface and Actuator Failure Detection and Isolation," *AIAA Guidance, Navigation and Control Conference*, pp. 732-740, 1985.

[6] Burken, J. J., and Burcham, F. W., Jr., "Flight-Test Results of Propulsion-Only Emergency Control System on MD-11 Airplane," *Journal of Guidance, Control and Dynamics*, Vol. 20, No. 5, pp. 980-7, September-October 1997.

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

[8] Dahleh, M. A., and Diaz-Bobillo, I. J., "Control of Uncertain Systems: A Linear Programming Approach," Prentice Hall, Upper Saddle River, NJ, 1996.

[9] Weiss, J. L., and J. Y. Hsu, "Integrated Restructurable Flight Control System Demonstration Results," NASA-CR-178305, ALPHATECH, Inc., Burlington, MA, 1987.

[10] Gao, Zhiqiang, "Techniques in Reconfigurable Control System Design," *Control and Dynamics Systems: Advances in Theory and Applications*, Vol. 79, pp. 89-115, 1996.

[11] Wagner, E., Burken, J. J., Hanson, C., and Wohletz, J. M., "Deterministic Reconfigurable Control Design for the X-33 Vehicle," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4413, Boston, MA, 1998.

[12] Wise, K. A. and Sedwick, J. L., "Stability Analysis of Reconfigurable and Gain Scheduled Flight Control Systems Using LMIs," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4111, pp. 118-26, Boston, MA, 1998.

[13] Chen, J., Patton, R. J., and Chen, Z., "An LMI Approach to Fault-Tolerant Control of Uncertain Systems," *Proceedings of teh 1998 IEEE ISIC/CIRA/ISAS Joint Conference* pp. 175-80, Gaithersburg, MD, September 14-17, 1998.

[14] Athans, M, Baram, Y., Castanon, D., Dunn, K. P., Green, C. S., Lee, W. H., Sandell, N. R. Jr., and Willsky, A. S., "Investigation of the Multiple Model Adaptive Control (MMAC) Method for Flight Control Systems," NASA CR 3089, MIT Electronic Systems Laboratory, MA, May, 1979.

[15] Maybeck, P., "Application of Multiple Model Adaptive Algorithms to Reconfigurable Flight Control," *Control and Dynamic Systems: Advances in Theory and Applications*, Vol. 52, pp. 291-320, 1992.

[16] Maybeck, P. S., and Stevens, R. D., "Reconfigurable Flight Control Via Multiple Model Adaptive Control Methods ," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 27, No. 3, pp. 470-480, May 1991.

[17] Ljung, L., "System Identification: Theory for the User," Second Edition, Prentice Hall PTR, Upper Saddle River, NJ, 1999.

[18] Mendel, J. M., "Discrete Techniques of Parameter Estimation," Marcel Dekker Inc., NY, 1973.

[19] Chandler, P. R., Pachter, M., and Mears, M., "System Identification of Adaptive and Reconfigurable Control," *Journal of Guidance, Control and Dynamics*, Vol. 18, No. 3, pp. 516-24, May-June 1995.

[20] Smith, Capt. L., Chandler, P. R., and Pachter, M., "Regulation Techniques for Real-Time Identification of Aircraft Parameters," *AIAA Guidance, Navigation, and Control Conference*, 1997.

[21] Ward, D. G., and Barron, R. L., "A Self-Designing Receeding Horizon Optimal Flight Controller," *American Control Conference,* pp. 3490-4, Seattle, WA, June, 1995.

[22] Ward, D. G., Monaco, J. F., Barron, R. L., Bird, R. A., Viring, J. C., and Landers, T. F., "Self-Designing Controler Design, Simulation, and Flight Test Evaluation," Contractor Final Report WL-TR-97-3095, November, 1996.

[23] Buffington, J., Chandler, P. R., and Pachter, M., "On-Line System Identification for Aircraft with Distributed Control Effectors," *AIAA Guidance, Navigation, and Control Conference*, 1997.

[24] Wohletz, J. M., "Parameter Estimation for the Tailless Advanced Fighter Aircraft (TAFA)," Final Report for Summer Graduate Research Program, Air Force Office of Scientic Research, August 1998.

[25] Elgersam, M., Enns, D., and Voulgaris, P.,, "Parameter Identification for Systems with Redundant Actuators," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4110, pp. 109-17, Boston, MA, 1998.

[26] Chandler, P. R., "Title unknown at this time," *AIAA Guidance, Navigation, and Control Conference*, AIAA-99-unknown, Portland, OR, 1999.

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

[28] Bodson, M., "An Adaptive Algorithm with Information-Dependent Data Forgetting ," *Proceedings of the American Control Conference*, pp. 3485-9, Seattle, WA, 1995.

[29] Ward, D. G., Monaco, J. F., and Bodson, M., 'Development and Flight Testing of a Parameter Identification Algorithm for Reconfigurable Control," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 6, pp 948-56, November-December 1998.

[30] Michalska, H., and Mayne, D. Q., "Robust Receding Horizon Control of Constrained Nonlinear Systems," *IEEE Transactions on Automatic Control*, Vol. 38, No. 11, pp. 1623-33, November, 1993.

[31] Mehra, R. K., "Aerospace Applications of Model Predictive Control," *SAE Aerospace Control and Guidance Systems Committee Meeting*, No.78, Nashville, TN, October 9-11, 1996.

[32] Pachter, M., Chandler, P. R., and Mears, M., "Reconfigurable Tracking Control with Saturation," *Journal of Guidance, Control and Dynamics*, Vol. 18, No. 5, pp. 1016-22, September-October, 1995.

[33] Chandler, P. R., Mears, M., and Pachter, M., "A Hybrid LQR/LP Approach for Addressing Actuator Saturation in Feedback Control," *Proceedings of the 33rd Conference on Decision and Control*, pp. 3860-7, Lake Buena Vista, FL, December, 1994.

[34] Ahmed-Zaid, F., Ioannou, P., Gousman, K., and Rooney, R., "Accommodation of Failures in the F-16 Aircraft Using Adaptive Control," *IEEE Control Systems Magazine*, Vol. 11, No. 1, pp. 73-8, January, 1991.

[35] Casalino, G, Ferrara, A., Minciardi, R., and Parisini, T., "Implicit Model Techniques and Their Application to LQ Adaptive Control," *Control and Dynamics Systems*, Vol. 79, pp. 347-73, 1996.

[36] Honeywell Technology Center, "Application of Multivariable Control Theory to Aircraft Control Laws," Final Contractor Report, WL-TR-96-3099, USAF, May, 1996.

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

[38] Napolitano, M. R., and Swaim, R. L., "New Techniques for Aircraft Flight Control Reconfiguration," *Control and Dynamic Systems: Advances in Theory and Applications*, Vol. 52, pp 155-228, 1992.

[39] Enns, D., Bugajski, D., Hendrick, R., and Stein, G., "Dynamic Inversion: An Evolving Methodology for Flight Control Design," *International Journal of Control*, Vol. 59, No. 1, pp. 71-91, 1994.

[40] Khalil, H. K., "Nonlinear Systems," 2nd Edition, Prentice Hall, Upper Saddle River, NJ, 1996.

[41] Slotine, J-J. E., Li, W., "Applied Nonlinear Control," Prentice Hall, Englewood Cliffs, NJ, 1991.

[42] Brinker, J. S., and Wise, K. A., "Stabiilty and Flying Qualities Robustness of a Dynamic Inversion Aircraft Control Law," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 6, pp. 1270-7, November-December, 1996.

[43] Bodson, M., and Pohlchuck, W. A., "Command Limiting in Reconfigurable Flight Control," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 4, pp. 639-46, July-August 1998.

[44] Gao, Z, and Antsaklis, P. J., "Reconfigurable Control System Design via Perfect Model Following," *International Journal of Control*, Vol. 56, No. 4, pp.783-98, October, 1992.

[45] Dhayagude, N., and Gao, Z., "Novel Approach to Reconfigurable Control Systems Design," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 4, pp. 963-6, July-August 1996.

[46] Caliskan, F., and Vepa, R., "A Reconfiguration Algorithm for Real-Time Aircraft Flight Control Systems," *Aerospace Vehicle Dynamics and Control*, pp.125-44, Clarendon Press, Oxford, 1994.

[47] Balakrishnan, S.N., and Biega, V., "Adaptive-Critic-Based Neural Networks for Aircraft Optimal Control," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 4, pp. 893-8, July-August 1996.

[48] Chandler, P. R., Mears, M., and Pachter, M., "On-Line Optimizing Networks for Reconfigurable Control," *Proceedings of the 32rd Conference on Decision and Control*, pp. 2272-7, San Antonio, TX, December, 1993.

[49] 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, pp. 1000-7, September-October, 1995.

[50] Durham, W. C., and Bordignon, K. A., "Multiple Control Effector Rate Limiting," *Journal of Guidance, Control, and Dynamics*, Vol. 19, no. 1, pp. 30-7, January-Febuary, 1996.

[51] Enns, D., "Control Allocation Approaches," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4109, pp. 98-108, Boston, MA, 1998.

[52] Buffington, J. M., "Tailless Aircraft Control Allocation," *AIAA Guidance, Navigation, and Control Conference*, 1997.

[53] Narendra, K. S., and Annaswamy, A. M., "Stable Adaptive Systems," Prentice Hall, 1989.

[54] Åstrom, K.J. and Wittenmark, B., "Adaptive Control," Addison-Wesley, 1989.

[55] Karason, S. P. and Annaswamy, A. M., "Adaptive Contro in the Presence of Input Contraints," *IEEE Transactions of Automaitc Control*, Vol. 39, No. 11, pp. 1-6, 1994.

[56] Annaswamy, A. M., Wong, J., "Adaptive Control in the Presence of Saturation Non-Linearity," *International Journal of Adaptive Control and Signal Processing*, Vol. 11, pp. 3-19, 1997.

[57] Annaswamy, A. M. and Karason, S. P., "Discrete-Time Adaptive Control in the Presence of Input Contraints," *Automatica*, Vol. 31, No. 10, pp. 1421-31, 1995.

[58] Narendra, K. S., and Parthasarthy, K., "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Transactions on Neural Networks*, Vol. 1, pp.4-27, March, 1990.

[59] Narendra, K. S., "Workshop Number 4: Neural Networks for Intelligent Control," *American Control Conference*, Albuquerque, NM, 1997.

[60] Troudet, T., Garg, S., and Merill, W., "Neurocontrol Design and Analysis for a Multivariable Aircraft Control Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, pp. 738-47, July-August 1993.

[61] Kim, B. S., Calise, A. J., "Nonlinear Flight Control Using Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, pp. 26-33, January-February 1997.

[62] McFarland, M. B., Calise, A. J., " Neural Networks for Stable Adaptive Control of Air-To-Air Missiles," *AIAA Guidance, Navigation, and Control Conference*, AIAA-95-3315, pp. 1280-5, Baltimore, MD, 1995.

[63] McFarland, M. B., "Adaptive Nonlinear Control of Missiles Using Neural Networks," Ph.D. Thesis, School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, September 1997.

[64] Leitner, J., Calise, A. J., and Prasad, J. V. R., "Analysis of Adaptive Neural Networks for Helicopter Flight Control," *AIAA Guidance, Navigation, and Control Conference*, pp. 871-9, Baltimore, MD, 1995.

[65] McFarland, M. B., " Augmentation of Gain-Scheduled Missile Autopilots Using Adaptive Neural Networks," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4491, pp. 1786-1792, Boston, MA, 1998.

[66] "Reconfigurable Flight Control for a Tailless Advanced Fighter Aircraft," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4107, pp. 75-87, Boston, MA, 1998.

[67] "Direct Adaptive Reconfigurable Control of a Tailless Fighter Aircraft," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4108, pp.88-98, Boston, MA, 1998.

[68] Lin, C., Xu, R., Kwan, C., and Haynes, L.,'Submarine Pitch and Depth Control using FCMAC Neural Networks ,'*American Control Conference*, 1998.

[69] Krstic, M., Kanellakopoulos, I., and Kokotovic, P., "Nonlinear and Adaptive Control Design," John Wiley & Sons, Inc, NY, 1995.

[70] Shtessel, Y., Buffington, J., Pachter, M., Chandler, P., and Banda, S., "Reconfigurable Flight Control on Sliding Modes Addressing Actuator Delfection and Deflection Rate Saturation," *AIAA Guidance, Navigation, and Control Conference*, AIAA-98-4112, pp.127-35, Boston, MA, 1998.

[71] Stevens, B. L., and Lewis, F. L., "Aircraft Flight Dynamics and Automatic Flight Control," John Wiley & Sons, Inc, 1992.

[72] Maine, R. E., and Iliff, K. W., "Application of Parameter Estimation oto Aircraft Stability and Control: /cm the Output-Error Approach," NASA Reference Publication RP-1168, June, 1986.

[73] Maine, R. E., and Iliff, K. W., "The Theory and Practice of Estimating the Accuracy of Dynamics Flight-Determined Coefficients," NASA Reference Publication RP-1077, July, 1981.

[74] Roskam, J., "Airplane Flight Dynamics and Automatic Flight Controls: Part I," Roskam Aviation and Engineering Corporation, Ottawa, KS 66067.

[75] Peterson, G.,; Bond, W., Germann, R., Streeter, B., Urnes, J., "Using Neural Networks for Aerodynamic Parameter Modeling," 1995 American Control Conference, 14th, Seattle, WA, June 21-23, 1995, Proceedings. Vol. 2 (A95-4164911-63), Piscataway, NJ, IEEE, 1995, p. 1360-1361

[76] Gelb, A., "Applied Optimal Estimation," Analytic Sciences Corporation, M.I.T Press, 1974.

[77] Brown, R. G., and Hwang, P. Y.C., "Introduction to Random Signals and Applied Kalman Filtering," 2nd Edition,John Wiley & Sons, Inc, 1992.

[78] Friedland, B., and Director, S. W., "Control System Design:An Introduction to State-Space Methods," McGraw-Hill, 1985.

[79] Miotto, P., Paduano, J. D., Feron, E., and Burken, J. J., "Modern Fixed Structure Control Design Part I:Gain Adjustment to Improve Handling Qualities," *AIAA Guidance, Navigation, and Control Conference*, AIAA-97-37001, pp. 144-154, New Orleans, 1997.

[80] Miotto, P., Paduano, J. D., Feron, E., and Burken, J. J., "Modern Fixed Structure Control Design Part II: Automated Gain Scheduling," *AIAA Guidance, Navigation, and Control Conference*, New Orleans, 1997.

[81] Blake, M. W., "The NASA Advanced Concepts Flight Simulator: A Unique Transport Aircraft Research Enviroment," *AIAA Flight Simulation Technologies Conference*, AIAA-96-3518-CP, San Diego, CA, 1996.

[82] "Jane's All the World's Aircraft," Sampson Low, Marston & Co., London, England, 1998.

[83] Kaneshige, J., Bull, J., Kudzia, E., and Burcham, F. W., "Propulsion Control with Flight Director Guidance as an Emergnecy Flight Control System," *AIAA Guidance, Navigation, and Control Conference*, AIAA-99-3962, Portland, OR, 1999.

THIS PAGE LEFT BLANK INTENTIONALLY