AD NUMBER
ADB269236
NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies only; Proprietary Info.; Jul 2001. Other requests shall be referred to Air Force Office of Scientific Research, AFSOR/NM, 801 N. Randolph St., rm 732, Arlington, VA 22203-1977.
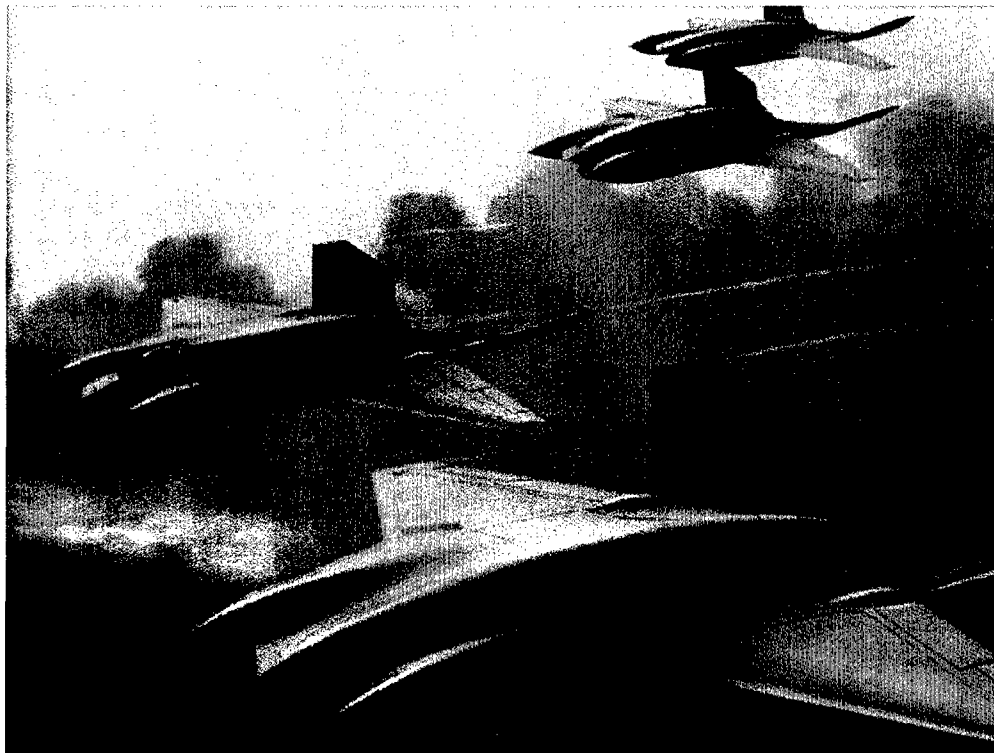
AUTHORITY

AFOSR ltr, 30 May 2002

FZM-8690
07/03/01

LOCKHEED MARTIN

EXPORT
CONTROL

Robust Nonlinear
Aircraft Flight Control

FINAL REPORT

James M. Buffington, Ph.D.

Principal Investigator

20010815 023

REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-01-

$C430$

1. REPORT DATE (DD-MM-YYYY) 05-07-2001	2. REPORT TYPE Final Technical Report	3. DATES COVERED (From - To) June 1998 – May 2001

4. TITLE AND SUBTITLE	
Robust Nonlinear Aircraft Flight Control	5a. CONTRACT NUMBER F49620-98-C-0027
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Buffington, James M.; Niestroy, Michael A.; Ha, Chi M.	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Lockheed Martin Aeronautics Company Air Vehicle Sciences and Systems P.O. Box 748 Fort Worth, TX 76101 Attn: Dr. James M. Buffington, MZ 9338	Lockheed Martin Aeronautics Company Report No. FZM-8690

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
Air Force Office of Scientific Research AFOSR/NM 801 N. Randolph St., Rm 732 Arlington, VA 22203-1977 Attn: Dr. Marc Jacobs	AFOSR/NM
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT

DISTRIBUTION STATEMENT B
Distribution authorized to
U. S. Govt. Agencies Only
Other requests shall be referred to:
(3 JUL 2001)

PROPRIETARY INFORMATION

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This project addresses the development of both control synthesis and analysis methods for affordable, highly integrated nonlinear aircraft control systems through basic research and development of control science and dynamical systems theory. The technical approach is to explore the following three primary research areas: 1) stability analysis, 2) robust nonlinear control, and 3) robust reconfigurable control. The stability analysis task assesses robustness of a baseline flight control system. The analysis results are used to develop on-board model and control allocation requirements for the other two tasks. The robust nonlinear control task includes development of compact on-board model and dynamic control allocation synthesis methods. The reconfigurable control design task includes development of an on-board model update synthesis method. The main accomplishment for the stability analysis task is using structured singular value analysis to formalize control allocation stability implications and to specify stability and control derivative accuracy requirements for on-board model synthesis. The main accomplishments for the robust nonlinear control design task are development of a dynamic control allocation formulation and development of a compact on-board model synthesis method. The main accomplishment for the robust reconfigurable control design task is the development of an on-line learning method for on-board model updates to enhance the robustness of indirect adaptive flight control systems.

15. SUBJECT TERMS
nonlinear flight control, control allocation, on-board models, intelligent control, stability analysis, robust control, reconfigurable control

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Dr. James M. Buffington
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified	UL	59	19b. TELEPHONE NUMBER (include area code) 817.935.1030

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

LOCKHEED MARTIN

Aeronautics Company
Code Ident 81755

ROBUST NONLINEAR AIRCRAFT FLIGHT CONTROL FINAL REPORT

CONTRACT F49620-98-C-0027

CLIN 0002AA

Prepared and Approved for Release by:

James M. Buffington, Ph.D. - Principal Investigator

LOCKHEED MARTIN AERONAUTICS COMPANY

DISTRIBUTION STATEMENT

WARNING - This document contains technical data whose export is Restricted by the Arms Export Control Act (Title 22, U.S.C. Sec 2751, et seq.) or the Export Administration Act of 1979, as amended, Title 50, U.S.C., App 2401 et seq. Violations of these export laws are subject to severe criminal penalties. Disseminate in accordance with provisions of AFR 80-34.

Releasability of this material under the Freedom of Information Act is subject to the restrictions on release in DoD Regulation 5400.7-R and DoD Directive 5230.25.

FOREWORD

Lockheed Martin Aeronautics Company prepared this document for the Air Force Office of Scientific Research under contract F49620-98-C-0027. The work documented was conducted between June 1998 and April 2001. Contributors from Lockheed Martin Aeronautics Company were Dr. Jim Buffington, Dr. Mike Niestroy, Dr. Chi Ha, Dr. Paul Wei and Dr. Rowena Eberhardt. The work was performed under the direction of Dr. Marc Jacobs from the Air Force Office of Scientific Research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.PROJECT OVERVIEW

1.1. Scope

The scope of this project is basic research and development of control science and dynamical systems theory with applications toward aircraft flight control. This project addresses the development of both control synthesis and analysis methods for affordable, highly integrated nonlinear aircraft control systems.

1.2. Background

Dynamic inversion is a nonlinear control law synthesis technique that has found widespread application in the flight control community [2],[3],[4]. A major reason for the appeal of this full state feedback synthesis technique is the ease with which it can be applied to an entire flight envelope without traditional gain scheduling, and the direct accommodation of stability and performance requirements. Since dynamic inversion may be cast in the general setting of control-affine nonlinear systems, the design methodology has the advantage of inherently accounting for nonlinearities in aerodynamics, propulsion, and kinematic couplings. Dynamic inversion will continue to become prevalent due to the potential advantages it offers toward achieving more stringent requirements of future aircraft such as uninhabited combat air vehicles (UCAV). However, current theoretical tools are insufficient to fully analyze and certify the complex systems that must be ultimately implemented to realize maximum capability.

A critical challenge in applying dynamic inversion is the control allocation and optimization (CAO). CAO is employed by the dynamic inversion methodology to optimally combine redundant control effectors such that their effect is reduced to a single effective control input producing a body moment in the pitch, roll, and yaw axes [3], [8]. While CAO ensures that each effector does not violate its rate or position limit, current allocation methods do not properly account for closed-loop stability due to the static nature of the accepted problem formulation. Simulation responses have shown that dissimilar actuator bandwidths in redundant systems using current control allocation methods may cause degradation of stability margins [3]. Although practical solutions have been implemented as protection, they tend to be conservative and have no stability guarantees. A thorough analysis is needed for a better understanding of the stability implications of advanced control allocation. Once the stability has been analyzed, new dynamic control allocation methods may be required to accommodate actuator dynamics that currently are unmodeled.

The on-board model (OBM) is another important element that provides a nominal estimate of the vehicle dynamics in a dynamic inversion control system. The OBM

1

provides information to model-based control laws such as dynamic inversion and to parameter identification algorithms for adaptation. However because of on-board computational memory and throughput limitations, an emphasis is placed on creating a compact on-board model. Therefore, model accuracy may be sacrificed to meet these requirements. Additionally, wind tunnel tests and CFD analyses may introduce modeling errors that will not appear until an aircraft reaches the flight test phase. While many methods exist to provide improved estimates of the vehicle dynamics either on-line or off-line, these methods typically provide only linear stability and control derivative information. These linear derivative estimates, while useful for analysis purposes, cannot be used to directly to modify the nonlinear on-board model. Methods are needed to directly accommodate model accuracy requirements during the synthesis of compact on-board models. Methods that address OBM reconfiguration/adaptation and are consistent with the nonlinear aerodynamic model architecture are needed as well.

1.3. Goal and Objectives

The primary goal of this project is to develop, extend, and apply dynamical system and control theoretic methods to enable cost-effective development of advanced aircraft flight control systems. Specific objectives have been devised to increase transition opportunity by highly leveraging and integrating with other research projects and aircraft programs at Lockheed Martin Aeronautics Company (LM Aero).

The primary objective is to enhance the LM Aero modular flight control system architecture that includes an on-board model and a control allocator. Specific objectives include development of: 1) stability analysis methods for control allocation, 2) dynamic control allocation methods to accommodate actuator dynamics, 3) on-board model accuracy requirements using stability analysis, 4) compact on-board model synthesis methods, and 5) nonlinear on-board model reconfiguration methods.

1.4. Benefits

Advanced control allocation and on-board modeling methods offer the ability to improve aircraft safety, agility, and signature by optimally exploiting control redundancy and providing on-line control parameter computation. These methods also decrease operations and support (O&S) costs by replacing hardware redundancy with analytical control reconfiguration. However, inefficient and ineffective analysis and synthesis methods increase flight control system development costs and prevent minimal development schedules. Development, extension, and application of dynamical system and control theoretic methods will enable compressed flight control system development schedules and decreased costs.

1.5. Approach

The program approach is to develop technology with high transition potential and maximum leveraging ability with other research and development projects. LM Aero product programs have been included in the task definition and project plan development. This improves strategic value for LM Aero and potential for technology transition to Air Force assets. The tasks have been tailored to heavily leverage other research and development programs that are funded by AFOSR or being worked at LM Aero under funding from sources other than AFOSR. Results from the LM Aero RESTORE program, funded by AFRL/VAC, and the Integrated Adaptive Control Systems project, from LM Aero IRAD funds, are being leveraged by this project.

The technical approach is to explore the following three primary research areas: 1) stability analysis, 2) robust nonlinear control, and 3) robust reconfigurable control. The stability analysis task assesses robustness of the baseline flight control system. The analysis results are used to develop on-board model and control allocation synthesis methods in the other two tasks. The robust nonlinear control task includes development of compact on-board model and dynamic control allocation synthesis methods. The reconfigurable control design task includes development of an on-board model update synthesis method. These three tasks use neural networks, μ-analysis, dynamical systems, optimization, and parameter identification theory to accomplish the project objectives.

1.6. Summary of Accomplishments

There have been significant accomplishments in the development of stability analysis, robust nonlinear control design, and robust reconfigurable control design methods for aircraft flight control systems. These accomplishments are summarized below and have been used as a foundation for the recently submitted Autonomous Flight Control proposal.

There have been primary accomplishments for each of the three project tasks. The main accomplishment for the stability analysis task is using μ-analysis to formalize control allocation stability implications and to specify stability and control derivative accuracy requirements for on-board model synthesis. The main accomplishments for the robust nonlinear control design task are development of a dynamic control allocation formulation and development of a compact on-board model synthesis method. The main accomplishment for the robust reconfigurable control design task is the development of an on-line learning method for on-board model updates to enhance the robustness of indirect adaptive flight control systems. These accomplishments provide capabilities that are applicable to current aircraft programs and R&D projects thereby increasing transition opportunity.

LM Aero has used μ-analysis to assess robust stability and performance in the presence of on-board model uncertainties and dissimilar actuator bandwidths. This

3

analysis has been used to develop on-board model accuracy requirements for on-board model synthesis. We also analyzed stability of a static control allocator to confirm stability margin degradation in the presence of dissimilar actuator bandwidths observed in simulation. This has facilitated a better understanding of the stability implications of advanced control allocation, which has allowed development of a dynamic control allocator.

LM Aero has developed a preliminary dynamic control allocation synthesis method that enables solutions to address closed loop stability and exploit control redundancy to optimize stability margins. Dynamical systems and optimization theory has been successfully used for dynamic control allocation synthesis that directly accommodates actuator bandwidths. We have also developed an on-board modeling method using neural networks. This method provides an efficient process to develop compact and accurate on-board models that directly satisfy accuracy requirements.

LM Aero has developed an approach that uses the fundamental aerodynamic build-up equations and a parameter identification algorithm to directly produce on-board model increments for reconfiguration. These increments are consistent with the nonlinear aerodynamic model architecture which facilitates easy integration with the nominal on-board model to provide a more accurate dynamical representation of the aircraft.

1.7. Personnel

The LM Aero principal investigator for this effort is Dr. Jim Buffington. The primary technical contributors are Dr. Mike Niestroy and Dr. Chi Ha. Other contributors include Dr. Paul Wei and Dr. Rowena Eberhardt.

2.TECHNICAL

2.1. Task Description

A dynamic inversion flight control system is used as the baseline architecture in this project. The three technical tasks for this project are stability analysis for flight control systems, robust flight control design, and reconfigurable flight control design.

The stability analysis task includes robust stability and performance analysis of the LM Aero modular flight control system. Frequency-domain structured singular value analysis (μ-analysis) is used to assess robust stability and performance of a dynamic inversion controller for the ICE tailless fighter aircraft [9]. On-board models in the presence of aerodynamic model parametric uncertainty and static control allocation with unmodeled dissimilar actuator dynamics are the focus of the analysis. On-board model accuracy requirements are developed from the stability analysis results. The analysis results are also used to formalize control allocation stability implications.

The robust nonlinear control design task includes development of a compact on-board model synthesis method and preliminary development of a dynamic control allocation synthesis method. Neural networks are used for compact on-board model synthesis.   Dynamical systems and optimization theory is used for dynamic control allocation synthesis that directly accommodates dissimilar actuator bandwidths to improve robustness and lessen design conservatism.

The reconfigurable control design task includes development of an on-board model update synthesis method. Neural network and parameter identification theory is used to provide a model update that is consistent with the nonlinear aerodynamic model architecture.

2.2. Baseline Flight Control System

The LM Aero baseline flight control system is model-based and adaptive. This system, shown in Figure 1, has evolved from many years of advanced research, development, and production of digital fly-by-wire aircraft. Key features are on-line gain computation within the OCD module, optimal effector blending within the CAO module, separation of aircraft dependent model parameters within the OBM, and on-line adaptation within the PID module. Together these modules comprise a modular, model-based, adaptive system that provides optimal control reconfiguration. The focus of this project is enhancement of the OBM and CAO modules with a dynamic inversion OCD module. The dynamic inversion OCD and nominal CAO are derived in the following two sections.
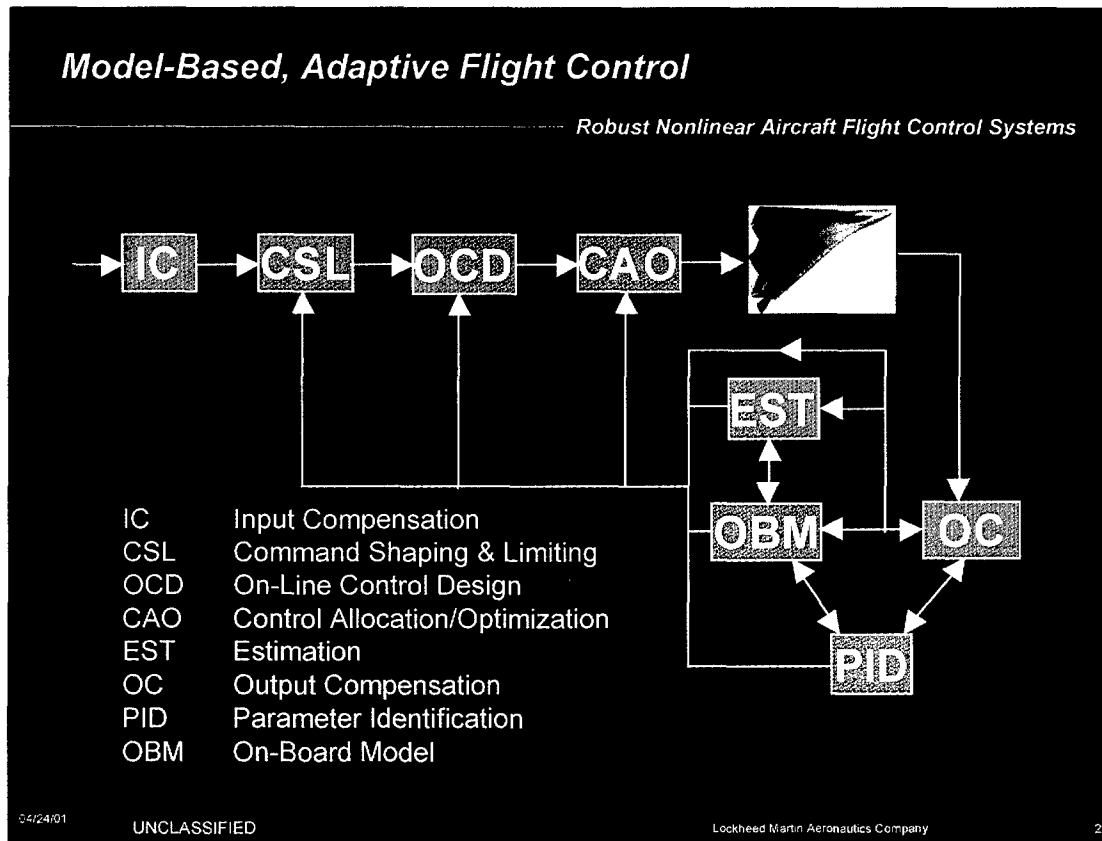
Figure 1 – Modular Flight Control System Architecture

2.2.1. Rigid-Body Aircraft Model

A general form of aircraft nonlinear equations of motion is expressed as

(1) $\quad \dot{X}(t) = f(X(t), \delta(t))$

where $X(t)$ is the state vector, $\delta(t)$ is the control vector and f is a vector-valued nonlinear function of the individual states and controls. The linearized aircraft rigid-body model is a five-state approximation of eq. (1) given by

(2) $\quad \dot{x} = Ax + B\delta$

$$x = \begin{bmatrix} q, \text{pitch rate}, \text{deg/sec} \\ \alpha, \text{angle of attack}, \text{deg} \\ p, \text{body} - \text{axis roll rate}, \text{deg/sec} \\ r, \text{body} - \text{axis yaw rate}, \text{deg/sec} \\ \beta, \text{sideslip}, \text{deg} \end{bmatrix} \in R^5$$

$$\delta = \begin{bmatrix} \delta_{el}, \text{left elevon}, \text{deg} \\ \delta_{er}, \text{right elevon}, \text{deg} \\ \delta_{pflap}, \text{pitch flap}, \text{deg} \\ \delta_{oblfl}, \text{left outboard leading} - \text{edge flap}, \text{deg} \\ \delta_{oblfr}, \text{right outboard leading} - \text{edge flap}, \text{deg} \\ \delta_{ptv}, \text{pitch thrust vectoring}, \text{deg} \\ \delta_{ytv}, \text{yaw thrust vectoring}, \text{deg} \\ \delta_{ssdl}, \text{left spoiler} - \text{slot deflector}, \text{deg} \\ \delta_{ssdr}, \text{right spoiler} - \text{slot deflector}, \text{deg} \\ \delta_{amtl}, \text{left all moving wing tip}, \text{deg} \\ \delta_{amtr}, \text{right all moving wing tip}, \text{deg} \end{bmatrix} \in R^{11}$$

for the ICE aircraft model [9] used during this project.

2.2.2. Dynamic Inversion Controller

A summary derivation of the dynamic inversion controller is given below while a complete derivation is given in [3]. The derivation begins with the definition of the commanded variables which are given by

$$(3) \quad y = \begin{bmatrix} MCV \\ LCV \\ NCV \end{bmatrix} = C(q, \alpha, p, r, \beta) = \begin{bmatrix} q + K_\alpha \alpha \\ p\cos\alpha + r\sin\alpha \\ -p\sin\alpha + r\cos\alpha + K_\beta \beta \end{bmatrix} \in R^3$$

or the lineraized version given by

$$(4) \quad \begin{bmatrix} MCV \\ LCV \\ NCV \end{bmatrix} = T \begin{bmatrix} q \\ \alpha \\ p \\ r \\ \beta \end{bmatrix}$$

$$T = \frac{\partial C}{\partial x}\bigg|_{\alpha_{trim}} = \begin{bmatrix} 1 & K_\alpha & 0 & 0 & 0 \\ 0 & (-p\sin\alpha + r\cos\alpha)*\left(\frac{\pi}{180}\right) & \cos\alpha & \sin\alpha & 0 \\ 0 & (-p\cos\alpha - r\sin\alpha)*\left(\frac{\pi}{180}\right) & -\sin\alpha & \cos\alpha & K_\beta \end{bmatrix}_{\alpha_{trim}}$$

$$K_\alpha = \frac{\overline{q}S(C_{L_\alpha})_{ref}}{mV_{co}}$$

$$K_\beta = \frac{\overline{q}S(C_{Y_\beta})_{ref}}{mV_{co}}$$

where

MCV: pitch-axis controlled variable in *deg/sec*

LCV: roll-axis controlled variable in *deg/sec*

NCV: yaw-axis controlled variable in *deg/sec*

S: wing reference area = 808.6 *ft²*

C_{L_α} : lift-curve slope (α=0) = 0.0423 *1/rad*

m: aircraft mass = $\dfrac{32750}{g}$ *slugs*

V_{co} : "crossover" airspeed = 400 *ft/sec*

C_{Y_β} : desired side force slope = -1.0 *1/rad*

g: gravitational constant = 32.17 *ft/sec²*

$\overline{q}$: dynamic pressure in *lb/ft²*

V: velocity in *ft/sec*

θ: Euler pitch angle in *deg*

ϕ: Euler roll angle in *deg*

Using eq.(2), the derivative of eq.(4) becomes

(5) $\qquad \begin{bmatrix} M\dot{C}V \\ L\dot{C}V \\ N\dot{C}V \end{bmatrix} = TAx + TB\delta$

Both the controlled variables and their commands form error signals that are filtered to produce desired closed-loop response dynamics given by the following

8

$$(6) \quad \begin{bmatrix} M\dot{C}V \\ L\dot{C}V \\ N\dot{C}V \end{bmatrix}_{desired} = \omega_c \left(f_c \begin{bmatrix} MCV_c \\ LCV_c \\ NCV_c \end{bmatrix} - \begin{bmatrix} MCV \\ LCV \\ NCV \end{bmatrix} + \frac{\omega_c f_i}{s} \left\{ \begin{matrix} MCV_c - MCV \\ LCV_c - LCV \\ NCV_c - NCV \end{matrix} \right\} \right)$$

where MCV_c (deg/sec) is the pitch-axis controlled variable command, LCV_c (deg/sec) is the roll-axis controlled variable command, NCV_c (deg/sec) is the yaw-axis controlled variable command.

The parameters are chosen to be ω_c = 5 *rad/sec*, f_c = 0.5, f_i = 0.25 for the ICE aircraft application which result in closed-loop poles of $\omega_c / 2$ *rad/sec*. The desired signals in eq. (6) are then used to define the following dynamic inversion algorithm

$$(7) \quad TB\delta = \begin{bmatrix} M\dot{C}V \\ L\dot{C}V \\ N\dot{C}V \end{bmatrix}_{desired} - TAx \equiv m$$

which is the OCD module in Figure 1.

A final calculation is required to obtain effector commands. If all effector actuator dynamics are fast relative to the desired closed-loop response then the actuator deflections are approximately equal to the commands ($\delta \approx \delta_c$), and the pseudo inverse of TB matrix in eq. (7) completes the following dynamic inversion algorithm

$$(8) \quad \delta_c = (TB)^T \left(TB(TB)^T \right)^{-1} m$$

The pseudo inverse $(TB)^T \left(TB(TB)^T \right)^{-1}$ is referred to as "static control allocation" and is the foundation for the CAO module in Figure 1.

The control law in eq.(8) will give the following desired response in the absence of any uncertainty

$$(9) \quad \begin{bmatrix} M\dot{C}V \\ L\dot{C}V \\ N\dot{C}V \end{bmatrix} = \begin{bmatrix} M\dot{C}V \\ L\dot{C}V \\ N\dot{C}V \end{bmatrix}_{desired}$$

However, uncertainty exists in many forms including parametric uncertainty in the A and B matrices (OBM) and unmodeled actuator dynamics in the following form

$$(10) \quad \delta = G_{act}(s)\delta_c$$

The following section describes stability analysis in the presence of OBM parametric uncertainty and unmodeled actuator dynamics.

2.3. Stability Analysis of Flight Control Systems

The purpose of this task is to analyze the robust stability and performance of the baseline flight control system. Robustness to OBM parametric uncertainty and CAO

9

unmodeled actuator dynamics is assessed. OBM parametric uncertainty robustness is analyzed to develop OBM accuracy requirements. Unmodeled actuator dynamics robustness is analyzed to confirm and formalize stability margin degradation in the presence of dissimilar actuator bandwidths. The task was performed using μ-analysis with the structure of the aerodynamic uncertainties modeled via Linear Fractional Transformations (LFT) [1] within Xmath/SystemBuild [13]. The first step in μ-analysis is the development of an uncertain model in an LFT form. The next section describes development of an aerodynamic uncertainty model for use as an OBM.

2.3.1. Uncertainty Model

The linear dynamic inversion controller in eq.(8) uses the non-dimensional stability and control derivative coefficients to compute the matrices A and B. Uncertainty in each matrix element is modeled as the summation of a nominal value and a perturbation as shown below

$$
A = \begin{bmatrix}
\overline{M}_q\left(1+\Delta_{M_q}\right) & \overline{M}_\alpha\left(1+\Delta_{M_\alpha}\right) & 0 \\
\overline{Z}_q\left(1+\Delta_{Z_q}\right) & \overline{Z}_\alpha\left(1+\Delta_{Z_\alpha}\right) & 0 \\
0 & \overline{L}_\alpha\left(1+\Delta_{L_\alpha}\right) & \overline{L}_p\left(1+\Delta_{L_p}\right) \cdots \\
0 & \overline{N}_\alpha\left(1+\Delta_{N_\alpha}\right) & \overline{N}_p\left(1+\Delta_{N_p}\right) \\
0 & \overline{Y}_\alpha\left(1+\Delta_{Y_\alpha}\right) & \overline{Y}_p\left(1+\Delta_{Y_p}\right)
\end{bmatrix}
$$

(11)

$$
\begin{bmatrix}
0 & \overline{M}_\beta\left(1+\Delta_{M_\beta}\right) \\
0 & \overline{Z}_\beta\left(1+\Delta_{Z_\beta}\right) \\
\overline{L}_r\left(1+\Delta_{L_r}\right) & \overline{L}_\beta\left(1+\Delta_{L_\beta}\right) \\
\overline{N}_r\left(1+\Delta_{N_r}\right) & \overline{N}_\beta\left(1+\Delta_{N_\beta}\right) \\
\overline{Y}_r\left(1+\Delta_{Y_r}\right) & \overline{Y}_\beta\left(1+\Delta_{Y_\beta}\right)
\end{bmatrix}
$$

$$B = \begin{bmatrix} \overline{M}_{\delta el}\left(1+\Delta_{M_{\delta el}}\right) & \overline{M}_{\delta er}\left(1+\Delta_{M_{\delta er}}\right) & \overline{M}_{\delta pflap}\left(1+\Delta_{M_{\delta pflap}}\right) \\ \overline{Z}_{\delta el}\left(1+\Delta_{Z_{\delta el}}\right) & \overline{Z}_{\delta er}\left(1+\Delta_{Z_{\delta er}}\right) & \overline{Z}_{\delta pflap}\left(1+\Delta_{Z_{\delta pflap}}\right) \\ \overline{L}_{\delta el}\left(1+\Delta_{L_{\delta el}}\right) & \overline{L}_{\delta er}\left(1+\Delta_{L_{\delta er}}\right) & 0 & \cdots \\ \overline{N}_{\delta el}\left(1+\Delta_{N_{\delta el}}\right) & \overline{N}_{\delta er}\left(1+\Delta_{N_{\delta er}}\right) & 0 \\ \overline{Y}_{\delta el}\left(1+\Delta_{Y_{\delta el}}\right) & \overline{Y}_{\delta er}\left(1+\Delta_{Y_{\delta er}}\right) & 0 \end{bmatrix}$$

$$\begin{array}{ccc} \overline{M}_{\delta oblfl}\left(1+\Delta_{M_{\delta oblfl}}\right) & \overline{M}_{\delta oblfr}\left(1+\Delta_{M_{\delta oblfr}}\right) & \overline{M}_{\delta ptv}\left(1+\Delta_{M_{\delta ptv}}\right) \\ \overline{Z}_{\delta oblfl}\left(1+\Delta_{Z_{\delta oblfl}}\right) & \overline{Z}_{\delta oblfr}\left(1+\Delta_{Z_{\delta oblfr}}\right) & \overline{Z}_{\delta ptv}\left(1+\Delta_{Z_{\delta ptv}}\right) \\ \overline{L}_{\delta oblfl}\left(1+\Delta_{L_{\delta oblfl}}\right) & \overline{L}_{\delta oblfr}\left(1+\Delta_{L_{\delta oblfr}}\right) & \overline{L}_{\delta ptv}\left(1+\Delta_{L_{\delta ptv}}\right) & \cdots \\ \overline{N}_{\delta oblfl}\left(1+\Delta_{N_{\delta oblfl}}\right) & \overline{N}_{\delta oblfr}\left(1+\Delta_{N_{\delta oblfr}}\right) & \overline{N}_{\delta ptv}\left(1+\Delta_{N_{\delta ptv}}\right) \\ \overline{Y}_{\delta oblfl}\left(1+\Delta_{Y_{\delta oblfl}}\right) & \overline{Y}_{\delta oblfr}\left(1+\Delta_{Y_{\delta oblfr}}\right) & \overline{Y}_{\delta ptv}\left(1+\Delta_{Y_{\delta ptv}}\right) \end{array}$$

$$\begin{array}{ccc} \overline{M}_{\delta ytv}\left(1+\Delta_{M_{\delta ytv}}\right) & \overline{M}_{\delta ssdl}\left(1+\Delta_{M_{\delta ssdl}}\right) & \overline{M}_{\delta ssdr}\left(1+\Delta_{M_{\delta ssdr}}\right) \\ \overline{Z}_{\delta ytv}\left(1+\Delta_{Z_{\delta ytv}}\right) & \overline{Z}_{\delta ssdl}\left(1+\Delta_{Z_{\delta ssdl}}\right) & \overline{Z}_{\delta ssdr}\left(1+\Delta_{Z_{\delta ssdr}}\right) \\ \overline{L}_{\delta ytv}\left(1+\Delta_{L_{\delta ytv}}\right) & \overline{L}_{\delta ssdl}\left(1+\Delta_{L_{\delta ssdl}}\right) & \overline{L}_{\delta ssdr}\left(1+\Delta_{L_{\delta ssdr}}\right) & \cdots \\ \overline{N}_{\delta ytv}\left(1+\Delta_{N_{\delta ytv}}\right) & \overline{N}_{\delta ssdl}\left(1+\Delta_{N_{\delta ssdl}}\right) & \overline{N}_{\delta ssdr}\left(1+\Delta_{N_{\delta ssdr}}\right) \\ \overline{Y}_{\delta ytv}\left(1+\Delta_{Y_{\delta ytv}}\right) & \overline{Y}_{\delta ssdl}\left(1+\Delta_{Y_{\delta ssdl}}\right) & \overline{Y}_{\delta ssdr}\left(1+\Delta_{Y_{\delta ssdr}}\right) \end{array}$$

(12)

$$\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ \overline{L}_{\delta amtl}\left(1+\Delta_{L_{\delta amtl}}\right) & \overline{L}_{\delta amtr}\left(1+\Delta_{L_{\delta amtr}}\right) \\ \overline{N}_{\delta amtl}\left(1+\Delta_{N_{\delta amtl}}\right) & \overline{N}_{\delta amtr}\left(1+\Delta_{N_{\delta amtr}}\right) \\ 0 & 0 \end{array}$$

where

$\overline{M}_i, \overline{Z}_i, \overline{L}_i, \overline{N}_i, \overline{Y}_i$ are nominal values

$$\left\|\Delta_{k_i}\right\|_\infty = \sup_\omega \sigma_{max}\left[\Delta_{k_i}(j\omega)\right] \le 1$$

i = q, α, p, r, β, and δj

j = el, er, pflap, oblfl, oblfr, ptv, ytv, ssdl, ssdr, amtl, amtr

k = M, Z, L, N, and Y

The dynamic inversion controller in eq.(8) with OBM parametric uncertainty in eqs.(11) and (12) is interconnected with other model elements to form an LFT [1]. The inputs to the perturbations (Δ_{k_i}) are collected as a single vector z, and the outputs collected as vector v. The total perturbation block Δ consists of uncertain scalars arranged in the following block-diagonal form

(13) $\Delta = diag\left(\Delta_1, \Delta_2, \Delta_3, ..., \Delta_m\right)$

11

The resulting closed-loop LFT is shown in Figure 2.
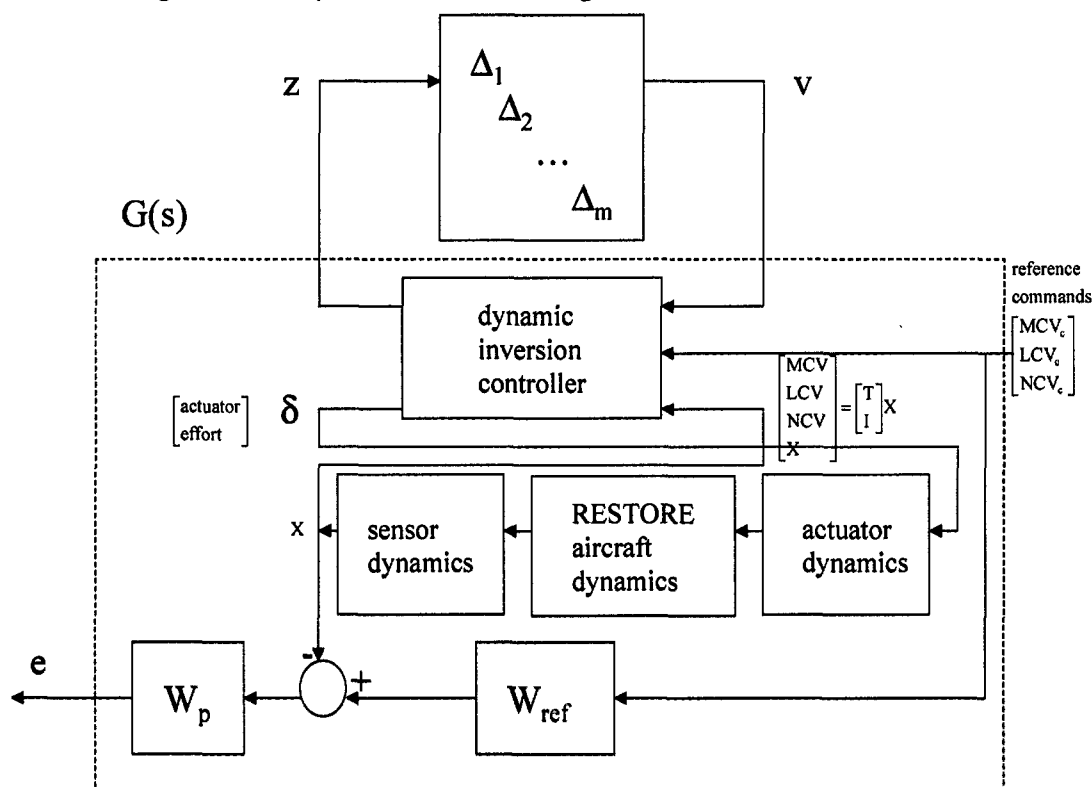
Figure 2 – Closed Loop Interconnection LFT

The dynamic inversion controller receives five sensor measurements (x), three exogenous command signals (CV$_c$), and produces eleven control signals for the control effector commands.

The aircraft dynamics are the linearized RESTORE aircraft dynamics [9] in the form of unperturbed (Δ=0) eqs.(11) and (12). The preliminary sensor dynamics are chosen to be a unity gain, but higher fidelity sensor dynamics are planned for future analysis. The actuator dynamics are modeled as first order low-pass filters described in the next section.

The performance objective is to have the aircraft respond to commands (CV$_c$) without coupling between axes. The desired aircraft response is the following ideal first-order model

$$(14) \quad W_{ref} = diag(\frac{2.5}{s+2.5}, \frac{2.5}{s+2.5}, \frac{2.5}{s+2.5}).$$

A performance weight is placed on the error between the ideal response of the transfer function and the actual aircraft response. The preliminary weight is chosen as follows to place more emphasis on low frequency ideal response

12

$$(15) \quad W_p = diag(\frac{10}{s+10}, \frac{10}{s+10}, \frac{10}{s+10}) \, .$$

More restrictive performance weights will be explored in future analysis.

2.3.2 Analysis Results

In this section, flight control system robustness results are presented in the form of the structured singular value (μ). The structured singular value (μ) is equivalent to the inverse of the smallest destabilizing perturbation. Sixteen flight conditions (Table 1) were selected [9], and μ-analysis was performed at each flight condition to determine the robustness to equally weighted structured uncertainties in the on-board model and sensitivities to dissimilar actuator dynamics.

Table 1 – Analysis Flight Conditions

Flight Condition	Mach	Altitude (ft)	Angle-of-attack (deg)	PLA (deg)
1	0.22	500	15.0	45.6
2	0.35	1000	7.27	31.9
3	0.35	5000	8.25	34.1
4	0.35	15000	11.2	46.5
5	0.45	1000	4.58	32.4
6	0.55	1000	3.32	34.7
7	0.59	15000	4.46	36.6
8	0.65	1000	2.61	39.0
9	0.75	1000	2.21	45.2
10	0.85	1000	2.01	51.1
11	0.95	1000	1.68	65.6
12	1.0	500	1.48	86.9
13	1.1	500	1.17	108
14	1.2	500	1.17	127

A nominal set of actuator dynamics is defined for analysis. This set contains similar bandwidths for all actuators and is referred to as the similar actuator set. The actuator dynamics in the similar actuator set are chosen to contain the following dynamics

$$(16) \quad G_{act}^{sim}(s) = diagonal\left(\frac{20}{s+20}, \frac{20}{s+20}, \frac{20}{s+20}, ..., \frac{20}{s+20}\right)$$

13

Note that the actuator bandwidths are well beyond the controller crossover frequency (ω_c = 5 *rad/sec*).

By collecting all non-perturbation elements into a single block, the block diagram interconnection structure in Figure 2 may be collapsed into the structure shown in Figure 3.
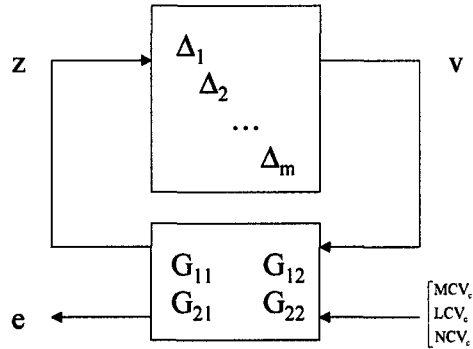

Figure 3 - Closed-Loop Model for Robustness Analysis

The non-perturbation block may be partitioned into four parts. The upper left partition (G_{11}) is related to robust stability, the lower right partition (G_{22}) is related to nominal performance, and robust performance is related to all partitions. For example, the peak of the upper bound for μ of $G_{11}(s)$ implies that there is a perturbation equal to its inverse ($1/\mu$) that causes instability.

Robust stability and performance are analyzed, and the results are presented as the smallest destabilizing perturbation plotted versus dynamic pressure. The aerodynamic parametric uncertainties are assumed to be complex scalars for the preliminary analysis, while future analysis will utilize real-valued perturbations.

The following robust stability uncertainty structure contains 64 complex-valued scalars(18 uncertainties in the stability derivative matrix A and 46 in the control derivative matrix B)

(17) $\Delta_{RS} = \{diag[\Delta_1, \Delta_2, ..., \Delta_{64}] : \Delta_i \in C\}$

To accommodate robust performance analysis, three additional perturbations that connect the commands (CV_c) to the errors (e) are required as follows

(18) $CV_c = \Delta_p e$

The appropriate uncertainty block structure for the robust performance is given by

(19) $\Delta_{RP} = \{diag[\Delta_1, \Delta_2, ..., \Delta_{64}, \Delta_p] : \Delta_i \in C, \Delta_p \in C^{3x3}\}$

which is simply an augmentation of the robust stability uncertainty set, Δ_{RS}, with a complex 3 x 3 full block to include the performance objectives.

2.3.2.1.   OBM Parametric Uncertainty

The purpose of the OBM parametric uncertainty analysis is to utilize robust stability and performance analysis to develop accuracy requirements for synthesis of the

14

OBM. The linear Xmath/SystemBuild model of the baseline flight control system in Figure 1, in the presence of the OBM uncertainty in eqs.(11) and (12) and unmodeled actuator dynamics in eq.(16), is analyzed using μ-analysis. The robust stability and performance results are shown in Figure 4 for the LFT model in Figure 2.
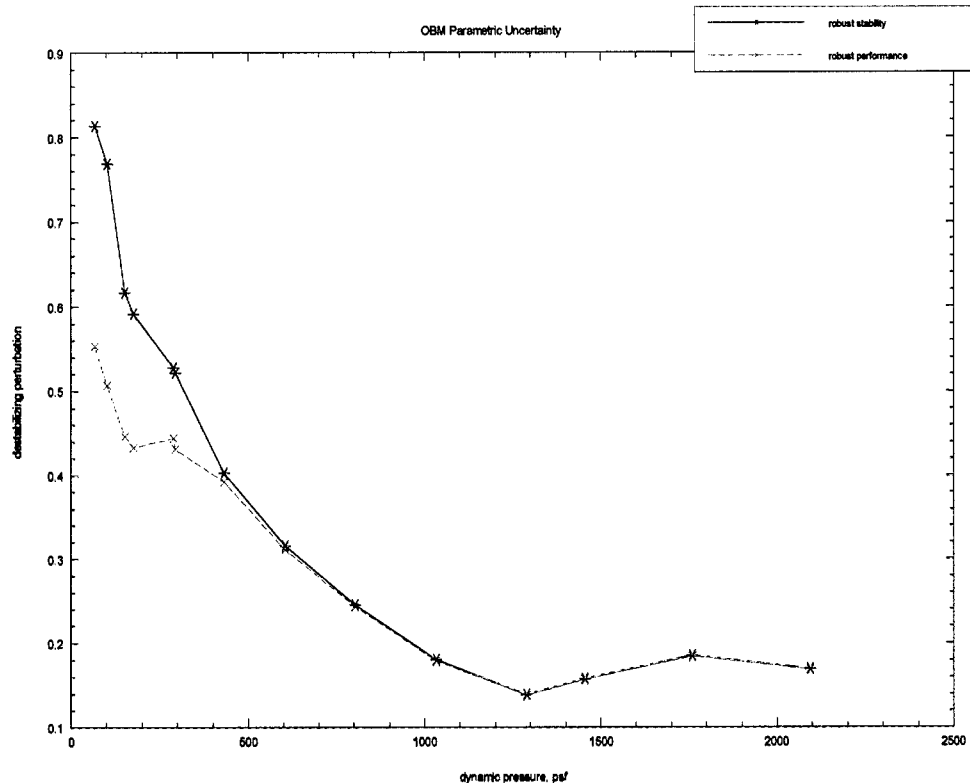
Figure 4 – OBM Parametric Uncertainty Analysis Results

Note that robust performance is less than robust stability at low dynamic pressure conditions, but robust stability and performance are equivalent at higher dynamic pressure conditions. A more restrictive performance weight will most likely cause further reduction in robust performance. Higher dynamic pressure conditions are generally less robust than lower dynamic pressure conditions. These results have been formulated into a preliminary OBM accuracy requirement of 14% for all parameters at all flight conditions. This requirement is the worst case at an approximate dynamic pressure of 1300 lb/ft^2. This requirement will be used in future development of a nominal OBM described in section 2.4.

2.3.2.2. CAO Unmodeled Dynamics

The purpose of the CAO unmodeled dynamics analysis is to utilize robust stability and performance analysis to confirm and formalize stability margin degradation in the presence of dissimilar actuator bandwidths. The sensitivity of realistic sets of dissimilar unmodeled actuators has been observed with aerodynamic and propulsive effectors for both fighter aircraft [3] and reusable launch vehicles [4].

15

A second actuator set is developed as a notional set of dynamics to facilitate the study of interactions between dissimilar unmodeled actuators, control allocation, and closed-loop stability. This set contains dissimilar bandwidths and is referred to as the dissimilar actuator set. The dissimilar actuator set has the following dynamics

$$(20) \quad G_{act}^{dis}(s) = diagonal\left(\frac{1}{s+1}, \frac{2}{s+2}, \frac{3}{s+3}, ..., \frac{11}{s+11}\right)$$

Note that the dissimilar actuator bandwidths are near the controller crossover frequency (ω_c = 5 rad/sec). We have confirmed sensitivities to unmodeled dissimilar actuator dynamics by comparing the similar actuator model in eq.(16) and the dissimilar actuator model in eq.(20) using an ICE aircraft simulation.

The sensitivities have been additionally confirmed through a formal μ-analysis for the two actuator configurations within the closed-loop LFT in Figure 2. The static control allocator in eq.(8) and the parametric OBM uncertainty in eqs.(11) and (12) were used for the analysis in this section. The robust stability and robust performance analysis results are shown in Figure 5 and Figure 6, respectively.
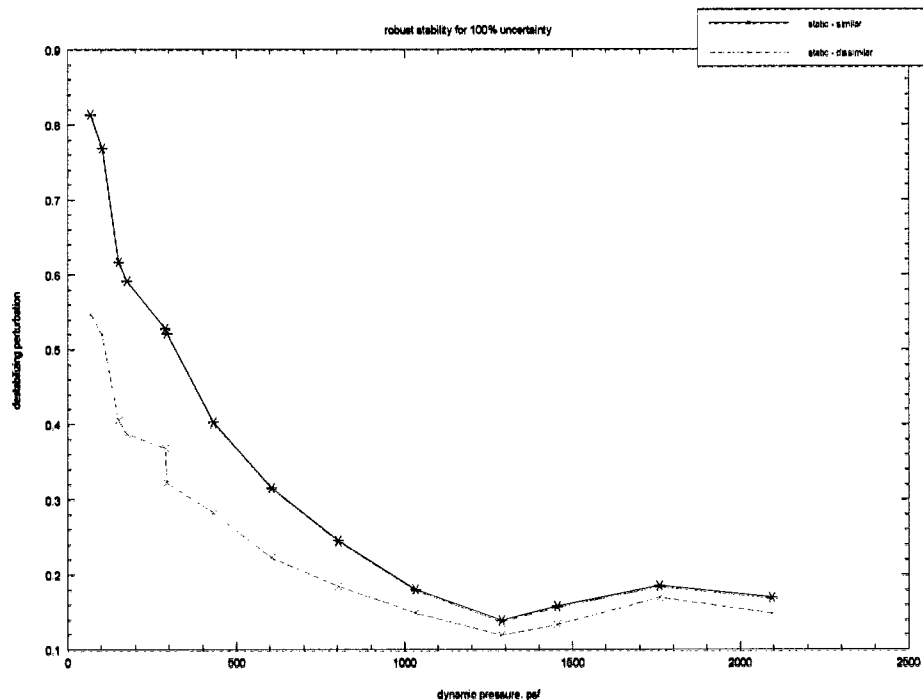
Figure 5 - Robust Stability with Static Control Allocation

Note that robustness is significantly reduced for the dissimilar actuator set at low dynamic pressure conditions. This is primarily due to the fact that the bandwidths of the similar actuator set are equivalent and fast relative to the aircraft closed-loop dynamics. At high dynamic pressure flight conditions, the robustness degradation is less. Direct accommodation of the actuator dynamics in the CAO design may lessen the sensitivity of the dissimilar actuator set.

16
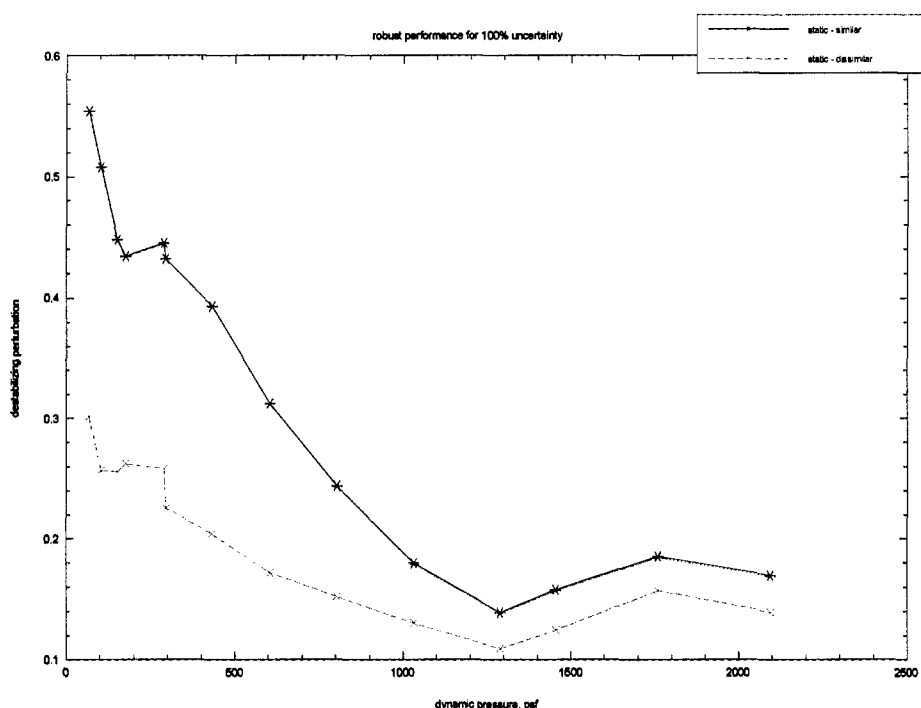
Figure 6 - Robust Performance with Static Control Allocation

2.4. Robust Flight Control Design

The robust nonlinear control design task includes development of a compact OBM synthesis method and preliminary development of a dynamic control allocation synthesis method. Neural networks are used as basis functions for the development of compact on-board models.   Dynamical systems and optimization theory is used for dynamic control allocation synthesis that directly accommodates dissimilar actuator bandwidths to improve robustness and lessen design conservatism.

2.4.1. Compact OBM Synthesis

On-board models provide estimates of vehicle dynamics and are central elements within indirect adaptive control systems, such as those that use dynamic inversion [3] or receding horizon optimal control [9], for example. The indirect adaptive control architecture, shown in Figure 1, utilizes the on-board model to predict the aircraft dynamics and determine the control output necessary to both stabilize the vehicle and accomplish the pilot or autopilot commands. The on-board model also provides information to the route planning algorithms and the prognostic and health management system. The route planner uses the aircraft model to determine the optimal feasible route or maneuver necessary to accomplish the respective long-term

17

or short-term goals. The prognostic and health management system uses the on-board model to predict the nominal behavior of the air vehicle system and compares the expected behavior to the actual system behavior. Differences that develop either slowly over time or suddenly time may indicate an impending or immediate failure, respectively.

Accuracy requirements have been established for the OBM in section 2.3. However, there is a need to directly incorporate these accuracy requirements for development of an OBM while keeping it compact enough to satisfy on-line computational constraints. Utilizing the results of the robustness analysis during synthesis of the nominal OBM, however, is challenging. On-board models typically fit total or incremental aerodynamic coefficients, whereas the robustness analysis results are in the form of linear derivatives. Further, truth derivative information is not well-defined since model data is typically in total form. Once truth derivative information is defined, direct incorporation of derivative accuracy requirements is relatively straightforward. The derivatives of the appropriate coefficients are included in a performance index, as the coefficients are fit, to ensure they meet the requirements.

A compact OBM synthesis is described below using a pitch rate model. However, this method has been applied to the complete ICE aerodynamic database and verified in a high-fidelity nonlinear simulation. Consider the following simplified nonlinear pitch rate equation for the description of the OBM synthesis method

(21) $\dot{Q} = \Delta M_Q(\alpha, M)Q + M_{base}(\alpha, M) + \Delta M_1(\alpha, \delta_1, M)$.

This relationship assumes no lateral/directional affects and only a single control effector for brevity. Numerical values for each of the terms are typically determined through wind tunnel tests. For example, ΔM_Q is found from forced oscillation tests with the control effectors set to a specific position and M_{base} is determined by setting all aerodynamic control effectors to their zero or neutral position. These wind tunnel tests are typically set up to obtain non-dimensional coefficients. For example, consider the base pitching moment increment

(22) $M_{base}(\alpha, M) = \dfrac{C_{M_{base}}(\alpha, M)\bar{q}S\bar{c}}{I_{yy}}$

where $C_{M_{base}}$ is the non-dimensional base pitching moment coefficient. For simulation purposes, the coefficient data is interpolated between data points. However for on-board modeling, compact data fits are necessary to accommodate on-line computational requirements. The base pitching moment coefficient ($C_{M_{base}}$) data is fit using the following feedforward neural network

$$C_{M_{base}}(\alpha, M) = w_{out} f\left(w_{in}\begin{bmatrix} \alpha \\ M \end{bmatrix} + b_{in}\right) + b_{out}$$

$$(23) \qquad f(u) = \begin{cases} 1 - \dfrac{1}{(1+y)^2}, & y \geq 0 \\ \dfrac{1}{(1-y)^2} - 1 & y < 0 \end{cases}$$

An automated tool has been developed for rapid construction of these neural networks. The input and output weights and biases are chosen to minimize a cost function that consists of total fit accuracy. Future work will incorporate linear derivative accuracy.

The resulting neural network approximation for the base pitching moment coefficient is shown in Figure 7. The accuracy of the neural network approximation of the base pitching moment coefficient is shown in Figure 8 at Mach 0.3. All of the coefficients (76 total) for the ICE aerodynamic database have been similarly fit using the neural network architecture in eq. (23) with similar results.
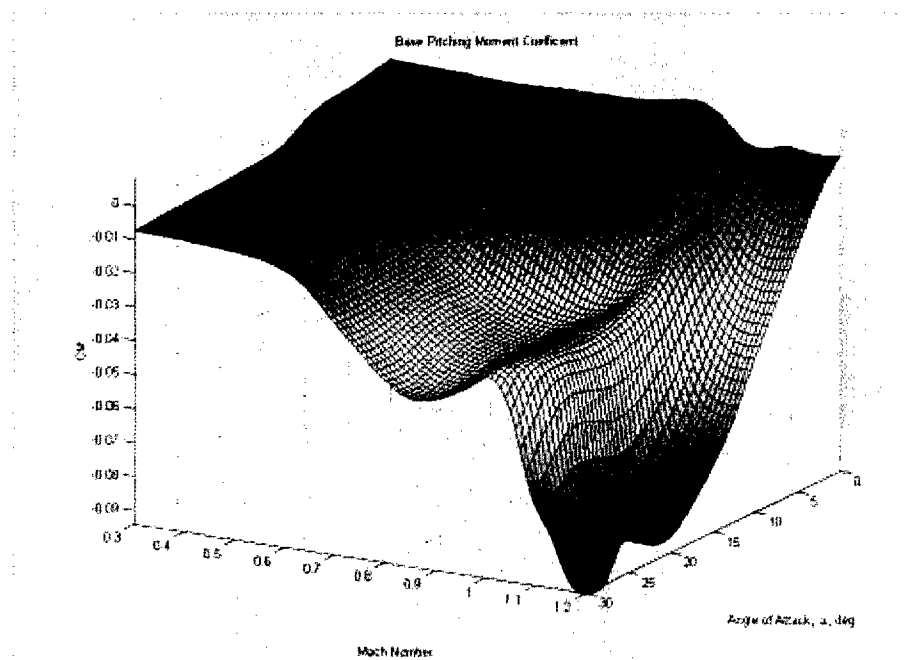

Figure 7 – Base Pitching Moment Coefficient Neural Network Approximation
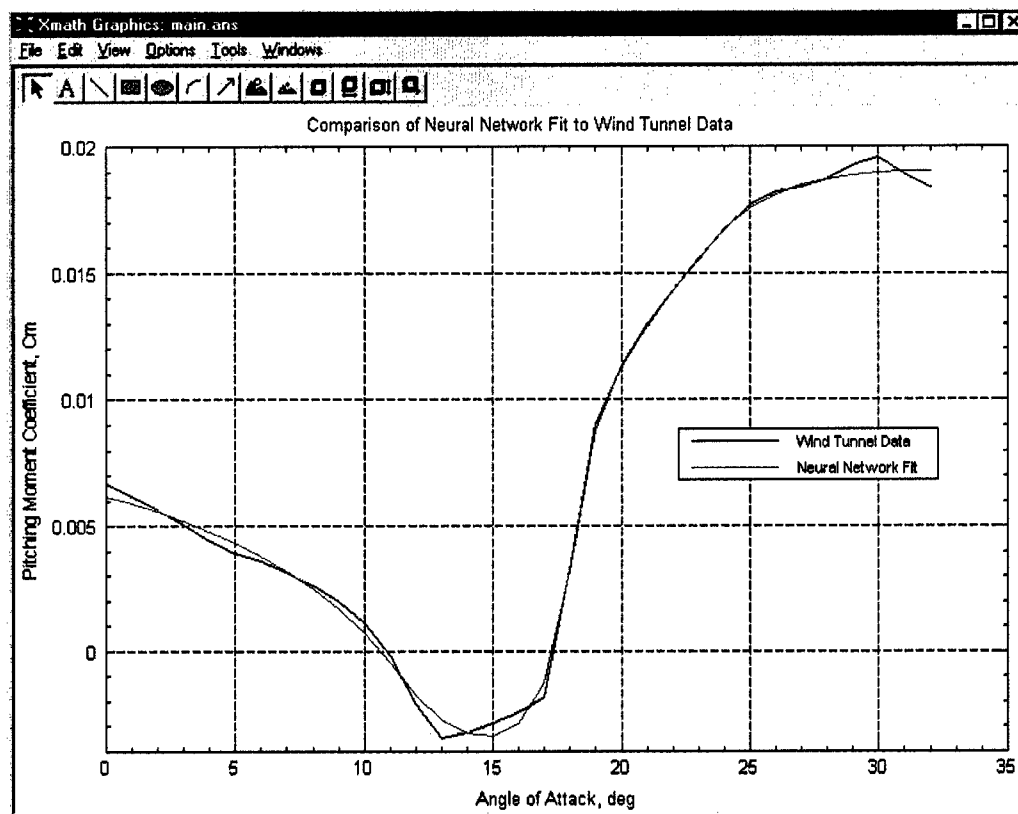
Figure 8 – Base Pitching Moment Coefficient Accuracy

2.4.2. Dynamic Inversion Controller with Dynamic Control Allocation

Current CAO methods are static in nature, which does not allow them to properly address stability. The derivation of these static CAO methods neglects actuator dynamics by assuming that they are fast relative to the aircraft closed-loop dynamics. This assumption allows treatment of the actuator dynamics as unmodeled dynamics.

However if all effector dynamics have different bandwidths from slow to fast relative to the desired closed-loop response, then a dynamical systems approach is required to properly address stability. As shown by the robustness results in section 2.3, there is a need to directly accommodate actuator dynamics within the CAO flight control system element. A preliminary technique for directly incorporating individual effector bandwidths into the CAO design procedure is proposed. Although this technique still requires matrix inversion and satisfies rate and magnitude limits, it requires less computation than other proposed methods [11] where a high-gain compensator is designed for each effector that may adversely affect unmodelled dynamics. The frequency- and time-domain results show promise of this dynamic CAO technique, however additional research is still required to incorporate position and rate limits.

20

Derivation of the proposed dynamic CAO begins by combining the dynamic inversion control derivation eq.(7) and the actuator dynamics model eq.(10) to get the following relationship

(24) $\quad TBG_{act}(s)\delta_c = m$

where $G_{act}(s)$ is a (11 x 11) diagonal matrix of simple first-order actuator dynamics of the following form

(25) $\quad G_{act}(s) = diagonal\left(\dfrac{a_1}{s+a_1}, \dfrac{a_2}{s+a_2}, \dfrac{a_3}{s+a_3}, ..., \dfrac{a_{11}}{s+a_{11}}\right).$

The dynamic control allocation eq.(24) may be inverted if $G_{act}(s)$ is proper[5] which in its present form is not. An augmentation matrix may be included as follows

(26) $\quad \delta_c = \widetilde{G}\widetilde{\delta}$

where

(27) $\quad \widetilde{G}(s) = diagonal\left(\dfrac{s+100}{100}, \dfrac{s+100}{100}, \dfrac{s+100}{100}, ..., \dfrac{s+100}{100}\right)$

$\widetilde{G}$ is chosen fast relative to the desired closed-loop response as well as the highest important effector bandwidth chosen so as not to excite unmodeled high-frequency dynamics. Now the dynamic control allocation relationship becomes

(28) $\quad TBG_{act}(s)\widetilde{G}(s)\widetilde{\delta} = TBG_{aug}(s)\widetilde{\delta} = m$

Assume that $TBG_{aug}(s)\widetilde{\delta} = m$ has the following minimal state-space model

(29) $\quad \begin{aligned} \dot{\chi} &= \Lambda\chi + \Gamma\widetilde{\delta} \\ m &= H\chi + J\widetilde{\delta} \end{aligned}$

Let $J^{\#}$ be the pseudo-inverse of J which exists since $G_{aug}(s)$ is proper. The inverse of eq.(29) becomes [5]

(30) $\quad \begin{aligned} \dot{\chi} &= \left(\Lambda - \Gamma J^{\#}H\right)\chi - \Gamma J^{\#}m \\ \widetilde{\delta} &= J^{\#}H\chi + J^{\#}m \\ J^{\#} &= J^T\left(JJ^T\right)^{-1} \\ \delta_c &= \widetilde{\delta} \end{aligned}$

Note that there is no direct accommodation of position and rate limits. Inclusion of limits will be the subject of future research. Equation (30) becomes the dynamic CAO that is implemented for the following analysis.

Robust stability and performance analysis was performed with the dynamic CAO and parametric OBM uncertainty. The similar and dissimilar actuator sets were both used. The results are presented with the corresponding static CAO cases for comparison.

The robust stability and performance μ-analysis results for the dynamic control allocation with similar actuators are shown in Figure 9 and Figure 10 respectively. Note that the dynamic CAO is only slightly more robust than static CAO at low dynamic pressure with the similar actuator set.

The robust stability and performance μ-analysis results for the dynamic control allocation with dissimilar actuators are shown in Figure 11 and Figure 12, respectively. Note that the benefits of the dynamic control allocation become more apparent with the dissimilar actuator set. The robustness is significantly improved by using the dynamic control allocation. Again, the most dramatic improvement is observed at the low dynamic pressure conditions.
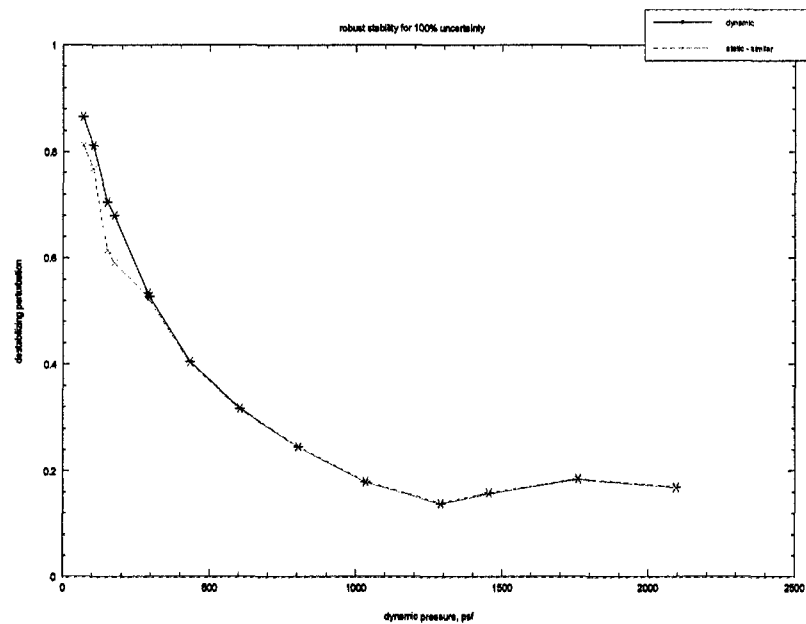
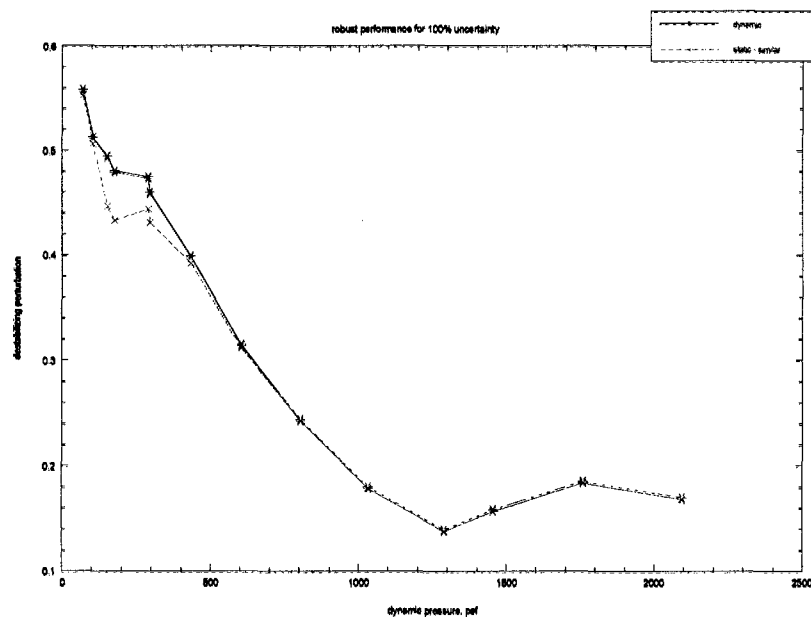**Figure 9 - Robust Stability of Dynamic Control Allocation, Similar Actuators**

Figure 10 - Robust Performance of Dynamic Control Allocation, Similar Actuators
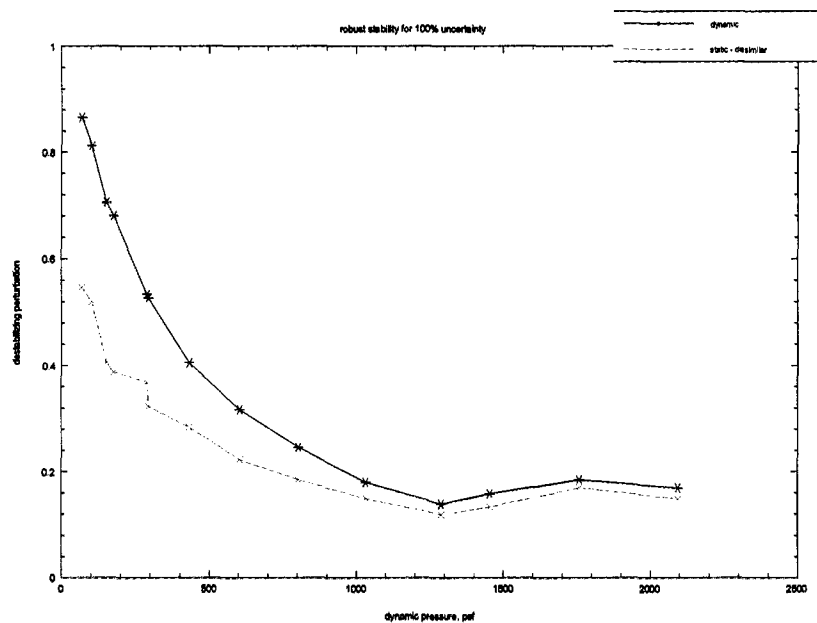
Figure 11 - Robust Stability of Dynamic Control Allocation, Dissimilar Actuators
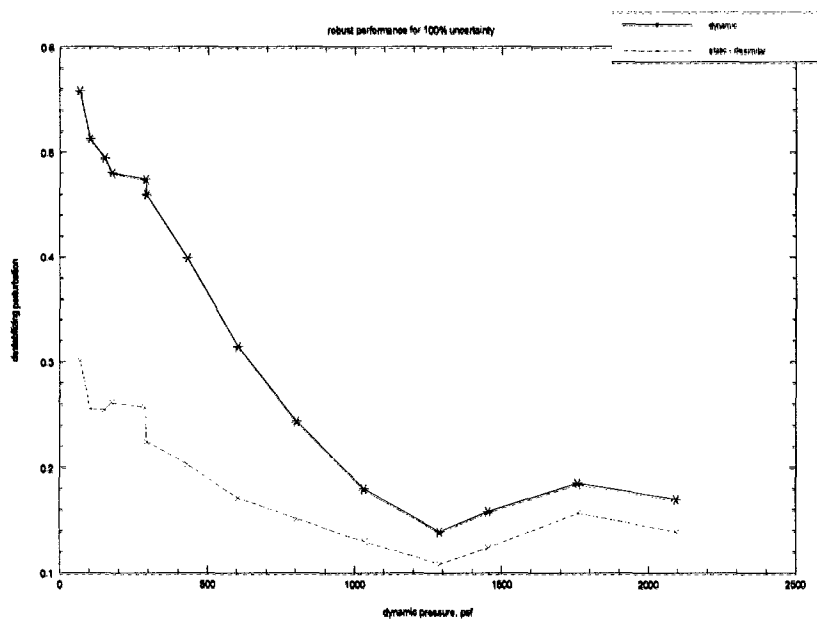

Figure 12 - Robust Performance of Dynamic Control Allocation, Dissimilar Actuators

2.5. Reconfigurable Flight Control Design

The reconfigurable control design task includes development of an OBM update synthesis method. This update method provides the flight control system with the ability to reconfigure and adapt to changing conditions. Neural network and parameter identification theory is used to construct a model update that is consistent with the nonlinear aerodynamic model architecture. Upcoming sections will present 1) background information concerning the problems with using linear parameter information to update nonlinear aerodynamic databases, 2) a method for parameter identification (PID), and 3) the application of PID to the nonlinear aerodynamic build-up equations. Further, a process is presented to reduce PID and aircraft state information for the development of an OBM update increment.

2.5.1. Background

Aerodynamic models generated from wind tunnel tests and CFD analyses have inherent errors associated with them. Additionally, OBM errors are introduced by approximations, such as those in section 2.4.1, to satisfy on-line computational requirements. Regardless of the quality of the OBM, errors are not typically manifest until flight test. While many methods exist to provide improved estimates of the vehicle dynamics either on-line [6],[9] or off-line [7],[10], these methods provide only stability derivative information. These derivatives, while useful for analysis purposes, cannot be used to directly modify the nominal OBM. An approach that uses the fundamental nonlinear aerodynamic build-up equations offers the opportunity to identify errors in the aerodynamic database and develop OBM update increments based upon this information.

2.5.2. Problem Definition

The fundamental problem addressed by this task is demonstrated by considering the following simplified nonlinear pitch rate dynamics

$$(31) \quad \dot{Q} = \Delta M_Q(\alpha)Q + M_{base}(\alpha) + \Delta M_{latdir}(\alpha, \beta) + \Delta M_1(\alpha, \delta_1) + \Delta M_2(\alpha, \delta_2)$$

This relationship assumes a constant Mach number and contains only the aerodynamic contributions to the pitch. A linearized representation of these pitch rate dynamics to facilitate the use of some control design and analysis methods is expressed as

$$(32) \quad \dot{Q} = M_Q Q + M_\alpha \alpha + M_\beta \beta + M_{\delta_1} \delta_1 + M_{\delta_2} \delta_2 + M_{bias}$$

The Modified Sequential Least Squares (MSLS) PID method has been used in adaptive control systems[9] to estimate the stability derivatives M_Q, M_α and M_β, the control derivatives M_{δ_1} and M_{δ_2} and the bias, M_{bias} given measurements of $\dot{Q}$, Q, α, β, δ_1 and δ_2. MSLS requires a priori values of M_Q, M_α, M_β, M_{δ_1}, M_{δ_2},

25

and M_{bias} to improve the estimates during times of insufficient excitation, for example. *A priori* values of the estimated parameters can be stored directly within the OBM in derivative form or computed as the appropriate derivatives of the nonlinear aerodynamic forces and moments. When utilizing the nonlinear representation expressed in eq.(31), the stability derivatives are computed as

$$(33) \quad M_Q = \Delta M_Q(\alpha)\Big|_\alpha \,, \quad M_\alpha = \frac{\partial M_{base}(\alpha)}{\partial \alpha}\Bigg|_\alpha \,, \quad M_\beta = \frac{\partial \Delta M_{latdir}(\alpha,\beta)}{\partial \beta}\Bigg|_{\alpha,\beta}$$

and the control derivatives are found from

$$(34) \quad M_{\delta_1} = \frac{\partial \Delta M_{\delta_1}(\alpha,\delta_1)}{\partial \delta_1}\Bigg|_{\alpha,\delta_1} \,, \quad M_{\delta_2} = \frac{\partial \Delta M_{\delta_2}(\alpha,\delta_2)}{\partial \delta_2}\Bigg|_{\alpha,\delta_2}$$

The bias term is then computed by subtracting from the total pitching moment the stability derivatives times the state values and the control derivatives times the control deflection, i.e.

$$(35) \quad M_{bias} = M_{tot} - M_Q Q - M_\alpha \alpha - M_\beta \beta - M_{\delta_1}\delta_1 - M_{\delta_2}\delta_2$$

where M_{tot} is computed from the total nonlinear aerodynamic pitching moment

$$(36) \quad M_{tot} = \Delta M_Q(\alpha)Q + M_{base}(\alpha) + \Delta M_{latdir}(\alpha,\beta) + \Delta M_1(\alpha,\delta_1) + \Delta M_2(\alpha,\delta_2)$$

Ideally, it is desired to use information from the PID process to update the OBM. However, recall that our compact OBM synthesis method in section 2.4 stores estimates of the nonlinear elements. These nonlinear element estimates accommodate nonlinear control design approaches, like dynamic inversion, as well as linear design approaches through the relationships in eqs. (33)-(36). If information from the PID process is to be used to update the terms in the nonlinear aerodynamic buildup equations, however, relationships from the linear parameters to the nonlinear aerodynamic terms need to be established. Unfortunately, finding such a relationship is not straightforward without significant modification of the problem. Note that the relationships in eqs. (33)-(36) only provide relationships from the nonlinear terms to the linear parameters.

To illustrate the challenge, consider the example in Figure 13. The solid line represents a nonlinear function such as base pitching moment as a function of angle of attack. The nonlinear function is determined by fitting the underlying aerodynamic data, represented by the solid markers. If MSLS identifies a change in the slope, M_α, at some angle of attack as represented by the dashed line, it is not clear how to modify the underlying base pitching moment function to reflect the new derivative. In this example, a quandary exists since, to achieve the slope change, either p3 could be reduced or p4 could be increased or both a reduction in p3 and simultaneous increase in p4 could achieve the desired effect.

The following section proposes an improved approach. This approach leverages our experience with MSLS PID and is consistent with the nonlinear aerodynamic formulation to accommodate OBM updates and nonlinear control methods.
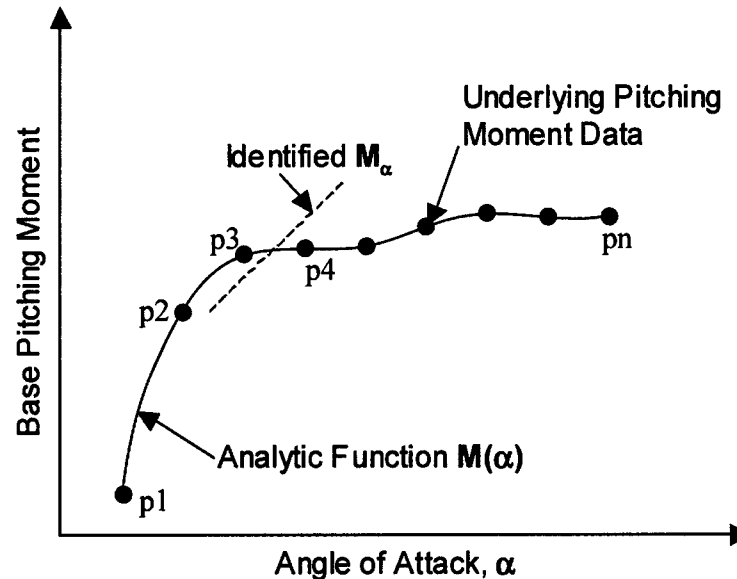
Figure 13 - Difficulty in Utilizing Linear Derivative Information

2.5.3. Proposed Solution

Consider the identification of multiplicative factors, K_i, in the example nonlinear aerodynamic buildup equation

(37) $\dot{Q} = K_1 \Delta \overline{M}_Q(\alpha)Q + K_2 \overline{M}_{base}(\alpha) + K_3 \Delta \overline{M}_{latdir}(\alpha, \beta) + K_4 \Delta \overline{M}_1(\alpha, \delta_1) + K_{bias}$

where the term K_{bias} is added to account for situations where one or more of the nonlinear terms is zero. The bar over a term indicates a nominal on-board model representation. In this equation, the first term represents pitching moment due to pitch rate, the second is the controls-neutral or base pitching moment and the third represents changes in the base pitching moment due to sideslip. The fourth term is the incremental pitching moment due to a control effector deflection. Other control effectors could be added as the configuration warrants. For this relationship, the nominal value of K_1 through K_4 is one and the nominal value of K_{bias} is zero. Note that since these are dimensionless multipliers, they can be applied directly to the underlying aerodynamic coefficients without modification.

For demonstration purposes, consider the simple system where

$\dot{Q} = K_1 \overline{M}_{base}(\alpha) + K_{bias}$ in which the true value of M_{base} is defined as

$M_{base}(\alpha) = -(0.2 + 0.1\alpha)$, $\alpha \in [0,10]$. The initial on-board model of $\overline{M}_{base}$ is

$\overline{M}_{base}(\alpha) = 0.9 \quad \forall \alpha \in [0,10]$.

The MSLS algorithms were implemented for this simple example in Xmath to facilitate multiple simulation passes rapidly. A pass consists of sweeping alpha from 0 to 10 degrees and using the parameter estimates to generate an increment. On the second and subsequent passes, this increment is added to the previous model. In the event the model is approximately zero and the associated K_1 is not zero, K_{bias} is added to the model. With each pass, the value of K_1 should approach 1 and the value of K_{bias} should approach 0 and the estimate of the on-board model should approach the true value. Figure 14 presents the evolution K_1 and the on-board model $\overline{M}_{base}(\alpha)$ as fifteen passes are made. Examination of this evolution shows that the model $\overline{M}_{base}(\alpha)$ approaches the true value, even though the original model had the wrong sign and that the value of K_1 is indeed approaching unity.
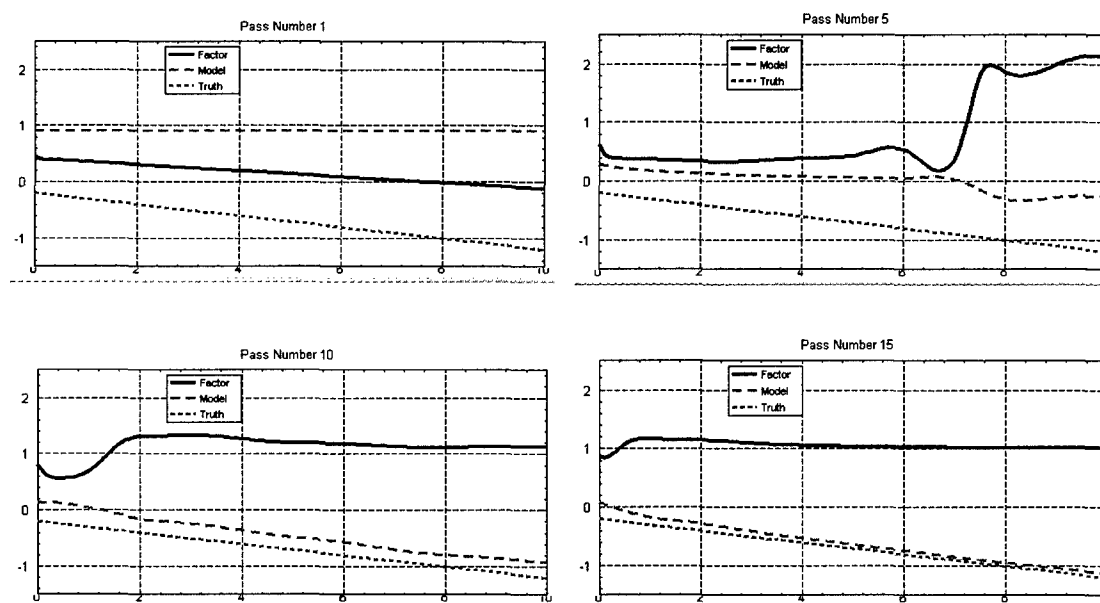
Figure 14 - Simple Example of On-Board Model Information Determination

Next, consider a more complicated dynamical system. A Systembuild simulation of the Innovative Control Effectors (ICE) vehicle is presented in Figure 15. The simulation consists of aerodynamic lookup tables for the aircraft aerodynamics and propulsion. Dynamic inversion provides the on-line control design and the control allocation subsystem is an implementation of the constrained pseudo inverse. The actuator models consist of first and second order linear transfer functions with considerations for hinge moment-limited rate limits. The OBM consists of a series of neural network fits of the aerodynamic and propulsive databases. The MSLS PID algorithms are implemented in their own separate subsystem. The OBM subsystem was modified to provide spatial constraints that contain the most recent multiplicative factor information.

28

Figure 15 - ICE Vehicle Systembuild Representation

In this example, the equation that represents the true pitching moment equation is represented as

(38) $\dot{Q} = \Delta M_Q(\alpha)Q + M_{base}(\alpha) +$

$\Delta M_{lel}(\alpha, \delta_{lel}) + \Delta M_{rel}(\alpha, \delta_{rel}) + \Delta M_{pflap}(\alpha, \delta_{pflap})$

The term that accounts for the change in pitching moment due to sideslip has been dropped since the simulations will focus on pure pitching maneuvers, hence little or no sideslip. The model equation that is used in the OBM model module and by the MSLS PID algorithms contains the multiplicative factors

(39) $\dot{Q} = K_1 \Delta \overline{M}_Q(\alpha)Q + K_2 \overline{M}_{base}(\alpha) +$

$K_3 \Delta \overline{M}_{lel}(\alpha, \delta_{lel}) + K_4 \Delta \overline{M}_{rel}(\alpha, \delta_{rel}) + K_5 \Delta \overline{M}_{pflap}(\alpha, \delta_{pflap}) + K_{bias}$

As mentioned before, the multiplicative factors have nominal values of one and the bias factor has a nominal value of zero. Multiple iterations are performed to facilitate identification of the factors. Figure 16 presents a description of the steps taken during an iteration. After the first simulation run, the time history data is processed and the data is used to train a neural network. After training, the neural network is used in the first iteration in the following manner.

29

$$(40) \quad \dot{Q} = K_1 K_1^0(\alpha)\Delta \overline{M}_Q(\alpha)Q + K_2 K_2^0(\alpha)\overline{M}_{base}(\alpha) + K_3 K_3^0(\alpha,\delta_{lel})\Delta \overline{M}_{lel}(\alpha,\delta_{lel}) +$$

$$K_4 K_4^0(\alpha,\delta_{rel})\Delta \overline{M}_{rel}(\alpha,\delta_{rel}) + K_5 K_5^0(\alpha,\delta_{pflap})\Delta \overline{M}_{pflap}(\alpha,\delta_{pflap}) + K_{bias}$$
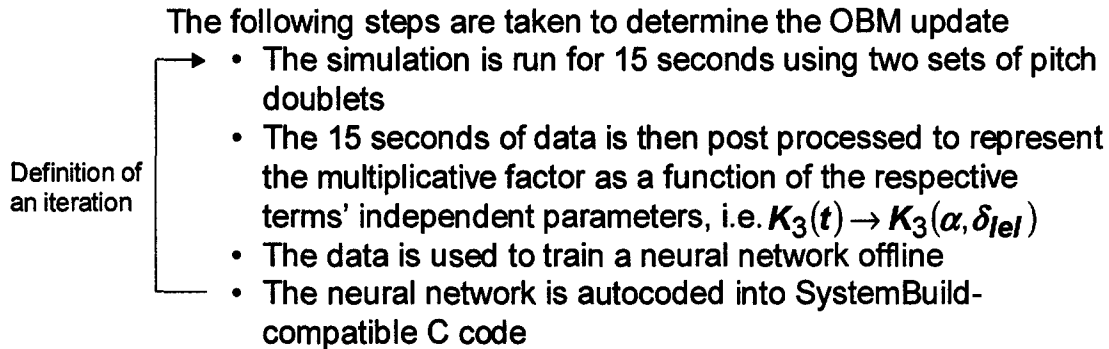
The following steps are taken to determine the OBM update

Definition of an iteration

- The simulation is run for 15 seconds using two sets of pitch doublets
- The 15 seconds of data is then post processed to represent the multiplicative factor as a function of the respective terms' independent parameters, i.e. $K_3(t) \rightarrow K_3(\alpha,\delta_{lel})$
- The data is used to train a neural network offline
- The neural network is autocoded into SystemBuild-compatible C code

Figure 16 - Iteration Definition

The superscript 0 denotes the iteration origin of the data that is post processed and used to train the neural network. Note the dependency of the multiplicative factor is the same as the term that it multiplies. Again, after post processing the data and training a neural network, the equation used for parameter identification on the second iteration is as follows.

$$(41) \quad \dot{Q} = K_1\left(K_1^0(\alpha)K_1^1(\alpha)\right)\Delta \overline{M}_Q(\alpha)Q + K_2\left(K_2^0(\alpha)K_2^1(\alpha)\right)\overline{M}_{base}(\alpha) +$$

$$K_3\left(K_3^0(\alpha,\delta_{lel})K_3^1(\alpha,\delta_{lel})\right)\Delta \overline{M}_{lel}(\alpha,\delta_{lel}) + K_4\left(K_4^0(\alpha,\delta_{rel})K_4^1(\alpha,\delta_{rel})\right)\Delta \overline{M}_{rel}(\alpha,\delta_{rel}) +$$

$$K_5\left(K_5^0(\alpha,\delta_{pflap})K_5^1(\alpha,\delta_{pflap})\right)\Delta \overline{M}_{pflap}(\alpha,\delta_{pflap}) + K_{bias}$$

Here the superscript 1 denotes that the data used to train the neural networks came from the first iteration. The process can be repeated until the multiplicative factors lie sufficiently close to unity.

2.5.4. Results

The ICE simulation is modified such that the only pitch control effectors are the left and right elevons and the pitch flap. The simulation also contains a modification to the truth data such that left elevon effectiveness is decreased by 50%. The effects of this modification are not modeled in either the roll or yaw axis, therefore only the pitch axis is affected. Two aggressive longitudinal stick doublets, shown in Figure 17, are used to excite the simulation with the total simulation time lasting fifteen seconds. The goal is to verify that the proposed method identifies a multiplicative factor of 0.5 on the left elevon term while other multiplicative factors remain close to one.
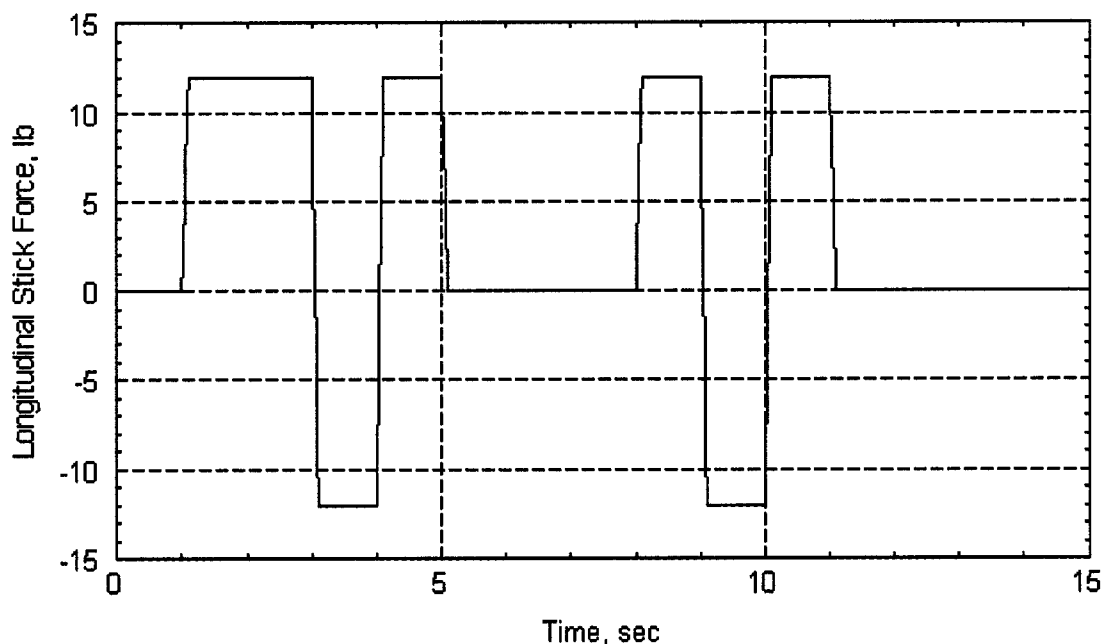
Figure 17 - Simulation Longitudinal Stick Inputs

After the appropriate modifications to the simulation are made and the simulation is run with the longitudinal stick input, an example of the multiplicative factor K_1 as a function of time is shown in Figure 18. However, the aerodynamic term associated with factor K_1 is the base pitching moment and it is represented as a function of angle of attack.

Figure 19 presents factor K_1 as a function of angle of attack using the angle of attack response. An Xmath script is used to sort K_1 as a function of angle of attack and averages those values over 0.5° increments of angle of attack. For example, the values of K_1 are averaged between 5.5° and 6° angle of attack, resulting in an average value of K_1 at 5.75° angle of attack. The result of this post-processing is shown in Figure 20 for K_1. These data points are then used as neural network training data. A similar post processing is performed on data for the other multiplicative factors, noting that the control effector factors are functions of angle of attack and their respective control deflection. Therefore, a two-step process is used for those terms, first sorting data as a function of angle of attack and second as a function of control deflection. The data are then averaged over 0.5° increments in angle of attack and over 0.5° increments in effector deflection.
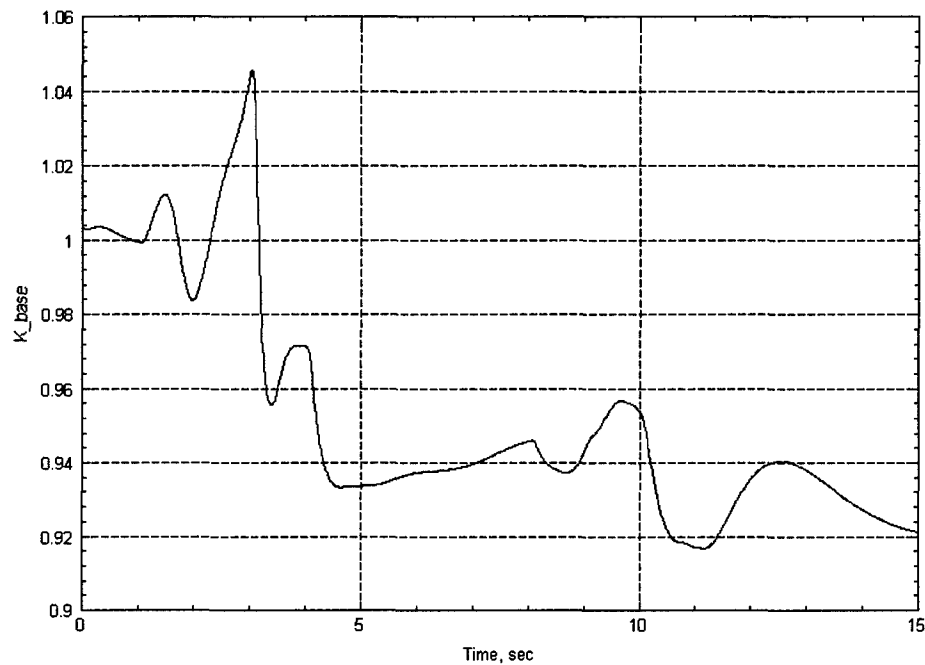
31

Figure 18 - Base Pitching Moment Multiplicative Factor from Iteration 0 as a Function of Time
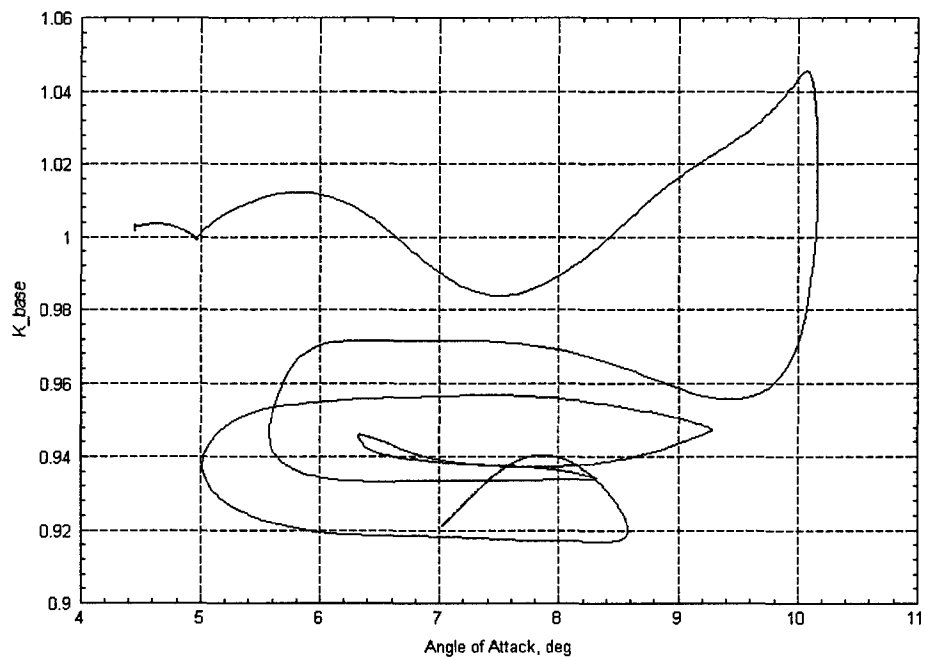

Figure 19 - Base Pitching Moment Multiplicative Factor from Iteration 0 as a Function of Angle of Attack
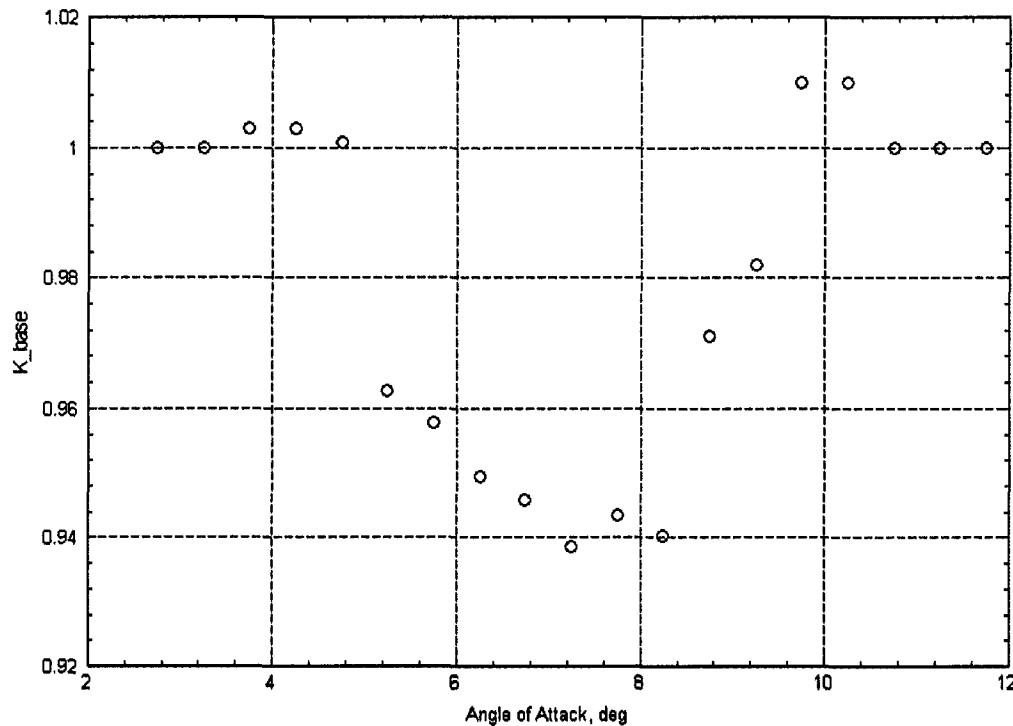
32

Figure 20 - Base Pitching Moment Multiplicative Factor After Post Processing

The multiplicative factor data associated with each of the terms is used to train a two-hidden layer feed forward neural network. Each neural network is then autocoded into C code that is compatible with the Systembuild user code block. The simulation is then rerun using the same pilot input, but the on-board model used by the dynamic inversion controller and by the MSLS PID code is modified by the multiplicative factor learned from the 0th iteration. The same process is again carried out: post processing the multiplicative factor data and generating the neural network Systembuild code until the third iteration is performed. Upcoming graphics present the multiplicative factors identified by MSLS after completion of a simulation run.

The multiplicative factor associated with the left elevon, K_3, had its truth term multiplied by a factor of 50% without modification to the OBM. Therefore, the cumulative effect of the factors should approach 0.5 as the number of iterations increases while the multiplicative factor identified by MSLS should approach one. The next two sections present the simulation results to the same set of inputs and flight condition. The first section has no noise imposed on the measurements and the second uses noise characteristics similar to those experienced by the F-16 VISTA aircraft.

2.5.4.1.   Simulations Without Measurement Noise

Figure 21 presents a sequence of four plots, each one generated by the simulation after its respective iteration. Each plot presents the true value of the pitching moment

increment as a dashed line and the on-board model representation as a solid line. The red line presents the multiplicative factor identified by MSLS. By the third iteration, the value of K_3 approached unity, indicating that the cumulative effect of

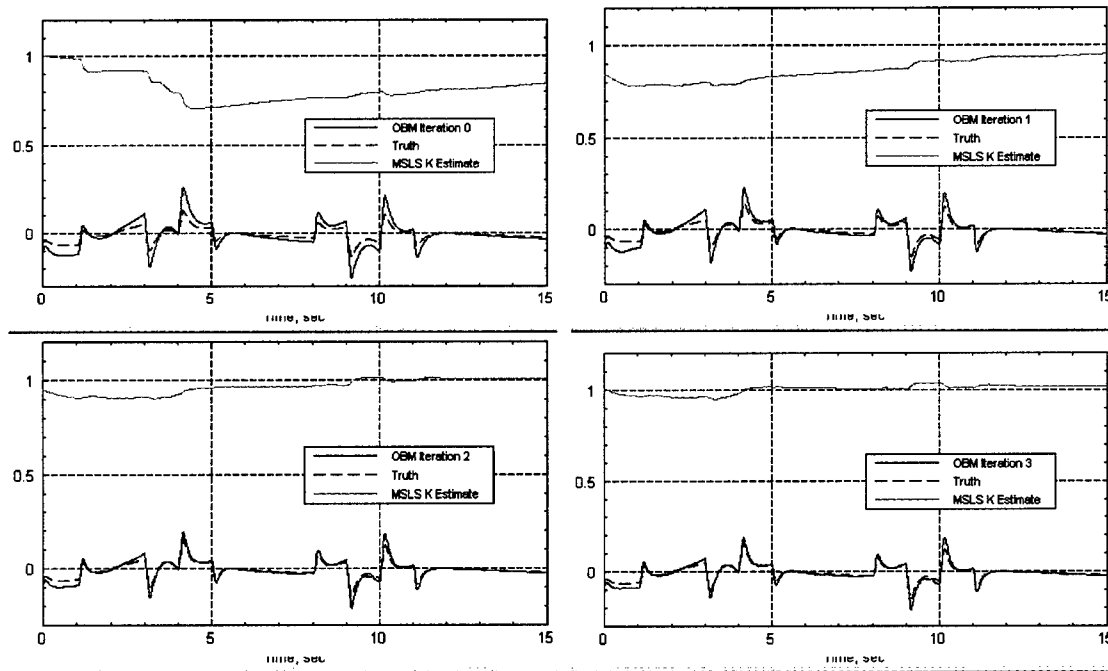$K_3^0 K_3^1 K_3^2$ has approximately reached a steady state value.

Figure 21 - Pitching Moment due to Left Elevon Deflection and Associated Multiplicative Factor Estimate

The next three graphics, Figure 22, Figure 23, and Figure 24 present the cumulative effects of the left elevon multiplicative factors as contour diagrams. Figure 22 presents the on-board multiplier K_3^0 that is formed by post-processing the data from Iteration 0. Similarly, Figure 23 and Figure 24 present the multipliers $K_3^0 K_3^1$ and

$K_3^0 K_3^1 K_3^2$, respectively, from Iterations 1 and 2. Each of these plots cover the same ranges of input space, -5° to 5° left elevon deflection and 0° to 15° angle of attack. Note that, since not all of the input space was covered by the respective simulation run, holes or bumps occur within the center region. As the on-board model is improved, the controller uses the effector in a slightly different manner resulting in changes in the covered input space. For example, as shown in Figure 25, the left elevon covers a range of –2.5° to 3° during the first simulation (Iteration 0) and covers a range of –3° to 3.5° during the fourth simulation (Iteration 3). The alternating bright and dark red regions around the edges of the diagrams are artifacts

34

of the neural network fits and can be removed through modifications to the neural network architecture.
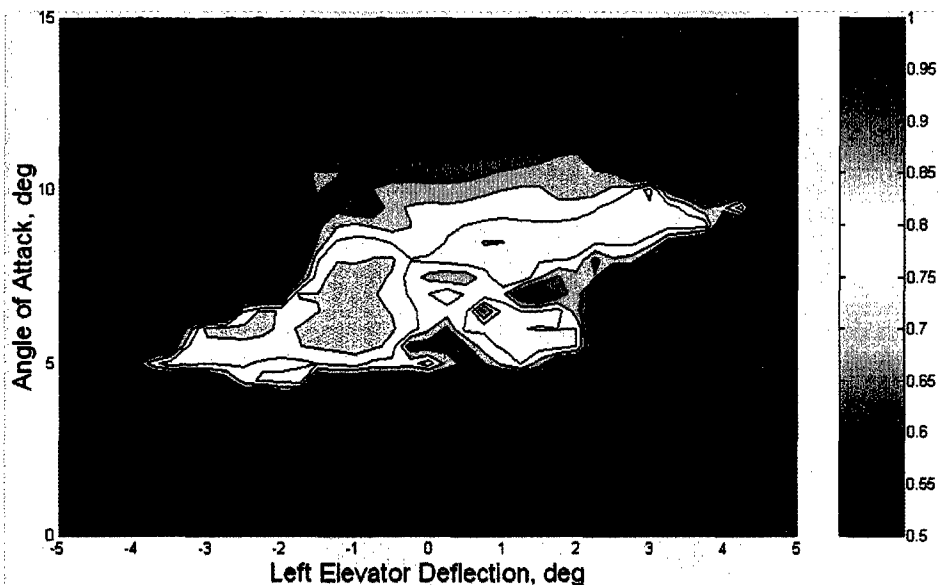

Figure 22 - Post-Processed Left Elevator Multiplicative Factor After Iteration 0.
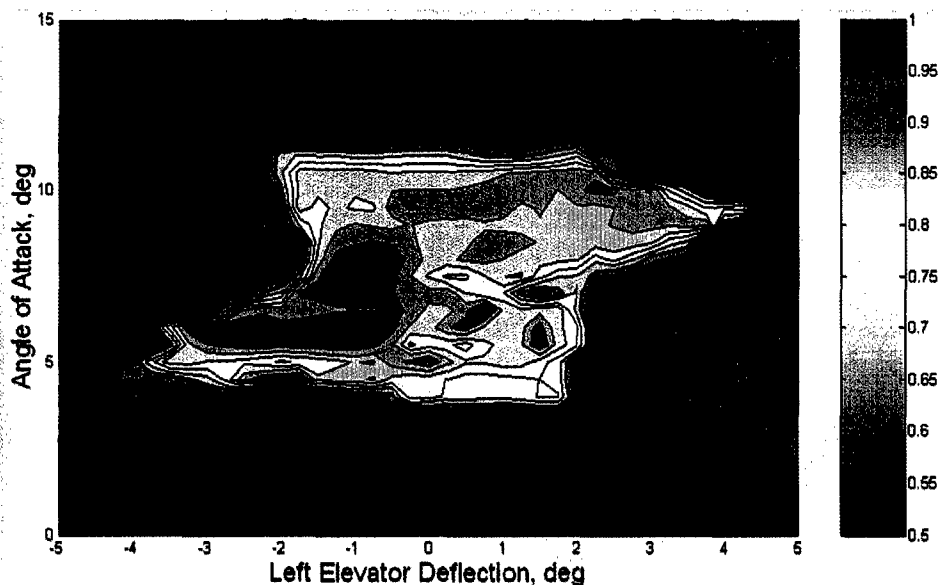

Figure 23 - Post-Processed Left Elevator Multiplicative Factor After Iteration 1.
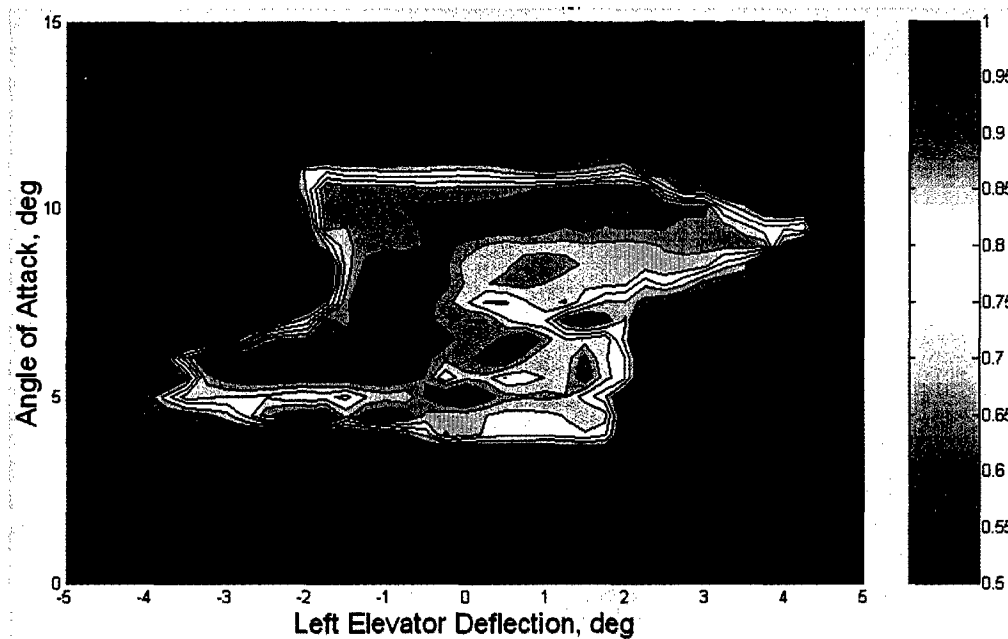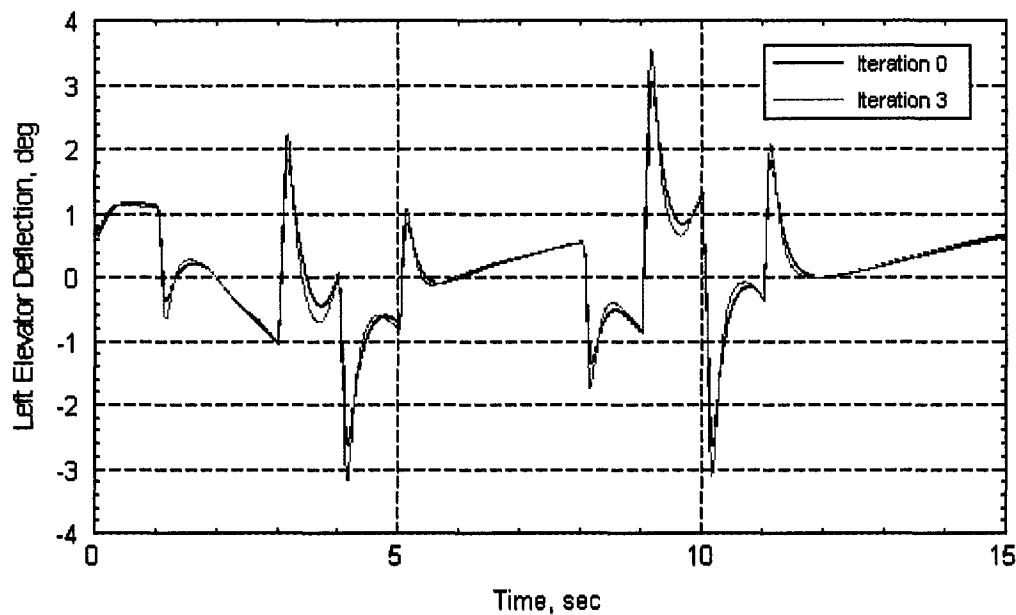
35

Figure 24 - Post-Processed Left Elevator Multiplicative Factor After Iteration 2.

Figure 25 - Comparison of the Left Elevon Time History Between Iterations 0 and 3.

The cumulative effects of the multiplicative factors determined from Iterations 0, 1, and 2 should produce values close to 0.5, which is approximately the case as shown

36

in Figure 24. For regions that have not been explored, the multiplicative factor has remained unity, maintaining the nominal on-board model for those regions.

The simulation time histories and the associated multiplicative factor, K_4, for the right elevon are shown in Figure 26. Note that the time history for Iteration 0 shows a very close agreement between the on-board model and truth. Therefore, the nominal value of one is appropriate for K_4. The values of K_4 estimated by MSLS has a value of approximately 0.9 for Iteration 0 and approximately 0.95 for Iteration 1 before reaching unity during Iterations 2 and 3. Figure 27 shows the contour diagram for the cumulative effects of $K_4^0 K_4^1 K_4^2$. Note that, although the truth model and the on-board model are in close agreement and hence the values of the multiplicative factors should have been one, MSLS spread some of the loss of effectiveness of the left elevon on the right elevon. Much of the region of input space covered by this maneuver concerning the right elevon has values in the 0.8 to 0.9 levels with a small region reaching down in to 0.7.
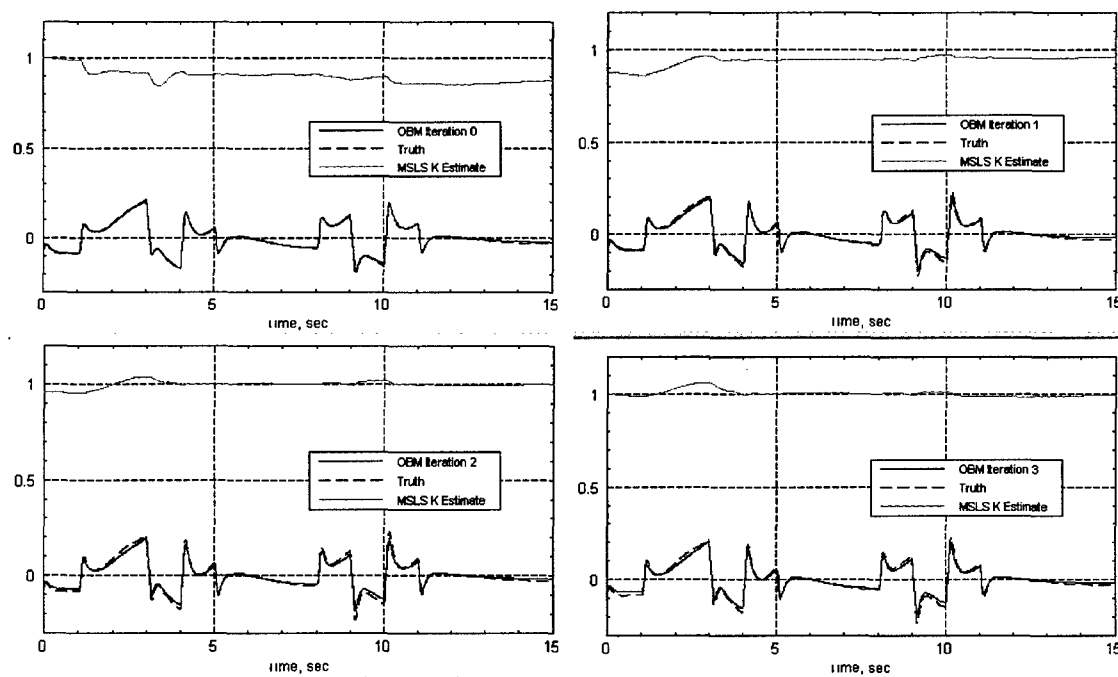

Figure 26 - Pitching Moment due to Right Elevon Deflection and Associated Multiplicative Factor Estimate.
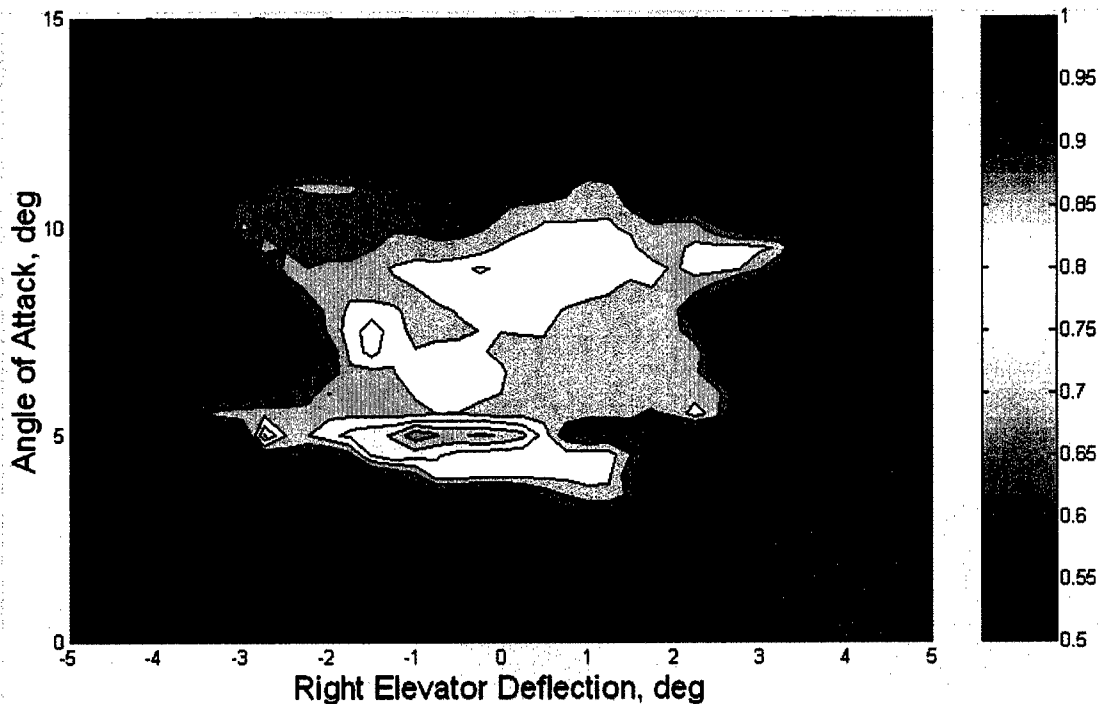
37

Figure 27 - Post-Processed Right Elevator Multiplicative Factor After Iteration 2.

The simulation time histories for the pitch flap for Iterations 0-3 are shown in Figure 28. The contribution of the pitch flap to the total pitching moment is relatively small when compared to either of the elevons. The multiplicative factors identified during the simulations typically lie close to unity, with the smallest values no lower than 0.95. Since the pitch flap truth model and the on-board models are in good agreement, the multiplicative factors should always lie close to one. Although the on-board model is modified slightly through the learning process, qualitatively speaking, there is no practical degradation in the on-board model. The contour diagram representing the cumulative effect of $K_\delta^0 K_\delta^1 K_\delta^2$ is shown in Figure 29. Since the MSLS estimates are always close to unity, the diagram shows that the values of $K_\delta^0 K_\delta^1 K_\delta^2$ are close to one within the region of input space covered by the maneuvers.
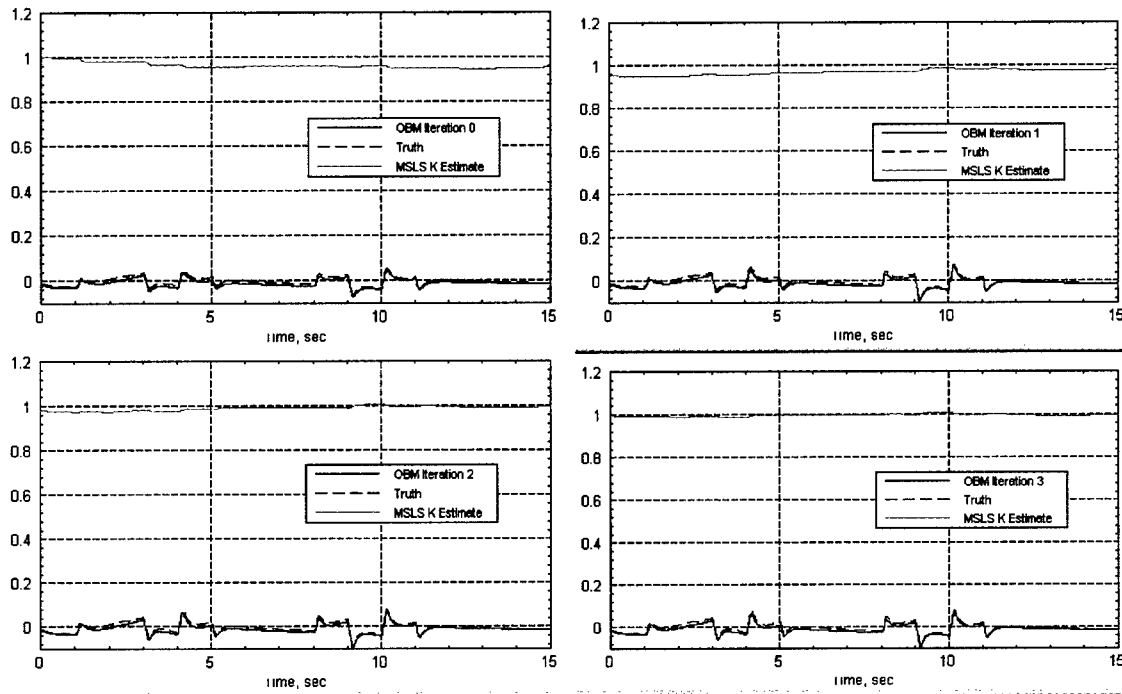
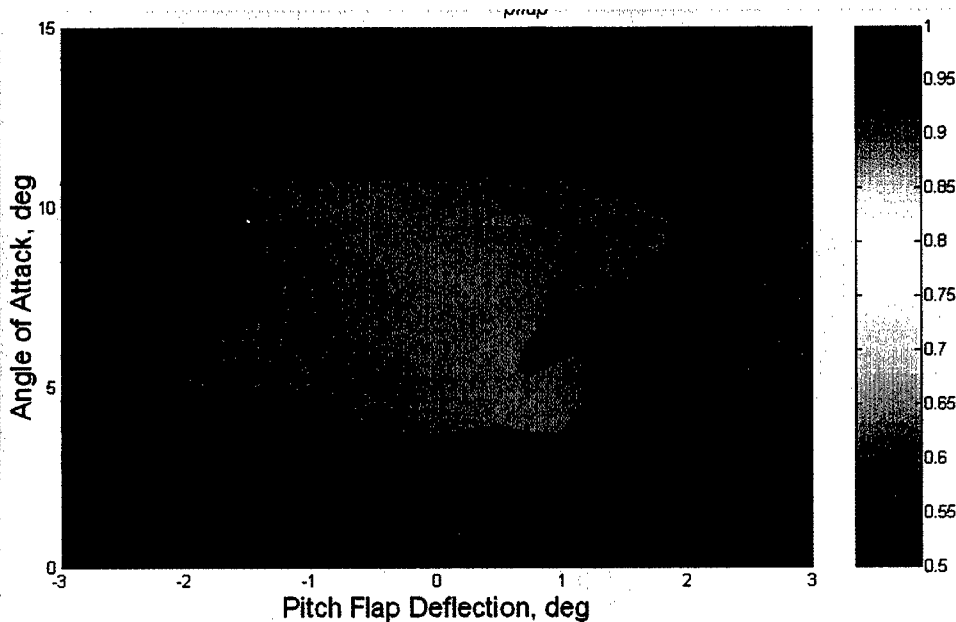**Figure 28 - Pitching Moment due to Pitch Flap Deflection and Associated Multiplicative Factor Estimate.**

Figure 29 - Post-Processed Pitch Flap Multiplicative Factor After Iteration 2.

The upcoming discussions focus on the two terms in the pitching moment equation that are functions of only one variable, angle of attack. The simulation time histories

39

showing the true and on-board model representations and the associated multiplicative factor for the base pitching moment term are shown in Figure 30. The MSLS estimates of the multiplicative factor for each simulation stay near one (0.9 to 1.05) and there is no qualitative change in the overall accuracy of the base pitching moment. Figure 31, which presents the cumulative effects of $K_2^0 K_2^1 K_2^2$, shows that the factor is approximately 0.9 at 4° and between 6° and 8° angle of attack.
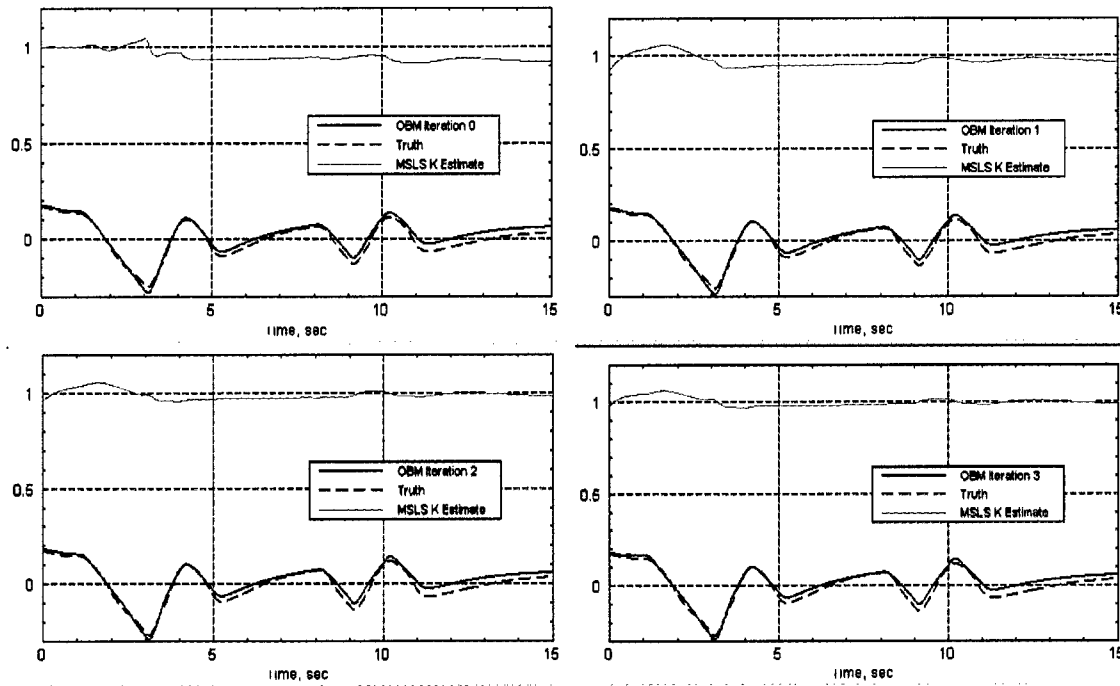

Figure 30 - Base Pitching Moment and Associated Multiplicative Factor Estimate.

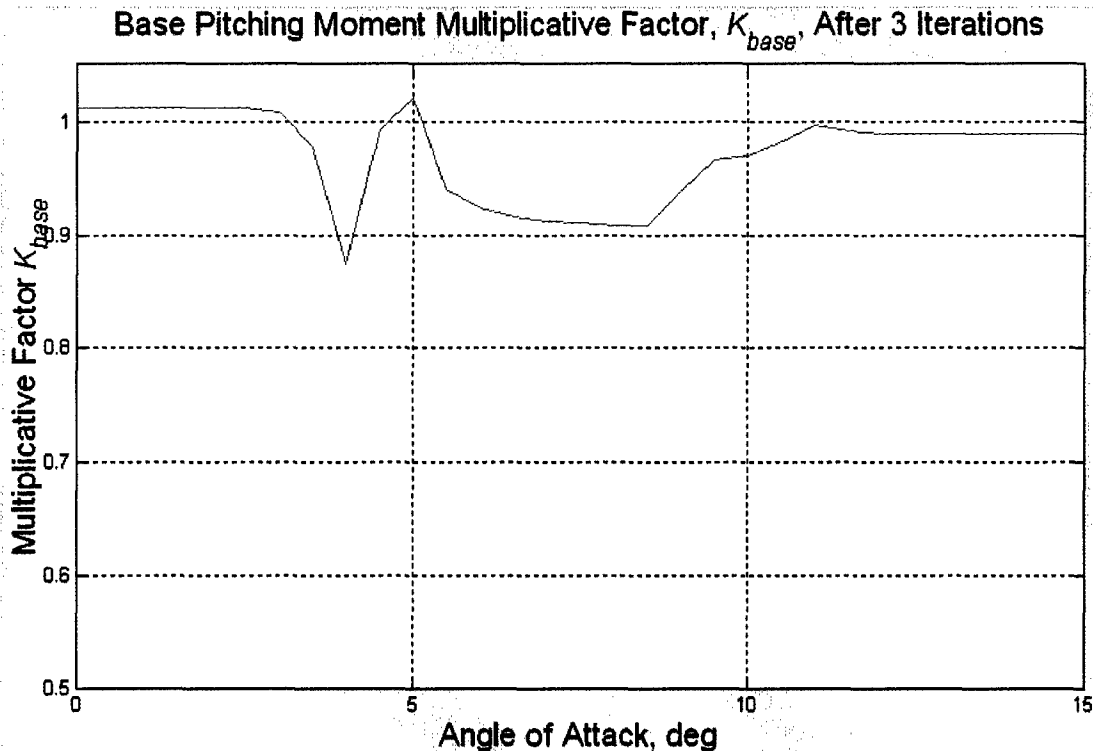Base Pitching Moment Multiplicative Factor, K_{base}, After 3 Iterations

**Figure 31 - Post-Processed Base Pitching Moment Multiplicative Factor
After Iteration 2.**

The final factor multiplies the pitching moment due to pitch rate term in the aerodynamic buildup equation. Figure 32 shows simulation time histories of the truth and on-board model values of the pitching moment due to pitch rate (multiplied by the pitch rate) as well as the associated multiplicative factor. This graphic shows that the original on-board model has excellent agreement with the truth, but that MSLS attributed some of the error in left elevon pitching moment to this term. By the third iteration, there is a noticeable amount of difference between the modified on-board model and truth and MSLS continues to produce estimates that indicate further decreases in the multiplicative factor are appropriate. The cumulative effect of

$K_1^0 K_1^1 K_1^2$ is presented in Figure 33 for the pitching moment due to pitch rate. The identification algorithms have estimated that the values of this term need to be approximately 85% of their nominal values.
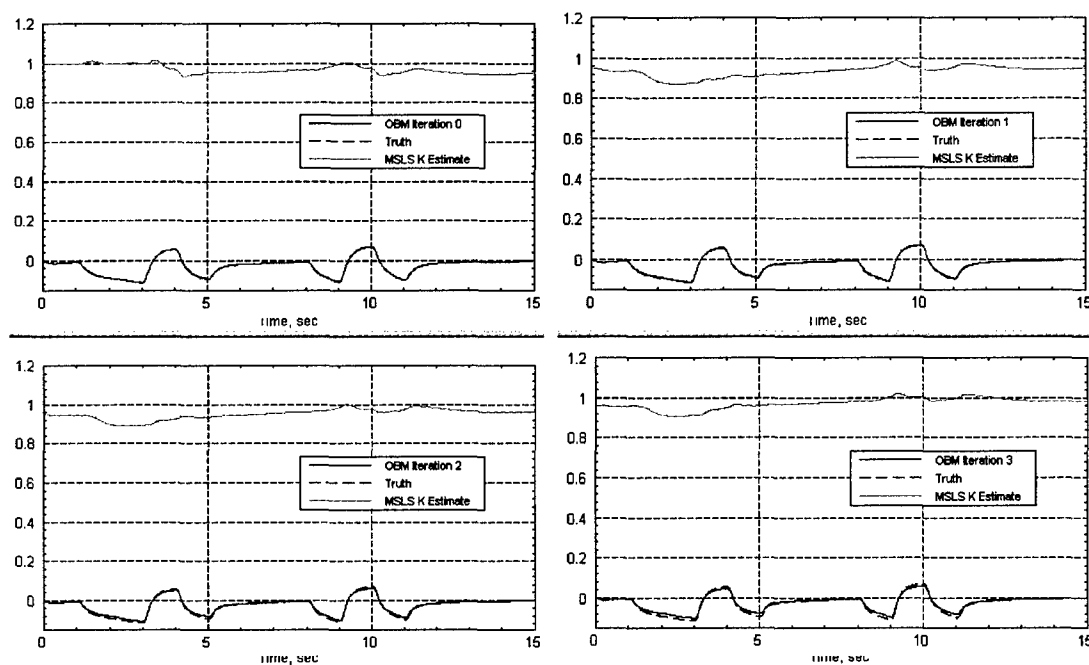
41

Figure 32 - Pitching Moment due to Pitch Rate and Associated Multiplicative Factor Estimate
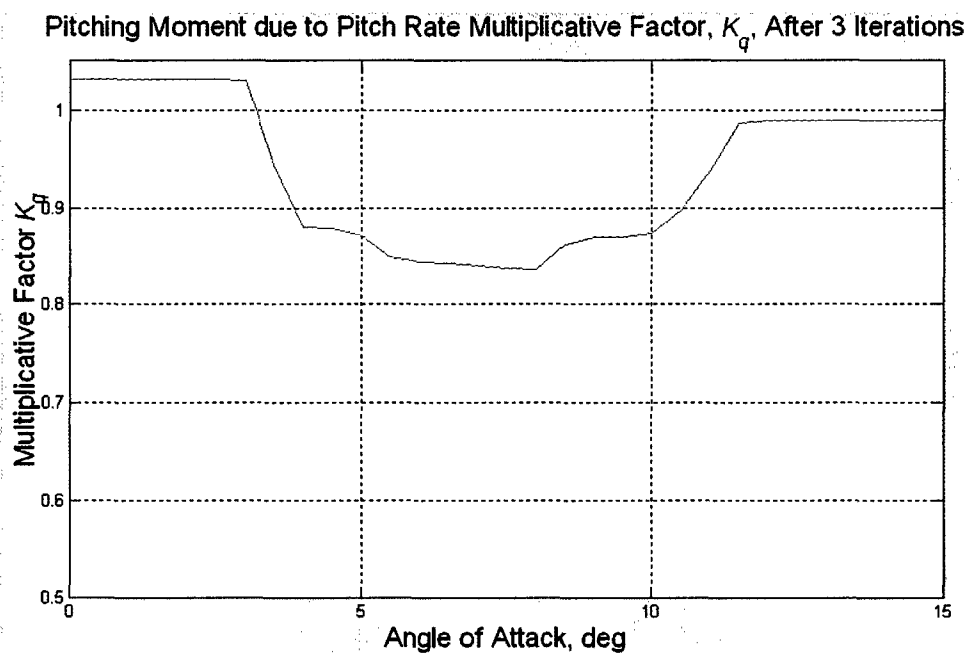

Figure 33 - Post-Processed Pitching Moment due to Pitch Rate Multiplicative Factor After Iteration 3.

The total aerodynamic pitching moment predicted by the on-board model compared to the true value is presented in Figure 34 for each of the four simulation runs. Examination of each graph demonstrates that the overall effect of the on-board model learning improves the total aerodynamic pitching moment predictions. The steady state error between the on-board model and truth from 11 seconds until the end of the simulation are not significantly affected through the learning process.
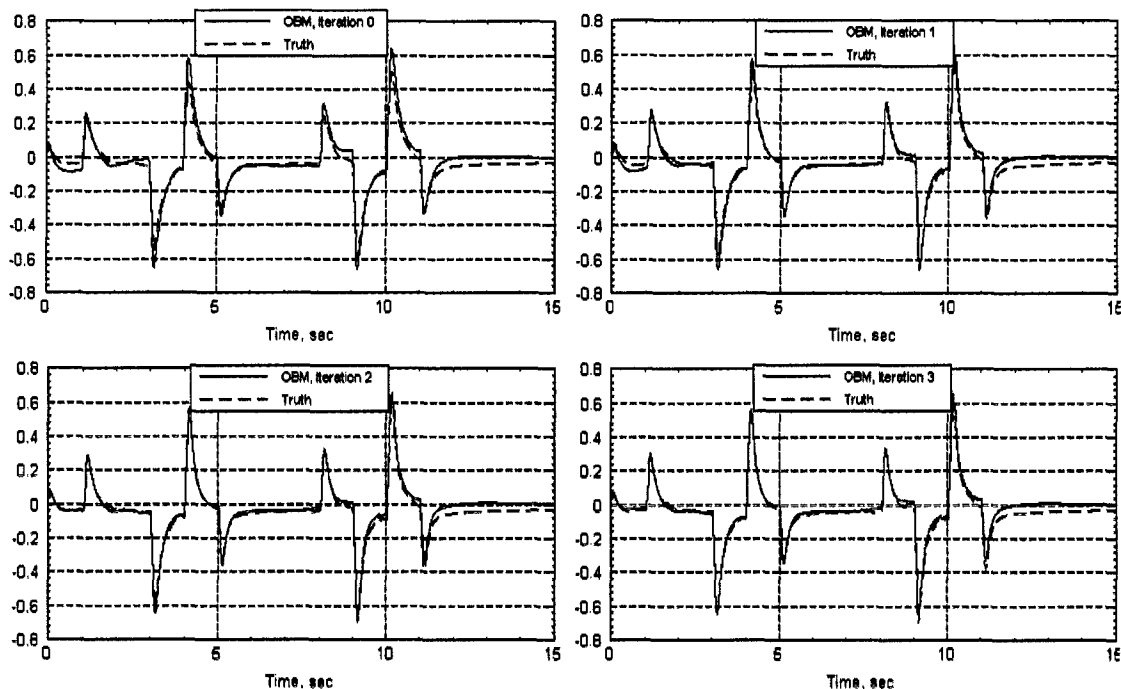

Figure 34 - Simulations Time Histories Comparing True and On-Board Models of Total Pitching Moment.

2.5.4.2. Simulations With Measurement Noise

As noted earlier, measurements of body axis pitch acceleration are used by MSLS to identify the multiplicative factors and associated bias. The approach taken by the Self-Designing Controller (SDC) [12] program VISTA/F-16 flight test used differences between two appropriate linear accelerometer measurements to estimate angular accelerations. An example plot from [12] indicated a noise level of approximately 2 deg/sec^2 on the estimate of body axis yaw acceleration. Assuming the pitch axis accelerometer arrangement would be similar, modifications to the Systembuild simulation were made that use a normal random number block with zero mean and 2 deg/sec^2 standard deviation. Figure 35 presents the body axis pitch acceleration as a function of time using the stick inputs described earlier with the added noise that is used by MSLS when performing the identification of the multiplicative parameters and the associated bias.
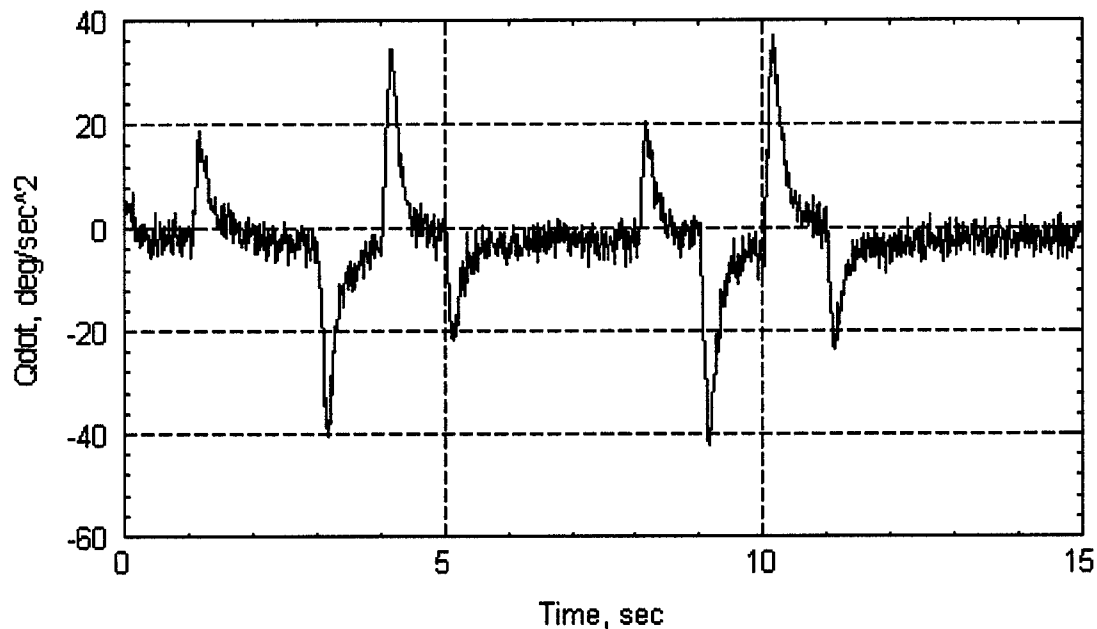
43

Figure 35 - Body Axis Pitch Acceleration With Gaussian Measurement Noise.

In a similar fashion to the previous section, Figure 36 presents a sequence of four plots, each one generated by the simulation after its respective iteration. Again, each plot presents the true value of the pitching moment increment as a dashed line and the on-board model representation as a solid line. The red line presents the multiplicative factor identified by MSLS. By the third iteration, the value of K_3

approached unity, indicating that the cumulative effect of $K_3^0 K_3^1 K_3^2$ has approximately reached a steady state value. Qualitatively, the results not vary significantly from those presented for the simulation without measurement noise.
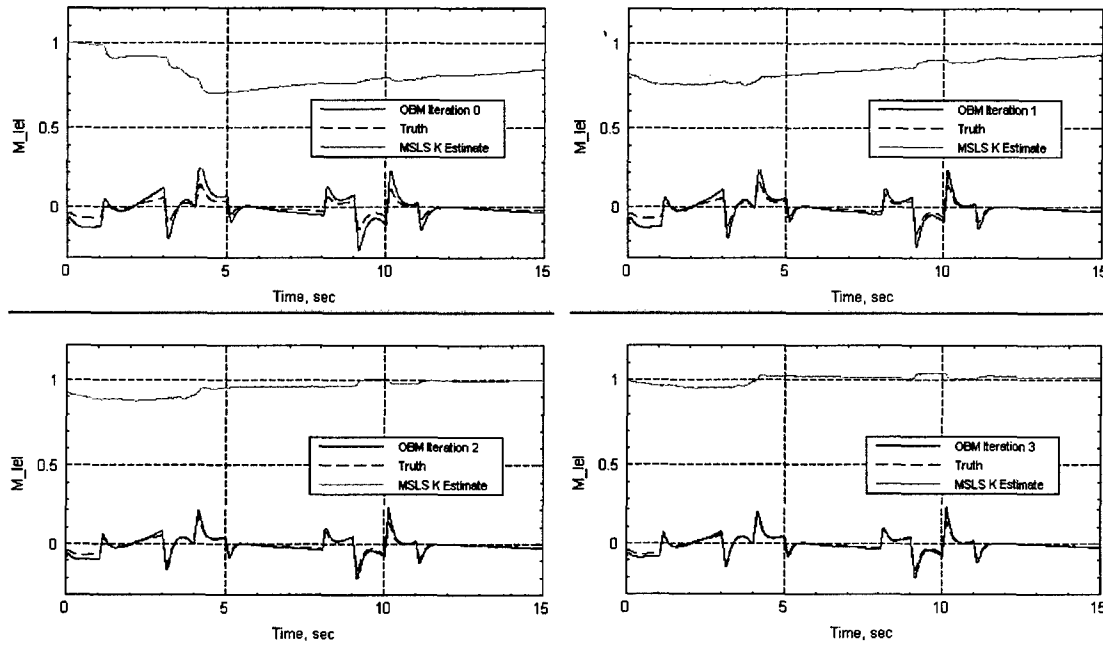
Figure 36 Pitching Moment due to Left Elevon and Associated Multiplicative Factor Estimate With Measurement Noise.

The next three graphics, Figure 37, Figure 38, and Figure 39, present the values of the multiplicative factors after iterations 0, 1 and 2, respectively. The multiplicative factor after iteration 0 is somewhat smaller for the case with measurement noise. After Iteration1, a small area near –1° left elevon deflection and 8° angle of attack has developed where the multiplicative factor is less than 0.5. The color scale for this plot is kept the same as the noiseless case for comparison purposes and therefore, regions less than 0.5 appear as white areas. By the end of Iteration 2, a second, smaller region near the same elevon deflection and 5° angle of attack has developed where the cumulative factor $K_3^0 K_3^1 K_3^2$ is less than 0.5. The minimum value of

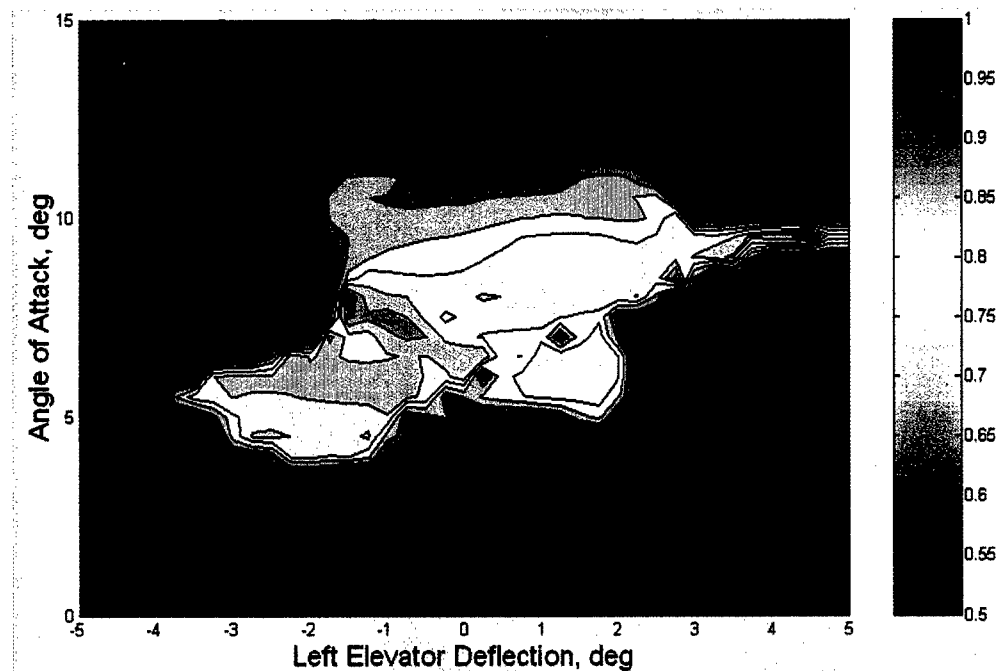$K_3^0 K_3^1 K_3^2$ is approximately 0.42 for the larger region.

Figure 37 - Post-Processed Pitching Moment due to Left Elevon Multiplicative Factor With Measurement Noise After Iteration 0.
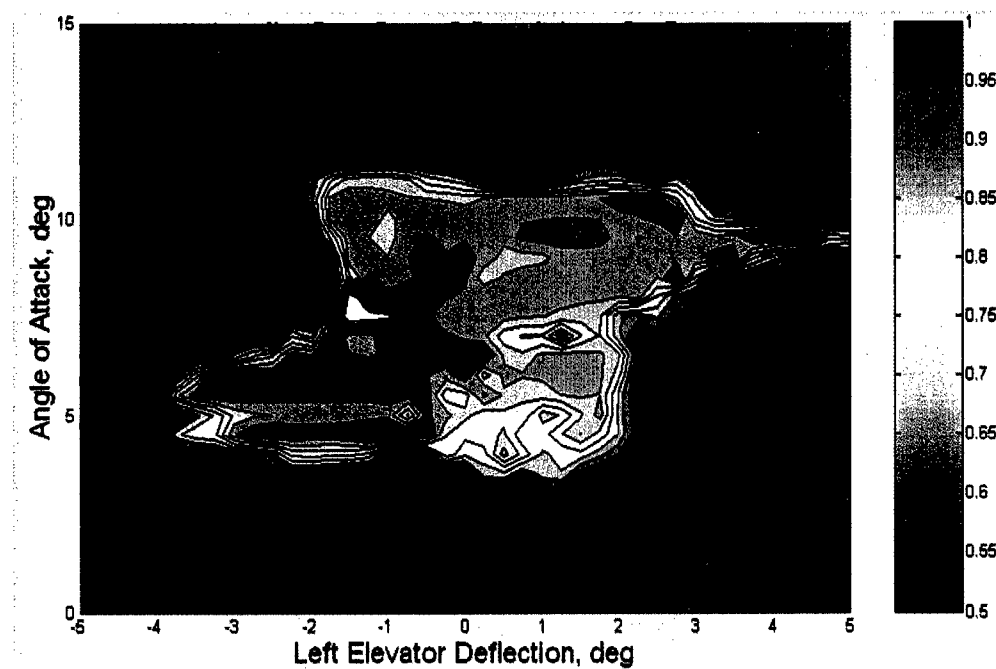

Figure 38 - Post-Processed Pitching Moment due to Left Elevon Multiplicative Factor With Measurement Noise After Iteration 1.
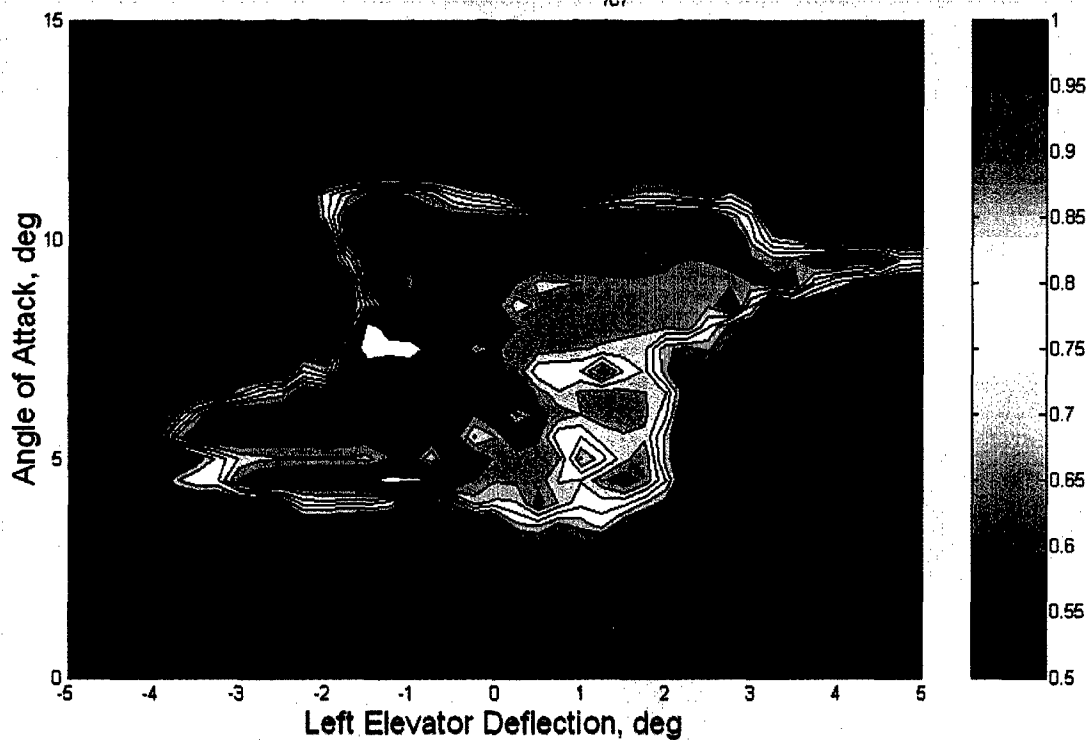
Figure 39 - Post-Processed Pitching Moment due to Left Elevon Multiplicative Factor With Measurement Noise After Iteration 2.

The right elevon multiplicative factor for each of the four simulations is shown in Figure 40. The results hear are similar to those in the noiseless measurement case. The value of the estimated factor approaches unity by the end of Iteration 2 and would be relatively unchanged if the results of Iteration 3 were considered. Figure 41 shows the contour plot of the cumulative multiplicative factor $K_4^0 K_4^1 K_4^2$ and it is also comparable to the noiseless measurement case.
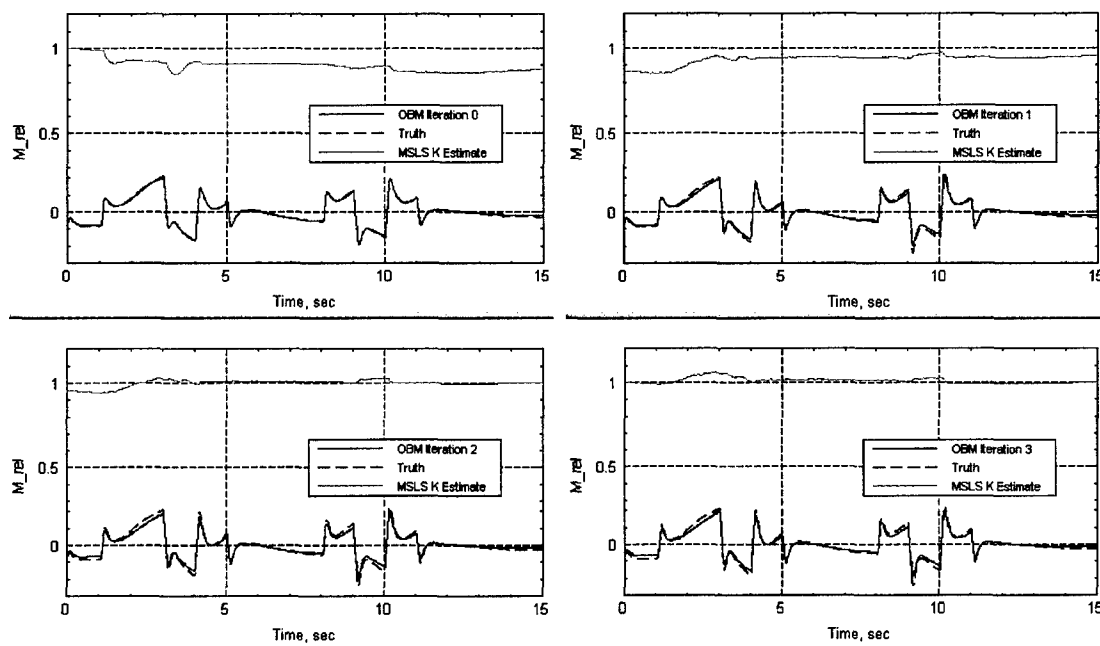
47

Figure 40 - Pitching Moment due to Right Elevon and Associated Multiplicative Factor Estimate With Measurement Noise.
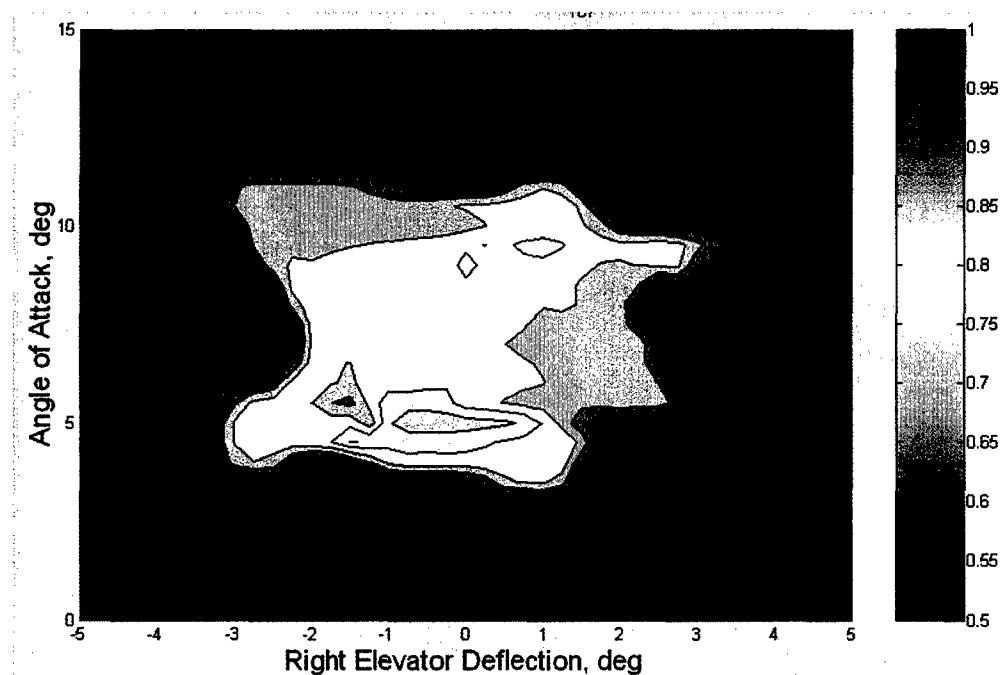

Figure 41 - Post-Processed Pitching Moment due to Right Elevon Multiplicative Factor With Measurement Noise After Iteration 2.

48

The pitch flap multiplicative factors and associated moments for the four simulations are shown in Figure 42. The cumulative effects of the three multiplicative factors $K_5^0 K_5^1 K_5^2$ are presented in Figure 43 as a contour diagram. Qualitative comparisons of these two graphs with the two without measurement noise show little difference between the individual multiplicative factor estimates or the cumulative $K_5^0 K_5^1 K_5^2$.
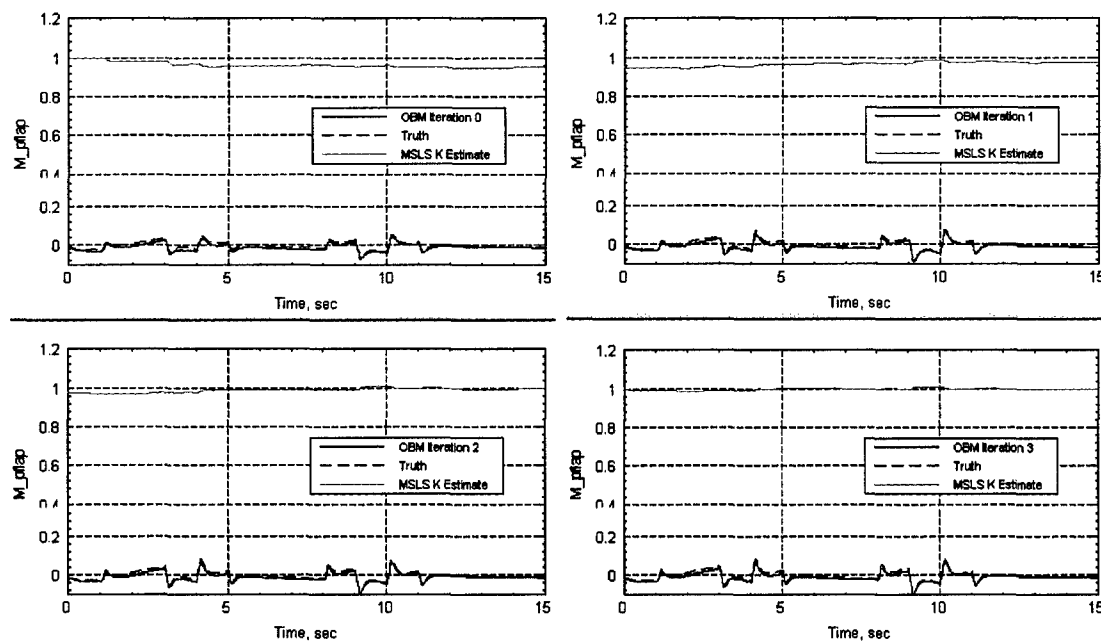

Figure 42 - Pitching Moment due to Pitch Flap and Associated Multiplicative Factor Estimate With Measurement Noise.

49
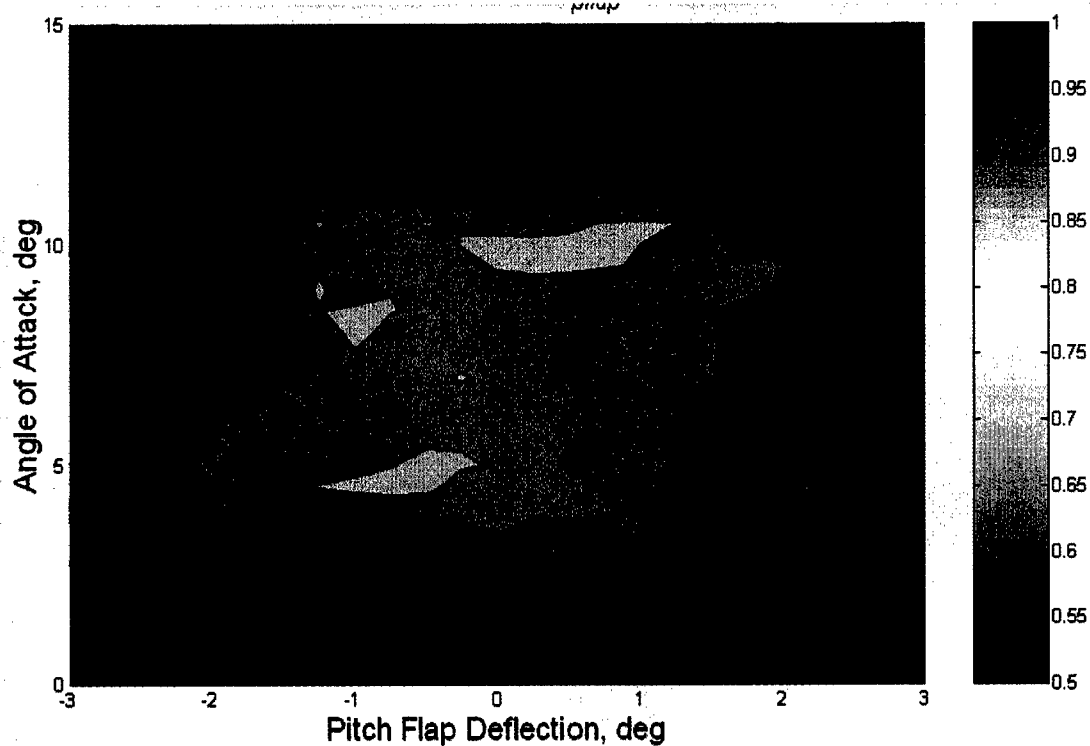
Figure 43 - Post-Processed Pitching Moment due to Pitch Flap Multiplicative Factor With Measurement Noise After Iteration 2.

The base pitching moment and associated multiplicative factors for each of the four simulations are shown in Figure 44 and the cumulative effect of $K_2^0 K_2^1 K_2^2$ is shown in Figure 45. Once again, the measurement noise has little effect on the estimates, when compared to the situation without measurement noise. The most notable exception is the reduction of $K_2^0 K_2^1 K_2^2$ at four degrees angle of attack. The result with measurement noise shows a slight decrease in $K_2^0 K_2^1 K_2^2$ across the range of angle of attack, compared to an increase at that angle of attack in the noiseless case.

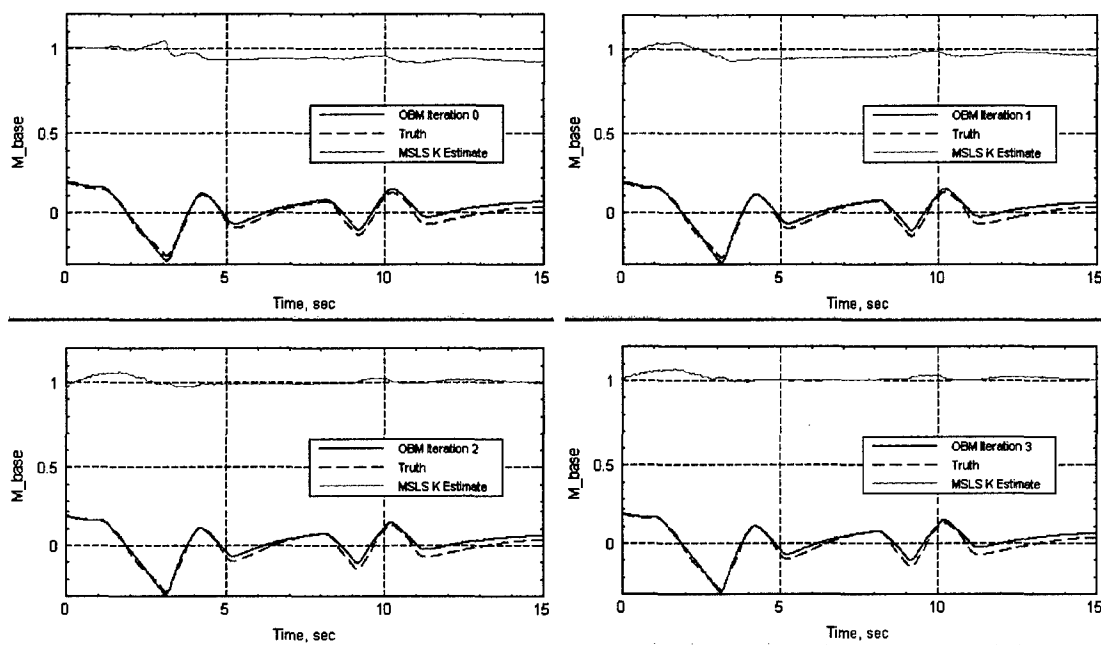**Figure 44 - Base Pitching Moment and Associated Multiplicative Factor Estimate With Measurement Noise.**
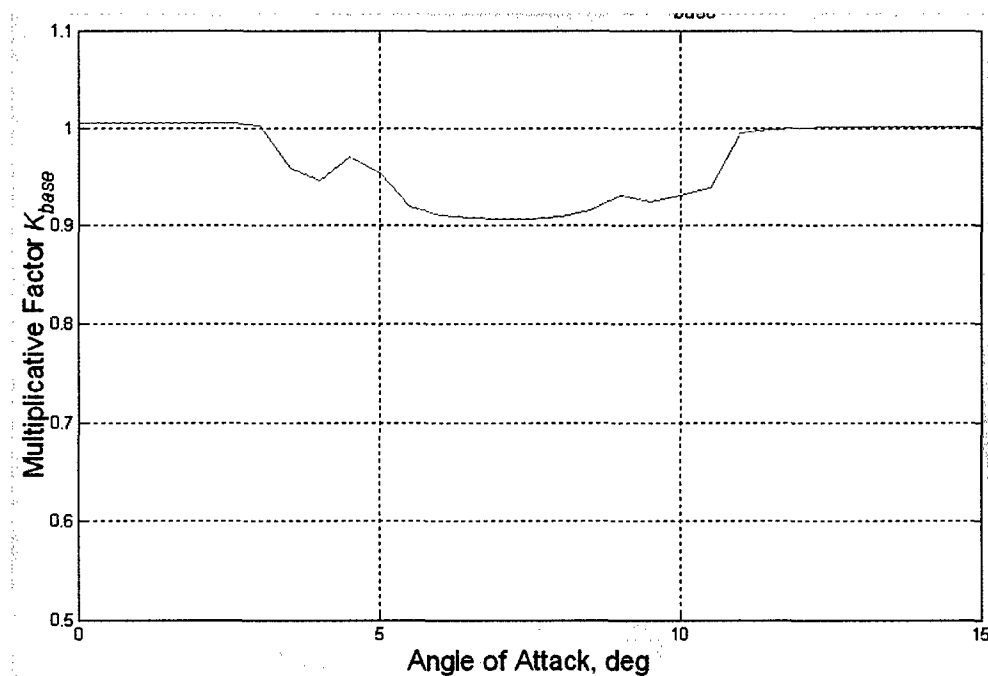
Figure 45 - Post-Processed Base Pitching Moment Factor With Measurement Noise After Iteration 2.

Figure 46 presents the four simulation time histories showing the pitching moment due to pitch rate and the associated multiplicative factor. The plot showing the cumulative $K_1^0 K_1^1 K_1^2$ is presented in Figure 47. As in all the previous cases, there is little difference between the cases with or without measurement noise. There are differences from 0° to 3° and 12° to 15° angle of attack that are produced by the neural network fitting process. At these ranges, the multiplicative factors should be unity. However, the neural network size is chosen not to fit exactly the training points but rather provide a smoother interpolation. As a consequence, values of one are not achieved.
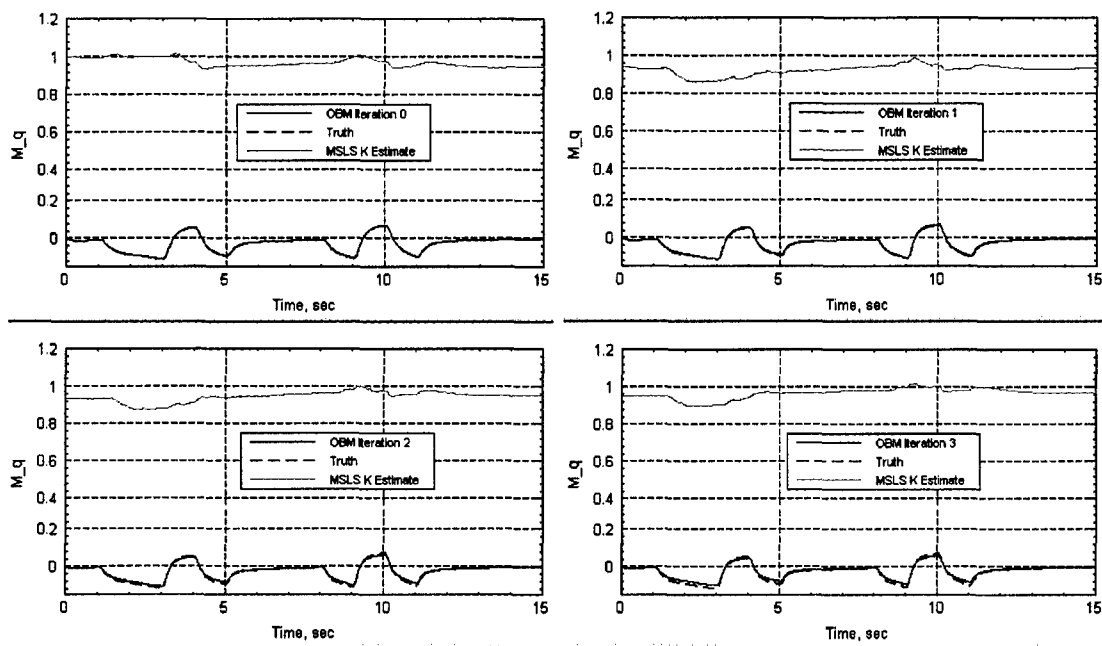
Figure 46 - Pitching Moment due to Pitch Rate and Associated Multiplicative Factor Estimate With Measurement Noise.
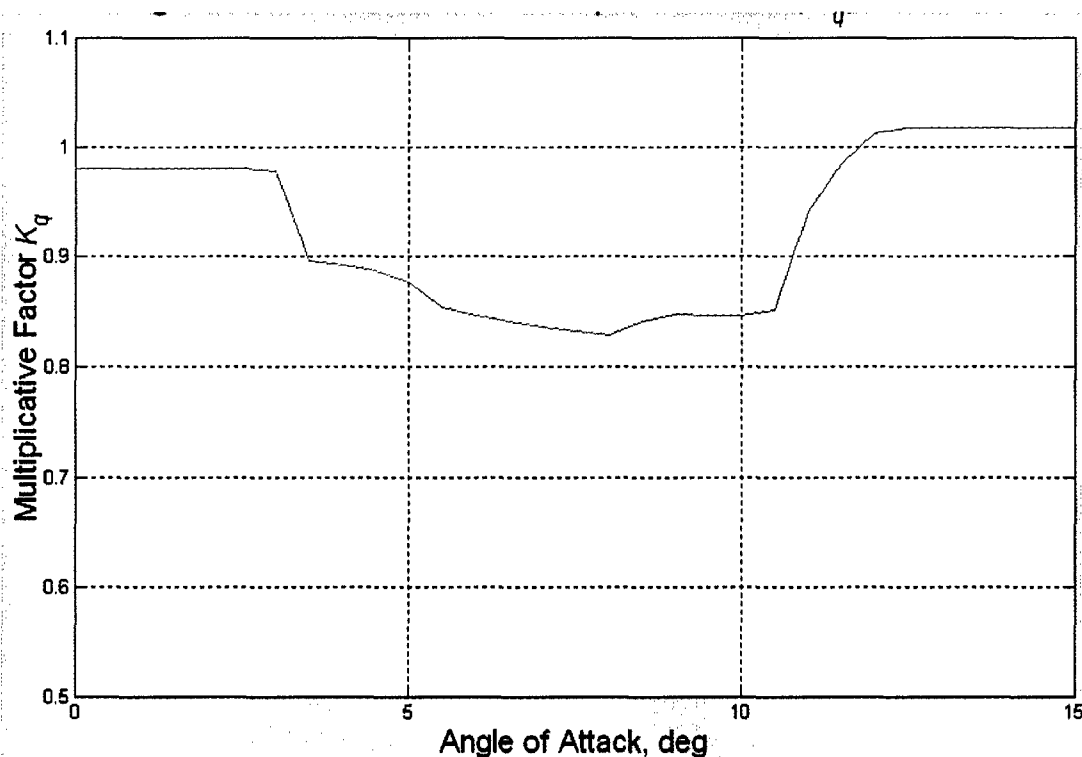
Figure 47 Post-Processed Pitching Moment due to Pitch Rate Multiplicative Factor With Measurement Noise After Iteration 2.

Finally, Figure 48 presents the total truth and on-board model pitching moment time histories for each of the four simulations. It comes as no surprise that there is little qualitative difference between these graphs and the ones presented earlier in Figure 34 since each of the components presented in this section displayed little difference between the simulations with and without measurement noise.

Figure 48 - Simulations Time Histories Comparing True and On-Board Models of Total Pitching Moment With Measurement Noise.

2.5.5. Discussion

The results presented above lend credence to the approach of using a multiplicative factor in the nonlinear aerodynamic buildup equation as a means of determining an on-board model increment. Several problems have also been brought to light as a result of this example, however. Their significance and possible remedies are discussed in this section.

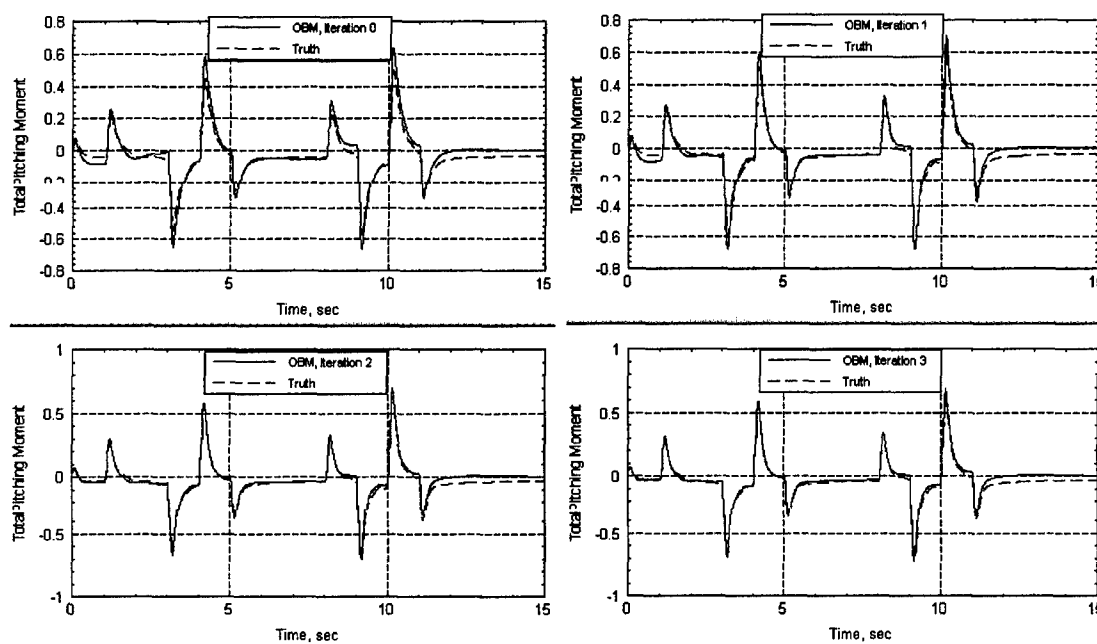Although the method did successfully identify the fifty percent loss of pitching moment effectiveness of the left elevon, MSLS erroneously spread the on-board model inaccuracy across the other terms in the build-up equation. Most notable, the term associated with the pitching moment due to pitch rate is identified as a major source of error by the parameter identification algorithms. Unfortunately, this term has an excellent nominal on-board model representation. Closer examination of this problem is certainly required and some remedies are under consideration.

Other techniques may be used to improve overall performance of the method. A threshold may be applied to each multiplier before neural network fitting. The threshold would restrict or filter the training data to consider only those points that varied from unity by more than a pre-specified amount. This step might lessen the tendency of the parameter identifications to spread or attribute effects across multiple terms in the buildup equations. Limits may also be placed on the maximum value a multiplicative factor could change. On a higher level, prognostics and health

management (PHM) methods could be used to identify effectors that are not operating within nominal ranges of movement, load, etc., and provide that information to the post-processing algorithms. Then only those terms in the aerodynamic buildup equation that are functions of the specified effector would be modified.

The consideration of measurement noise on the parameter identification shows little detrimental or beneficial effect. All of the factors in the build-up equation showed little difference between the simulations with or without measurement noise. The ability of MSLS to effectively filter out noise comes from the fact that the effective data window used for this work is set to approximately ten seconds. Also, the mathematical averaging that occurs when post processing the simulation data helped alleviate the effects of noise.

The use of the identified bias term may also be key to effective use of the parameter identification data. Situations where the control effectors are near or at their neutral positions may present problems for the PID algorithms since the associated effector increments are, by their definition, typically zero for those effector positions. A portion of the non-zero bias may be added to the base aerodynamic term or to the effector increment in an attempt to account for the non-zero bias. Fortunately, no numerical problems have been found thus far concerning the identification of multiplicative factors for terms that are zero for short periods of time.

2.5.6. Conclusion

The method presented herein demonstrates the ability to generate on-board model increments that improve the overall accuracy of the on-board model. The method is demonstrated during simulation of an aircraft with significant nonlinear aerodynamics and for cases with and without measurement noise.

Additional features such as applying thresholds and limits to the MSLS estimates before learning may lead to improved model increments. Similar improvements may be expected through the use of the identified bias. Future work will focus on the inclusion of the normal force, side force, rolling moment, and yawing moment equations. A re-examination of uses for the identified bias term will be examined as well.

3. TRANSITIONS

3.1. LM Aero Programs

This project resulted in four primary transitions to LM Aero Programs. The on-board modeling technology was successfully transitioned to the Joint Strike Fighter Program and the AIMSAFE contracted research and development project. The intelligent on-board modeling technology was also transitioned to the Joint Strike Fighter Program. The dynamic control allocation technology was transitioned to the Joint Strike Fighter Program as well. More detailed descriptions of the transistions follow.

Transition 1

Technology: Neural Network Compact On-board Models

Customer: JSF Program, Lockheed Martin Aeronautics Company

Result: Compact neural network modeling. Fundamental neural network theory was developed for compact analytical modeling of dynamical systems.

Application: The basic neural network modeling theory was applied to aircraft dynamical models and integrated into a tool with a graphical user interface under additional LM Aero IR&D funding. This tool was used to generate compact models of the JSF aerodynamics. These compact models are being evaluated for use as control law on-board models for JSF.

Transition 2

Technology: Neural Network Compact On-board Models

Customer: NASA AIMSAFE 6.2 R&D Program, Lockheed Martin Aeronautics Company

Result: Compact neural network modeling. Fundamental neural network theory was developed for compact analytical modeling of dynamical systems.

Application: The basic neural network modeling theory was applied to aircraft dynamics models and integrated into a tool with a graphical user interface under additional LM Aero IR&D funding. This tool was used to generate compact models of the aerodynamics and propulsion systems for a business jet. These compact models are being used as on-board models for the business jet control laws in the NASA AIMSAFE 6.2 R&D program.

Transition 3

Technology: Intelligent On-board Models

Customer: JSF Program, Lockheed Martin Aeronautics Company

Result: Fundamental neural network theory and parameter identification theory were integrated to provide an intelligent on-board model update capability for flight test and/or reconfiguration. The structure of the on-board model updates is consistent with nonlinear aerodynamic model architectures.

Application: The intelligent on-board models are being investigated for JSF on-board model design.

Transition 4

Technology: Dynamic Control Allocation

Customer: JSF Program, Lockheed Martin Aeronautics Company

Result: Preliminary development and analysis of dynamic control allocation.

Application: The dynamic control allocation technology is being investigated for integration into the JSF effector blender during the the EMD phase of the program.

3.2. Publications

The following publications are directly related to results from this effort.

- M. A. Niestroy and J. M. Buffington, "Intelligent On-board Models", _planned submittal to the 2002 AIAA Guidance, Navigation, and Control Conference._
- C. M. Ha and J. M. Buffington, "Dynamic Control Allocation", _planned submittal to the 2002 AIAA Guidance, Navigation, and Control Conference._

The following publications report on results of related projects, during the period of performance of this effort, that have indirectly contributed to the results and are authored by contributors of this program.

- V. Kapila, A. G. Sparks, J. M. Buffington, and Q. Yan, "Spacecraft Formation Flying: Dynamics and Control", _Journal of Guidance, Control, and Dynamics,_ Vol. 23, No. 3, 2000, pp.561-563.
- Y. Shtessel, J. Buffington, and S. Banda, "Multiple Timescale Flight Control Using Reconfigurable Sliding Modes", _Journal of Guidance, Control, and Dynamics,_ Vol. 22, No. 5, 1999.
- J. Buffington, P. Chandler, and M. Pachter, "Interaction of System Identification and Control Allocation in Reconfigurable Control Systems", _International Journal of Robust & Nonlinear Control,_ Vol. 9, No. 14, 1999.
- R. L. Eberhardt, D. G. Ward, "Indirect Adaptive Flight Control System Interactions", to appear in the _International Journal of Robust and Nonlinear Control._

- R. L. Eberhardt, D. G. Ward, "Indirect Adaptive Flight Control System Interactions", AIAA Guidance, Navigation, and Control Conference, August 1999.
- R. Lind , J. M. Buffington, and A. G. Sparks "Multi-loop Aeroservoelastic Control of a Hypersonic Vehicle," *1999 AIAA GN&C Conference*, Portland OR, Aug. 1999, AIAA 99-4123.
- D. S. Naidu, , J. M. Buffington, and S. S. Banda "Optimal Control of Singularly Perturbed Systems with Inequality Constraints," *1999 AIAA GN&C Conference*, Portland OR, Aug. 1999, AIAA 99-4125.
- D. S. Naidu, , J. M. Buffington, and S. S. Banda "Further results on nondimensional forms for singularly perturbed structures," *1999 AIAA GN&C Conference*, Portland OR, Aug. 1999, AIAA 99-3981.
- D. S. Naidu, , J. M. Buffington, and S. S. Banda "Resurrection in Hypersonics: Why, What and When," *1999 AIAA GN&C Conference*, Portland OR, Aug. 1999, AIAA 99-4053.
- Y. Shtessel, J. Buffington, and S. Banda, "Tailless Aircraft Flight Control Using Multiple Time Scale Reconfigurable Sliding Modes," *1999 AIAA GN&C Conference*, Portland OR, Aug. 1999, AIAA 99-4136.
- D. S. Naidu, S. S. Banda, and J. M. Buffington, "Unified Approach to H_2 and H_∞ Optimal Control of Hypersonic Vehicles," *Proceedings of the 1999 American Control Conference*, San Diego CA, Jun. 1999.
- V. Kapila, A. Sparks, J. Buffington and Q. Yan, "Spacecraft Formation Flying: Dynamics and Control," *Proceedings of the 1999 American Control Conference*, San Diego CA, Jun. 1999.

4. REFERENCES

[1] Balas, G. J., <u>Robust Multivariable Control Short Course - Theory and Applications Using Mu-Tools</u>, Musyn, Inc., P.O. Box 13377, Minneapolis, Minnesota 55414-5377, August 4-7, 1992.

[2] Buffington, J. M., Sparks, A. G., and Leggett, D. B., "Synthesis And Analysis Of Dynamic Inversion And Linear Parameter Varying Tailless Flight Control Law Designs," AIAA-98-4248.

[3] Buffington, J. M., "Modular Control Law Design for the Innovative Control Effectors (ICE) Tailless Fighter Aircraft Configuration 101-3," AFRL-VA-WP-TR-1999-3057.

[4] Doman, D. B., Ngo, A. D., Leggett, D. B., Saliers, M. A., and Pachter, M., "Development of a Hybrid Direct/Indirect Adaptive Control System for the X-33," AIAA-2000-4156.

[5] Doyle, J. C., <u>Matrix Interpolation Theory and Optimal Control</u>, Ph.D. Dissertation, University of California, Berkeley, California, 1984.

[6] Garcia-Velo, J. and Walker, B.K., "Aerodynamic Parameter Estimation for High Performance Aircraft Using Extended Kalman Filter," J. Guidance, Control and Dynamics, vol. 20, no. 6, pp. 1257-1260, Sept.-Oct. 1997.

[7] Morelli, E.A., *System IDentification Programs for AirCraft (SIDPAC)*, NASA Langley Research Center, Hampton, VA, 2001.

[8] Page, A. B. and Steinberg, M. L., "A Closed-Loop Comparison of Control Allocation Methods," AIAA-2000-4538.

[9] Tallant, et al, "Reconfigurable Systems for Tailless Fighter Aircraft – RESTORE", AFRL-VA-WP-TR-1999-3078, September, 1999.

[10] Tischler, M.B., "Comprehensive Identification from Frequency Responses", NASA Ames Research Center, Moffett Field, CA, 2001.

[11] Venkataraman, R. and Doman, D. B., "Control Allocation and Compensation for Over-Actuated Systems with Non-Linear Effectors," ACC01-SHT1-72.

[12] Ward, D.G., et al, "Self-Designing Controller: Design, Simulation and Flight Test Evaluation", WL-TR-97-3095, November, 1996.

[13] Xmath Version 6.1.3, Copyright @ 1999, Integrated Systems, Inc.

DEPARTMENT OF THE AIR FORCE

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH

801 N. Randolph St., Arlington, Va. 22203-1977

30 May 2002

DTIC
8725 Kingman Rd
Suite 944
Ft. Belvoir, VA 22060-6218
ATTN: Larry Downing

Re: ADB 269236

Dear Mr. Downing:

Please change the distribution statement of the above referenced AD number to reflect a Distribution A - *Unlimited Distribution.*

Feel free to call me at (703) 696-9518 with any further questions.

Sincerely,

Yvonne M. Mason

The Basic Research Manager of the Air Force Research Laboratory