

2008

Pilot in loop assessment of fault tolerant flight control schemes in a motion flight simulator

Girish Kumar Sagoo
West Virginia University

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Sagoo, Girish Kumar, "Pilot in loop assessment of fault tolerant flight control schemes in a motion flight simulator" (2008). *Graduate Theses, Dissertations, and Problem Reports*. 1918.
<https://researchrepository.wvu.edu/etd/1918>

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

**Pilot in Loop Assessment of Fault Tolerant Flight Control Schemes in a Motion
Flight Simulator**

Girish Kumar Sagoo

Thesis submitted to the
College of Engineering and Mineral Resources
at West Virginia University in
partial fulfillment of the requirements
for the degree of

Master of Science
in
Aerospace Engineering

Dr. Marcello R. Napolitano, Ph.D., Chair
Dr. Mario G. Perhinschi, Ph.D.
Dr. Bojan Cukic, Ph.D.

Department of Mechanical and Aerospace Engineering



Morgantown, West Virginia

2008

Keywords: Flight Simulator, Failure Detection and Identification, Pilot Assessment

ABSTRACT

Pilot in Loop Assessment of Fault Tolerant Flight Control Schemes in a Motion Flight Simulator

Girish Kumar Sagoo

This research presents the pilot in the loop tests carried out in a Six-Degree of Freedom (6-DOF) motion flight simulator to evaluate failure detection, isolation and identification (FDII) schemes for an advanced F-15 aircraft. The objective behind this study is to leverage the capability of the flight simulator at West Virginia University (WVU) to carry out a performance assessment of neurally augmented control algorithms developed on a Matlab/Simulink® platform. The experimental setup features an interface setup of *Gen-2* Simulink® schemes with MOTUS Flight Simulator (MFS). The set up is a close substitute to a real flight and thus is helpful in evaluation of the schemes in a realistic manner. The graphics in X-plane is used to obtain visual cues and the motion platform is used to obtain motion cues in the simulator cockpit. The whole set-up enables the pilot to respond with a joystick in the advent of a failure as he would otherwise in a real flight. The pilot response in maintaining the mission profile is different for different neural network augmentations and thus an indication of performance comparison of these schemes. Secondly, FDII schemes are developed for a sensor and actuator failure using an adaptive threshold for cross-correlation coefficients of the angular rates of the aircraft. Failure detection, isolation and identification logic is formulated based on monitoring the cross-correlation parameters with their Floating Limiter (FL) bounds. The FDII scheme developed shows a good performance with desktop simulation because of no pilot activity but with a pilot in the loop significant cross-correlation of the rates occur and hence the scheme become more susceptible to wrongs FDII. In addition, the pilot might induce some coupling of the cross-correlation parameters between detection and identification time which may trigger false detections and may configure the controller differently based on incorrect detection. Thus it is necessary that FDII scheme accommodate real flight conditions. The performance of the FDII schemes is improved with a pilot in the loop by monitoring the cross-correlation parameters and fine tuning FDII algorithms for real situations. This study has set up an excellent example to effectively utilize the aural, visual and motion cues to create a higher level of simulation complexity in designing control algorithms.

Acknowledgements

I would like to express my deep and sincere gratitude to my supervisor, Dr. Marcello R. Napolitano for his immense guidance throughout my degree. My research is the outcome his foresight and his guidance was indispensable to carry out this work in a proper manner. He was an exceptional mentor and had always offered his help in matters related to academic as well as personal. Words cannot possibly suffice my sincere thanks to him.

I am also grateful to my committee members Dr. Mario G. Perhinschi and Dr. Bojan Cukic to evaluate my work critically. Dr. Perhinschi has always been at help without reservation and he helped me out to understand the *Gen-2* architecture.

I am also thankful to Dr. Srikanth Gururajan, Dr. Yu Gu, Dr. Brad Seanor and Dr. Giampiero Campa for monitoring the progress of my research and providing constructive comments throughout this work.

I would also like to extend my gratitude to Dr. Jacky C. Prucz, Chairman Graduate Program at MAE for his immense help in many critical academic situations during my degree. His support is deeply appreciated.

During this work I have collaborated with my colleagues for whom I have a great regard and who had made my stay at WVU memorable and pleasurable. I would like to thank Marco, Josh, Jason Jarrel, Kerry, Hever, Sebastian and Sergio to make a productive working environment. My sincere thanks to Jason and Kerry in volunteering for carrying out flight tests in the first phase. My sincere thanks to Bob Campione, a licensed pilot and working at WVU for helping me out in carrying out the second phase of the piloted experiments. His feedback had helped me a lot in fine tuning the schemes.

I also owe my thanks to my wife who was always with me in thick and thin and provided a constant support and encouragement in carrying out this research work. My parents have been a constant source of moral and mental support to me and I am deeply indebted for their blessings.

Last but not least the financial support provided by West Virginia University and NASA is gratefully acknowledged.

Table of Contents

Title Page.....	i
ABSTRACT.....	ii
Acknowledgements	iii
Table of Contents	iv
List of Tables	vi
List of Figures	ix
Nomenclature.....	xiii
Chapter 1. Introduction.....	1
1.1. Importance of Failure Detection and Identification.....	1
1.2. Overview of Flight Simulation	3
1.3. Research Objectives.....	7
1.4. Overview of Thesis	9
Chapter 2. Literature Review.....	10
2.1. General Description	10
2.2. Fault Detection Methods	11
2.2.1. Model Based Methods.....	12
2.2.2. Knowledge Based Methods	14
Chapter 3. Theoretical Background	17
3.1. Description of Gen-2 Scheme	17
3.1.1. NLDI scheme	18
3.1.2. Neural Network Algorithms	19
3.2. Aerodynamic Modeling of the failure.....	22
3.2.1. Failure Detection and Identification concepts	22
3.2.2. The Adaptive Threshold Concept for FDII	22
3.2.3. The FL architecture for Integrated Sensor/Actuator Failure	24
3.2.4. The Adaptive Threshold Parameters for S/AFDII Scheme	25
3.2.5. The detection phase.....	27
3.2.6. The Isolation phase.....	27
3.2.7. Failure Identification for Actuator Isolated	28
3.2.8. Failure Identification for Sensor Isolated	28
3.3. Integrated Sensor/Actuator S/AFDII Scheme with FL.....	30
Chapter 4. Experimental Procedures.....	34
4.1. Experimental Environment Setup.....	34
4.1.1. Description of MOTUS Flight Simulator.....	35
4.1.2. Description of X-plane Software	36
4.2. Interface of Simulink Scheme with Motion Simulator.....	37
4.2.1. Motion Computer Data and Displays.....	39
4.2.2. UDP interface and IP address settings for Communications.....	41
4.2.3. Pilot interaction and Warning Systems in the Cockpit	44
4.3. Description of Simulink software.....	46
4.3.1. The Simulink graphic user interface	47
Chapter 5. Results and Evaluation	50
5.1. Overview of the experiments.....	50
5.2. Comparison of different NN schemes with Actuator failure (NO FDII).....	50
5.2.1. Tracking Error Performance for Nominal Condition (Baseline Doublets).....	51

5.2.2.	Analysis of Actuator Failures (No pilot- No FDII).....	53
5.2.3.	Analysis of Actuator Failures (With Pilot- No FDII)	54
5.3.	Evaluation of Integrated FDII Scheme with adaptive FL threshold	60
5.3.1.	Analysis of Actuator Failure Cases (Stabilator Failure)	62
5.3.1.1.	Tracking Error Analysis of Stabilator Failures	65
5.3.1.2.	Pilot Activity Analysis.....	68
5.3.1.3.	FDII Analysis of Stabilator Failures.....	73
5.3.2.	Analysis of Actuator Failure Cases (Aileron Failure).....	78
5.3.2.1.	Tracking Error Analysis.....	78
5.3.2.2.	Stick Activity Analysis	81
5.3.2.3.	FDII Analysis of Aileron Failures.....	85
5.3.3.	Analysis of Sensor Failure Cases (Roll Sensor)	88
5.3.3.1.	Tracking Error Analysis.....	88
5.3.3.2.	Stick Activity Roll Sensor Failures	89
5.3.3.3.	FDII Analysis of Roll Sensor Failures	90
5.3.4.	Analysis of Sensor Failure Cases (Pitch Sensor).....	97
5.3.4.1.	Tracking Error Analysis.....	97
5.3.4.2.	Stick Activity Analysis	98
5.3.4.3.	FDII Analysis of pitch sensor failures	99
5.3.5.	Analysis of Sensor Failure Cases (Yaw Sensor).....	106
5.3.5.1.	Tracking Error Analysis.....	106
5.3.5.2.	Stick Activity Analysis	107
5.3.5.3.	FDII Analysis of Yaw Sensor Failures	108
Chapter 6.	Conclusion and Recommendations	112
6.1.	Conclusion.....	112
6.2.	Recommendations.....	114
References	116

List of Tables

Table 3-1 Floating Limiter Tuning Parameters for Integrated FDII Scheme	30
Table 4-1 : Data Sent to X-Plane for Visual Cue.....	39
Table 4-2- Data Sent to Motion base for Motion Cue.....	39
Table 4-3- Motion Base Computer Data Format for Receiving via UDP	40
Table 4-4- Limits on acceleration for Motion base.....	40
Table 5-1 Longitudinal Doublet TE – (Nominal Condition).....	51
Table 5-2 Latéral Doublet TE Statistics (Nominal Condition).....	52
Table 5-3 -Stabilator Failure TE Statistics (No Pilot).....	53
Table 5-4 Aileron Failure TE Statistics (No Pilot)	53
Table 5-5 Stabilator Failure TE Statistics (Pilot 1).....	54
Table 5-6 Aileron Failure TE Statistics (Pilot 1)	55
Table 5-7 Stabilator Failure TE Statistics (Pilot 2).....	56
Table 5-8 Aileron Failure TE Statistics (Pilot 2)	57
Table 5-9 Stabilator Failure TE Statistics (Pilot 3).....	58
Table 5-10- Aileron Failure TE Statistics (Pilot 3).....	59
Table 5-11 Comparison done for finalizing NN for further studies.....	61
Table 5-12- Actuator Failure FDII results with Adaptive Threshold (Run1).....	63
Table 5-13- Actuator Failure FDII results with Adaptive Threshold (Run2).....	63
Table 5-14- Actuator Failure FDII results with Adaptive Threshold (Run3).....	63
Table 5-15- Actuator Failure FDII results with Adaptive Threshold (Run4).....	64
Table 5-16- Actuator Failure FDII results with Adaptive Threshold (Run5).....	64
Table 5-17-Summary of Piloted Tests of Adaptive FDII scheme for Actuator Failure....	64
Table 5-18- TE statistics for Left Stabilator (Run1)	65
Table 5-19- TE statistics for Left Stabilator (Run2)	65
Table 5-20- TE statistics for Left Stabilator (Run3)	65
Table 5-21- TE statistics for Left Stabilator (Run4)	65
Table 5-22- TE statistics for Left Stabilator (Run5)	66
Table 5-23- Average of 5 runs for Stabilator failure.....	68
Table 5-24- Pilot Activity for Left Stabilator Failure (Run 1)	69

Table 5-25- Pilot Activity for Left Stabilator Failure (Run 2)	70
Table 5-26- Pilot Activity for Left Stabilator Failure (Run 3)	70
Table 5-27- Pilot Activity for Left Stabilator Failure (Run 4)	70
Table 5-28- Pilot Activity for Left Stabilator Failure (Run 5)	70
Table 5-29 Average Pilot Activity for Stabilator Failure	71
Table 5-30 Composite Parameter for Stabilator Failure.....	72
Table 5-31 Average CP for Stabilator Failure Cases	73
Table 5-32-TE statistics for aileron failure (Run 1).....	79
Table 5-33-TE statistics for aileron failure (Run2).....	79
Table 5-34-TE statistics for aileron failure (Run3).....	80
Table 5-35-TE statistics for aileron failure (Run4).....	80
Table 5-36-TE statistics for aileron failure (Run5).....	80
Table 5-37-Average of 5 Runs for Aileron Failure.....	80
Table 5-38- Pilot Activity for Right Aileron Failure (Run 1).....	82
Table 5-39- Pilot Activity for Right Aileron Failure (Run 2).....	82
Table 5-40- Pilot Activity for Right Aileron Failure (Run 3).....	82
Table 5-41- Pilot Activity for Right Aileron Failure (Run 4).....	82
Table 5-42- Pilot Activity for Right Aileron Failure (Run 5).....	83
Table 5-43 Average Pilot Activity for Aileron Failure Cases	83
Table 5-44 Composite Parameter for Aileron Failures	84
Table 5-45 Average Composite Parameter for Aileron Failures	84
Table 5-46 Tracking Error (TE) statistics for Roll Sensor Failures.....	89
Table 5-47 Stick Activity for Roll Sensor Failures.....	90
Table 5-48 Average Stick Activity for Roll Sensor Failures.....	90
Table 5-49 FDII results for Roll Sensor Failure (Run1)	91
Table 5-50 FDII results for Roll Sensor Failure (Run2)	91
Table 5-51 FDII results for Roll Sensor Failure (Run3)	91
Table 5-52 FDII results for Roll Sensor Failure (Run4)	91
Table 5-53 FDII results for Roll Sensor Failure (Run5)	91
Table 5-54 Tracking Error (TE) statistics of Pitch Sensor Failures.....	98
Table 5-55 Stick Activity for Pitch Sensor Failures	99

Table 5-56 Average Stick Activity for Pitch Sensor Failures	99
Table 5-57 FDII results for Pitch Sensor Failure (Run1)	100
Table 5-58 FDII results for Pitch Sensor Failure (Run2)	100
Table 5-59 FDII results for Pitch Sensor Failure (Run3)	100
Table 5-60 FDII results for Pitch Sensor Failure (Run4)	100
Table 5-61 FDII results for Pitch Sensor Failure (Run5)	100
Table 5-62 TE statistics for Yaw Sensor Failures.....	106
Table 5-63 Stick Activity for Yaw Sensor Failures	107
Table 5-64 Average Stick Activity for Yaw Sensor Failures	108
Table 5-65 FDII results for Yaw Sensor Failure (Run1).....	108
Table 5-66 FDII results for Yaw Sensor Failure (Run2).....	108
Table 5-67 FDII results for Yaw Sensor Failure (Run3).....	108
Table 5-68 FDII results for Yaw Sensor Failure (Run4).....	109
Table 5-69 FDII results for Yaw Sensor Failure (Run5).....	109
Table 5-70-Summary of Piloted Tests of Sensor Failures with Adaptive FDII scheme .	111

List of Figures

Fig. 1-1-Boeing Rudder Failure Accidents (Courtesy Seattle Times)	2
Fig. 1-2-A locked slat on a Boeing 737 -844 (courtesy www.airliners.net).....	3
Figure 1-3 A comparison of visual cue improvement (1982 to 2004).....	4
Figure 1-4- LAMARS Motion Flight Simulator at Wright Patterson Air Force Base, Ohio	5
Figure 1-5-Visual Cues in a home-built Boeing Style Simulator	6
Figure 1-6 Six Electromechanical Actuators for MOTUS Flight Simulator at WVU	7
Fig. 3-1-General Block Diagram for IFCS Gen-2 Controller Scheme [46]	17
Fig. 3-2- The integration of NLDI based F-15 Controller with MFS	21
Figure 3-3- A FL approach to FDII showing the detection with MQEE	24
Figure 3-4- Block Diagram of Floating Limiter	25
Figure 3-5- Simulink block showing Bounds of Floating Limiter.....	25
Figure 3-6 Logical Diagram for FDII.....	29
Fig. 3-7-Integrated Simulink scheme for S/AFDIA with Floating Limiter and MFS Interface	31
Fig. 3-8-S/AFDIA Scheme incorporating the Floating Limiters	31
Figure 3-9 A warning window with Failure Detection	32
Figure 3-10 GUI for Actuator Failure Identification	32
Figure 3-11 Typical Warning received at lower left corner of pilot facing visual display	33
Fig. 4-1- Joystick Used for Piloted Experiments	34
Fig. 4-2-Fitment Placed in Cockpit to Place the joystick	34
Figure 4-3 Externally Hooked Monitor to inspect the Motion Base Data.....	35
Figure 4-4- Motion Base Control Box	36
Figure 4-5 A schematic of the MFS interface with the Gen-2 Simulink Scheme for experiments.....	38
Figure 4-6 The instructor console station at MFS.....	38
Figure 4-7 The external monitor showing the Motion Base data	41
Figure 4-8- Simulink Scheme integrated with the UDP interface	42
Figure 4-9- UDP interface Block in the F-15 Simulation Scheme	43

Figure 4-10- POP up window to specify IP address of the Motus Server machine for Visual UDP interface.....	43
Figure 4-11 POP up window to specify IP address of the Motus Server machine for Motion UDP interface	43
Figure 4-12 IP settings and UDP setting for Server Computer	44
Figure 4-13 Joystick Placed in the Cockpit	45
Figure 4-14-Pilot warning about an oncoming maneuver	45
Figure 4-15- A snapshot of warning message received in the cockpit.....	46
Figure 4-16- Instructor Console Snapshot for monitoring the flight experiments.....	46
Figure 4-17- Main GUI window for Flight Scenario Selection.....	47
Figure 4-18- Input command selection window	48
Figure 4-19-Selection of Neural Network Scheme	48
Figure 4-20- Defining the actuator failure parameters.....	49
Figure 4-21-Defining the Sensor Failure Parameters.....	49
Fig. 5-1-Pre-generated Doublet for longitudinal and lateral channel.....	50
Figure 5-2-Tracking Error for Longitudinal Doublet.....	51
Figure 5-3- Tracking Error for Lateral Doublet.....	52
Figure 5-4 TE with different NN (Stab. Failure, Pilot1)	54
Figure 5-5- TE with different NN (Ail. Failure, Pilot1).....	55
Figure 5-6- Stabilator Failure TE with different NN (Pilot 2).....	56
Figure 5-7 Aileron Failure TE with different NN (Pilot 2)	57
Figure 5-8- Stabilator Failure TE with different NN (Pilot 3).....	58
Figure 5-9-Aileron Failure TE with different NN (Pilot 3).....	59
Figure 5-10 Bias for Sensor Failures.....	62
Fig. 5-11-TE for Left Stabilator soft failure (Run1)	66
Fig 5-12-Pilot Activity for Left Stabilator Failure (Run1)	68
Figure 5-13 Pilot Activity for Medium Stabilator Failure (Run1).....	69
Fig. 5-14-Pilot Activity for Hard Stabilator Failure (Run1).....	69
Figure 5-15 Soft Failure Detection with FL Soft Bound for Stabilator using MQEE	73
Fig. 5-16- Medium Failure Detection with FL Hard Bound for Stabilator using MQEE.	74
Fig. 5-17- Hard Failure Detection with FL Hard Bound for Stabilator using MQEE	74

Fig. 5-18-No Detection with OQEE for Soft Stabilator Failure	74
Fig. 5-19- No Detection with OQEE for Medium Stabilator Failure	75
Fig. 5-20- Detection with OQEE for Hard Stabilator Failure.....	75
Fig. 5-21-Soft Failure Isolation with R_{pq}	75
Fig. 5-22- Medium Failure Isolation with R_{pq}	76
Fig. 5-23-Hard Failure Isolation with R_{pq}	76
Fig. 5-24-Identification of Soft Stabilator Failure with ω_{pq}	76
Fig. 5-25- Identification of Medium Stabilator Failure with ω_{pq}	77
Fig. 5-26- Identification of Hard Stabilator Failure with ω_{pq}	77
Fig. 5-27- TE for a Soft Aileron failure (Run1).....	78
Fig. 5-28- TE for a Medium Aileron failure (Run1).....	78
Fig. 5-29- TE for a Hard Aileron failure (Run1)	79
Figure 5-30 Pilot Activity for Soft Aileron Failure (Run 1).....	81
Figure 5-31 Pilot Activity for Medium Aileron Failure (Run 1)	81
Figure 5-32 Pilot Activity for Hard Aileron Failure (Run 1)	82
Figure 5-33- Detection of a soft aileron failure	85
Figure 5-34-Detection of a Medium aileron failure	85
Figure 5-35- Detection of a soft aileron failure	86
Figure 5-36- Isolation of a soft Aileron Failure.....	86
Figure 5-37- Isolation of a Medium Aileron Failure.....	87
Figure 5-38- Isolation of a Hard Aileron Failure.....	87
Figure 5-39-Identification of soft Aileron failure with ω_{pq}	87
Figure 5-40- Identification of Medium Aileron failure with ω_{pq}	88
Figure 5-41- Identification of Hard Aileron Failure with ω_{pq}	88
Figure 5-42- MQEE for a Roll Sensor Failure.....	92
Figure 5-43- OQEE for Detection (Plots Zoomed to show detections)	93
Figure 5-44- No crossing of Bounds with R_{pq} isolates sensor failure.....	94
Figure 5-45- Identification of Roll Sensor for SFDB (SF #1).....	94
Figure 5-46- Identification of Roll Sensor for LFDB (SF #2).....	95

Figure 5-47- Identification of Roll Sensor for SMDB (SF #3).....	95
Figure 5-48- Identification of Roll Sensor for LMDB (SF #4)	96
Figure 5-49- Identification of Roll Sensor for SSDB (SF #5).....	96
Figure 5-50- Identification of Roll Sensor for LSDB (SF #6).....	97
Figure 5-51- MQEE for Pitch Sensor Failure	101
Figure 5-52- OQEE for Pitch Sensor failure (Zoomed to show detections).....	101
Figure 5-53 Rpq monitored to isolate the sensor failure	102
Figure 5-54 Identification of Pitch Sensor failure (SF#1).....	103
Figure 5-55 Identification of Pitch Sensor Failure (SF#2)	103
Figure 5-56 Identification of Pitch Sensor Failure (SF#3)	104
Figure 5-57 Identification of Pitch Sensor Failure (SF#4)	104
Figure 5-58 Identification of Pitch Sensor Failure (SF#5)	105
Figure 5-59 Identification of Pitch Sensor Failure (SF #6)	105
Figure 5-60 MQEE variation for Yaw Sensor Failure	109
Figure 5-61 OQEE for Detection of Yaw Sensor Failures	110
Figure 5-62- Isolation of Yaw Sensor Failure with Rrr	110

Nomenclature

<i>Symbols</i>	Description
p, q, r	Measured Roll, Pitch and Yaw rate
B	Bias
η	Composite Parameter
ω_{pq}	Angular Rate Dominance Parameter
R_{pq}	Roll-Pitch Cross Correlation Coefficient
R_{rr}	Auto-Correlation Parameter in Yaw
β	Floating Limiter Bound Factor
σ	Standard Deviation
Δt_1	Time Interval for Failure Isolation
Δt_2	Time Interval for Failure Identification
X	Parameter Signal
X_{UB}, X_{LB}	Upper and Lower Bounds of Floating Limiter Signals
T	Time
<i>Subscripts</i>	
FL	Floating Limiter
FL_floating	Floating Limiter Bound of the signal
HB	Hard Bound
HLB	Hard Lower Bound
HUB	Hard Upper Bound
SLB	Soft Lower Bound
SUB	Soft Upper Bound
SB	Soft Bound
actual	Actual values of angular rates
MNN	Angular Rate Parameter Calculated with Main Neural Network
DNN	Angular Rate Parameter Calculated with Decentralized Neural Network

Acronym

AFDIA	Actuator failure Detection Identification and Accommodation
ANN	Artificial Neural Network
CP	Composite Parameter
DNN	Decentralized Neural Network
FDII	Failure Detection Isolation and Identification
FFS	Full Flight Simulator
FL	Floating Limiter
LFDB	Large Fast Drifting Bias
LMDB	Large Medium Drifting Bias
LSDB	Large Slow Drifting Bias
EMRAN	Extended Minimal Resource Allocating Neural
MBC	Motion Base Computer
MFS	MOTUS Flight Simulator
MNN	Main Neural Network
NN	Neural Network
SFDIA	Sensor Failure Detection Identification and Accommodation
TE	Tracking Error
PC	Pilot Compensation
AFDIA	Actuator failure Detection Identification and Accommodation
SD	Standard Deviation
SFDB	Small Fast Drifting Bias
SHL	Single Hidden Layer
SMDB	Small Medium Drifting Bias
SSDB	Small Slow Drifting Bias
MBC	Motion Base Computer
SFDIA	Sensor Failure Detection Identification and Accommodation
WVU	West Virginia University

Chapter 1. Introduction

1.1. Importance of Failure Detection and Identification

The most published work on failure detection, isolation and identification (FDII) is in the aerospace domain. A control flight system that is able to recover an aircraft from a failure by reconfiguration is a challenge among many aerospace researchers. The fault tolerance has gained a considerable academic research interest after two commercial carrier accidents in late 1970s and it was realized that there is a persistent need to warn the pilot of any malfunctions arising in the aircraft. The Delta 1080 flight accident¹ was because one of the elevators jammed at 19 deg and the pilot had no indication about its occurrence. Another accident of the AA DC10 aircraft² could have been avoided if the pilot had known about the failure. The pilot had 15 sec to react after he realized about the failure and subsequently the plane crashed. Therefore, a prompt display of failure information to the pilot started gaining significant importance among the aerospace researchers. Failures to the aircraft primary control surfaces are critical and can lead to the loss of the aircraft. Smart failure detection and identification schemes can immediately detect failures and can reconfigure the controller and help in situations where pilot becomes perplexed in the advent of failures. A similar accident due to actuator failure was the 30 December 2001 crash of Air Vehicle 5 due to rudder actuator, which became loose while conducting a mission. Operators redirected the UAV to return to base, though during the return the rudder began flapping excessively, causing a catastrophic failure. Similar incidents of Boeing aircraft accidents due to actuator failure (rudder) are shown in Fig. 1-1. The sensor failures can also lead to fatal accidents and is evident by the recent accident of B-2 bomber aircraft on 23rd February, 2008. The accident was because of moisture in sensors which distorted three of the aircrafts twenty four sensor readings forcing the bomber to pitch up on takeoff which resulted in stalling and subsequent crash. The sensor readings confused the pilot and resulted in the loss of this \$1.4 billion airplane and are one of the biggest losses in terms of money.

737 incidents in the '90s in which rudder has been suspected



1. United Airlines Flight 585 crash

Colorado Springs, Colo.,
March 3, 1991.
737-200 flips right, dives
on landing approach;
25 killed.

2. Copa Airlines crash

Tucuti, Panama,
June 6, 1992.
737-200 flips, dives;
47 killed.

3. Pilot Mack Moore discovers flaw

Chicago, Ill.,
July 16, 1992.
737-300's rudder pedals
stiff; mechanic later
triggers rudder reversal
by putting dual servo
valve's spring,
spring guide and end cap
out of alignment.

4. Sahara India Airlines crash

New Delhi, India,
March 8, 1994.
737-200C flips left,
crashes on runway;
9 killed. Rudder
spring guide discovered
machined to wrong size.

5. Continental Airlines incident

Gulf of Honduras,
April 11, 1994.
737-300 rolls
violently to right.
Pilot turns off
autopilot, struggles
18 minutes to control
wheel, lands safely.

6. USAir Flight 427 crash

Pittsburgh, Pa.,
Sept. 8, 1994.
737-300 rolls left,
flips upside-down
and nose dives into
ground; 132 killed.

7. Eastwind Airlines incident

Richmond, Va.,
June 9, 1996.
737-200 rolls right
twice; pilot regains
control and lands
safely.

KNIGHT-RIDDER NEWSPAPERS / SEATTLE TIMES

Fig. 1-1-Boeing Rudder Failure Accidents (Courtesy Seattle Times)

Typically, physical redundancy for the actuators of the primary control surfaces is not available. Therefore, actuators failures represent major threats to flight safety. Due to the unique dynamic signature of each and every failure, it might not be feasible to train pilots

in an exhaustive manner to handle every class of actuator failures. The existence of an actuator failure detection isolation and identification scheme can significantly improve the pilot reaction time and, in turn, enhance the success probability for the post-failure recovery. Fault tolerant flight control systems may allow an aircraft to avoid unrecoverable flight conditions, regain equilibrium, and continue the mission. Many of these systems are based on the availability of an AFDII scheme to trigger compensating changes in the control laws.



Fig. 1-2-A locked slat on a Boeing 737 -844 (courtesy www.airliners.net)

1.2. Overview of Flight Simulation

A flight simulator is a system imitating or simulating the experience of flying an aircraft. It can range from PC based video game, instrument only simulator, fixed simulator to full size cockpit replica mounted on hydraulic or electromechanical actuators. A full flight simulator has motion platforms and is capable of replicating the physical sensations of flight. In spite of varying complexities and functions of a flight simulator, the fundamental attribute is similar for desktop running flight simulation software as well for a multi- million dollar full scale flight simulator. Both represent an input-output representation of the system process varying only in comprehensiveness and complexity of information. The principle task of a flight simulator is to model the

dynamics of an aircraft which is achieved by way of a mathematical model of the system under consideration. The mathematical model generally needs to be executed in real time and mainly embedded in digital format through computers in the form of a software algorithm or program.

The pilot interface subconsciously tricks the brain into reflexive actions giving appropriate stimulations. Thus the inputs must be adequately timed and accurate for any realistic flight simulation experience and stipulates a perfect blend of aural, visual and motion cues. The aural cues replicate all sounds audible in a cockpit and ranges from engine, propellers, warnings, radio chatter and even aerodynamic sounds because of retracting landing gear, thrust vectoring, surface deflections as well as weather related such as thunderstorms and downbursts.

The main components of a simulator are visual cueing and motion cueing for pilot immersion during the simulator experience. The level of immersion measures closeness to flying the real aircraft. Recent advances in computers and graphics have vastly improved the visual cueing in flight simulators. Figure 1-3 shows the sub logic flight simulator (1982) and the Microsoft flight simulator (2004) and the evolution of graphics and resolution for visual cues is evident.



Figure 1-3 A comparison of visual cue improvement (1982 to 2004)

The motion cues in a flight simulator provide a real-time approximation of the translational and rotational accelerations experienced by the pilot. The exact replication of these acceleration are however not feasible in practice because of imposed mechanical constraints. The high fidelity flight simulators typically use a six-degree of freedom (DOF) motion simulation. The translational and rotational acceleration exerted by the airplane dynamics provides the motion cue necessary to make the pilot feel immersed in the simulator. This is achieved by the acceleration phase is followed by a washout phase beyond human threshold of perception. The washout phase is essential to maintain the working of simulator within a confined area. The simulator retrieves back to its original position during the washout phase without the perceivable sensory receptions of the pilot.

A full flight simulator (FFS) duplicates almost all aspects of the aircraft and its environment including the six-degrees of freedom motion. In the academic research flight simulators are used extensively for understanding various aerospace subjects such as flight dynamics and man-machine interaction (MMI). The range of these simulators can be anything from video games to extremely expensive simulator designs such as LAMARS Figure 1-4 installed at Wright-Patterson Air Force Base at Ohio.



Figure 1-4- LAMARS Motion Flight Simulator at Wright Patterson Air Force Base, Ohio

The simulators are characterized by Instructor operating stations (IOS) which allows an instructor to modify flight conditions within the flight simulator cockpit. Many flight

simulators have a provision to control the simulator from the cockpit, either from a console adjacent to the pilot or a co-pilot.

With the advent of low cost electromechanical actuators the flight simulators have become an affordable engineering solution. In the past its applications were limited because of the multi-million dollar hydraulic devices used in flight simulators. Thus the flight simulators were limited to larger corporations and military centers. High end flight simulators today incorporate motion bases or platforms to provide cues for real motion. These complement the visual cues shown in Figure 1-5 and are vital to simulate a real flight.



Figure 1-5-Visual Cues in a home-built Boeing Style Simulator

Also the motion cues are advantageous in case of poor visibility conditions. The motion platforms used commonly in the flight simulators are six cylinders Stewart platform called as hexapods. The electromechanical actuators for MOTUS flight simulator are shown in Figure 1-6.



Figure 1-6 Six Electromechanical Actuators for MOTUS Flight Simulator at WVU

1.3. Research Objectives

There are many advantages of flight simulation among aerospace researchers because of the significant reduction in cost. Also, for researchers it is the only way to test new equipments, approaches while remaining in the research budget as well as providing safety and fewer hazards in operation. A flight simulator is capable of reproducing a variety of human behavior and is a tool to evaluate new concepts, approaches, formulate new means of training pilots in critical situations without using a real aircraft.

This research is inspired from many advantages of using a flight simulator in the design process. The motion flight simulator is used to create a research test-bed for carrying out analysis of advanced flight control schemes. In this particular study it has been used to perform evaluation of FDII scheme and in principle used to demonstrate the design, implementation of this man-machine interface of Simulink scheme with the MOTUS flight simulator. The setup features a MOTUS Six-DOF flight simulator at West Virginia University (WVU) which is used as a test bed to test different control schemes for an F-15 aircraft. Specific algorithms will be used to evaluate the performance of different NN schemes. Since the FDII schemes take care of the failures automatically,

monitoring the pilot activity is an important indication of the efficacy of the FDII schemes. The pilot activity can significantly qualify the performance of the FDII scheme. If the pilot issues negligible stick activity in the advent of a failure then it's an indication of good failure detection and accommodation scheme. The tasks outlined to complete this effort are described in the following paragraph.

Initially, an interface for communication of the F-15 Gen-2 Schemes with the 6-DOF MOTUS flight simulator at WVU is developed. The X-plane software provides visual and aural cues while the Motion Base Computer (MBC) provides motion cues by way of electromechanical motors in the motion base. This is accomplished by developing plugins for X-plane software that enables execution of Simulink® based codes in the MOTUS flight simulator. The communication of data is through User Datagram Protocol (UDP) facilitated by the X-plane software.

In the next part of study, nominal conditions are tested in the flight simulator for 3 different pilots in order to reproduce the earlier work on the Gen-2 NLDI schemes. These tests are baseline tests for the study and serve to test the interface. This will be followed by failure tests with no failure detection and identification schemes.

Thereafter, a FDII scheme for an actuator/sensor is developed using adaptive threshold for cross-correlation coefficients of the angular rates. The scheme is capable of detecting failures and piloted experiments will be carried out for different failure conditions. The performance of EMRAN NN is compared with a No NN case considering the tracking error (TE), pilot activity etc for the FDII scheme. The technique for FDII warning system is utilized to warn pilot in advent of a failure by making a provision to receive text messages in the cockpit by a suitable plugin in X-plane.

1.4. Overview of Thesis

The organization of the thesis will be in the following manner:

Chapter 2 will cover the literature review on the failure detection and identification. The earlier work done on Actuator failure Detection Identification and Accommodation (AFDIA) and Sensor/Actuator Failure Detection, Identification and Accommodation (S/AFDIA) using fixed threshold is reviewed. The extension of the fixed threshold to an adaptive threshold for actuator and sensor FDII is discussed.

Chapter 3 will discuss the concept behind the F-15 Gen-2 Neurally Augmented Schemes, AFDIA/SFDIA schemes and the failure detection and identification concepts. The adaptive threshold approach for integrated FDII formulated for the research problem at hand is described in detail.

Chapter 4 will explain the setup of the experimental environment. This includes detailed descriptions the Motus flight simulator and the X-plane software. This will be followed by a detailed explanation of the interface and how the pilot is included in the loop and how the visual, aural, motion cues are obtained for F-15 Simulink schemes. The hardware and experimental setup will be explained in detail and the graphic user interface to carry out experiments will be explained.

Chapter 5 will show the simulation results obtained for different failure scenarios for a no FDII and a FDII case in respect to TE performance, pilot compensation and detection efficiency. The results of adaptive threshold approach for integrated FDII scheme will be presented.

Chapter 6 will contain the conclusion of the results that are obtained from the conducted experiments in the motion simulator. This will be followed by future recommendations on the work.

Chapter 2. Literature Review

2.1. General Description

A fault can be defined as an unexpected change in a system which can range from a malfunction to a catastrophic failure. Normally a fault tends to degrade the system performance and, sometimes, makes it inoperable. The term “fault” and “failure” are here used interchangeably. However, the term failure suggests complete breakdown while a fault is somewhat a malfunction rather than a catastrophe. Failure detection is very important in general as it can help us avoid major breakdown and is essential to maintain safety in critical systems such as chemical plants³, nuclear plants⁴, space systems⁵ etc. The FDII problem is widely investigated problem in flight control research because of increasing complexity and interconnectivity of system at end. In general FD implies some sort of continuous monitoring of the measurable outputs of the system. Under nominal conditions the variable tends to follow rather well established patterns. A failure in a flight control system (FCS) will induce deviations from nominal and predictable patterns.

Hardware redundancy is a traditional way to achieve fault tolerance in dynamical systems and the term strictly means duplicating the channels with hardware components such as sensors, actuators or even computers. Based on the desired level of reliability, duplex, triplex or quadruple parallel systems are often implemented. In case of a failure backup switch switches the control to the redundant system. Clearly an important way of achieving fault tolerance is by means of multiple lanes of hardware. Some of the examples of redundant control surfaces in modern aircrafts include speed brakes, wing flaps, differential canards, spoilers, rudder below fuselage. Redundancy in thrust control mechanism involves differential thrusts, thrust vectoring and canted engines. Nonetheless this type of redundancy comes with a price of additional weight, cost, volume and maintainability. In small systems like UAVs and small scaled prototypes where weight is a critical component of design hardware redundancy is not a feasible alternative. Recent advances in the control theory and advanced mathematical modeling facilitates development of a whole new FDII methods based on functional rather than

physical redundancy. The implementations of onboard digital computers have led to development of fault detection schemes relying on analytical rather than hardware redundancy. The analytical redundancy provision in the aircraft rather than hardware is also promising approach as it is cost effective, weight reducing and less power dependent whilst providing a substantial increase in performance and reliability. It should be noted that in aerospace systems there has been a little tendency to replace hardware redundancy due to the safety requirements. Therefore it acts somewhat to suppress some levels of replication in the system for example- replacing a quadruplex system with a triplex employing using analytical redundancy.

2.2. Fault Detection Methods

The traditional approaches to failure detection were based on hardware redundancy or signal processing techniques such as frequency spectrum analysis, limit checking or fault dictionary approach. The modern approach to the FDII is model based in which the failure detection scheme makes use of mathematical model of the system and knowledge based approach in which human knowledge of reasoning is used to detect and diagnose failures. FDII methods utilize parameter identification and state estimation, detection filters, statistical pattern recognition, multiple model estimators, maximum likelihood techniques and bayes theorem. In analytical methods a mathematical model acts as a substitute to the redundant hardware and is used to compare and monitor it with the actual system. One prime requirement of the algorithms is that they are simple enough to fit on the onboard computer and fast enough to provide a real-time calculation for safe and reliable system at the onset of a fault. Additionally, they should be robust enough to provide nominal performance in presence of parameter variations, turbulence and coupling effects during maneuvers. The analytical redundancy has its implicit fallbacks. One of the challenges in this process of model based method is to obtain an accurate mathematical model for complex systems. Poor modeling leads to more probability of false alarms and detections. However, the appeal to analytical methods lies in the fact that there no additive physical instrumentation in the plant. The analytical based methods can be classified into model based^{6,7} and the knowledge based^{8,9} or a combination of both¹⁰.The model based method is normally implemented as a software algorithm. The

earliest work on FDII is through the use of estimation filter¹¹, failure detection filter¹², band limiting filters¹³ and innovations tests^{14 15 16}.

2.2.1. Model Based Methods

These types of methods make explicit use of mathematical model of the system. The method relies on residual generation and decision making. The signals from the system are initially processed to enhance the effect of fault so that it can be recognized. These processed measurements are called residuals and the enhanced effect of fault on the residuals is called the signature of the fault. Analytical redundancy methods are based on detection, isolation and identification of failures from the comparison of system's available measurements with the prior information represented by system's mathematical models¹⁷. The failures are detected by setting fixed thresholds on the residuals generated from difference between real and estimates of the measurement using the mathematical model. A number of residuals can be designed with each having its own sensitivity to detect a failure. There are many methods for residual generation in which observer approach^{19,20}, parity relations approach^{21,22} and the parameter identification approach²³ have been widely used among early researchers.

The basic idea behind observer approach is the estimation of system outputs from the measurements by either full order or reduced order state observers. A suitable weighting of output estimation error is defined as the residual. This approach has advantages because of its flexibility in design, the relative ease of obtaining robustness, algorithms and software simplicity and the speed of response in detecting and isolating failures. The parity relations approach is based either on a technique of direct redundancy i.e. making use of static algebraic relationships between sensor and actuator signals or based on temporal redundancy when dynamic relations between inputs and outputs are used. An interesting feature of parity space methods for FDII is that given certain design conditions, the parity equation residual signals can become identical to those from observer approach. This has been described by Massoumnia²⁴ and Patton²⁵. The parameter identification approach makes use of the fact that failures are reflected in physical parameters such as friction, mass, viscosity, resistance, inductance or capacitance. This

approach is used to detect failures via the estimation or identification of model parameters using a non-parametric model.

The FDII for stochastic systems are based on statistical testing of the residuals. The noted work in this area is by Willsky using Chi-Squared Testing¹⁷, Basseville using Sequential probability ratio testing¹⁸ and Willsky using Generalized Likelihood Ratio¹⁷. In order to reduce the noise of residuals the residual generator has to accommodate the noise in the generation of residuals. The Kalman filter technique is a common approach to generate residuals. The structure of a Kalman filter is similar to an observer and is based on stochastic model of the dynamic system. If the system is nominal then the normalized innovation sequence in a correlated Kalman filter is a Gaussian white noise with zero mean and a unit covariance matrix⁶. Failures leads to changes in system dynamics causing drifts of the state vector components, abnormal measurements, sudden shifts in measurement channels leading to a drift from near zero mean and a change in the covariance matrix. The detection of the failure is based on detecting changes in these parameters from their nominal values. Some variants of Kalman filter include multiple model adaptive filters and two stage bias correction filters.

Kalman filtering techniques have been widely used for failure detection. Of these the work done by Mehra ^{6,7} makes use of multiple hypothesis EKF and several Kalman filters running in parallel and switching decision is based on innovation sequence and likelihood function of each filter. One drawback of these methods is detection delays due to likelihood functions for active hypothesis. To address detection delays issues an Interactive Multiple Models⁷ (IMM) approach with Extended Kalman Filter (EKF) is more time efficient than Multi hypothesis EKF. The failure detection scheme consists of representing each failure by a model and combining the outputs of EKFs based on different models in an optimal way. So in addition to fast failure detection it also provides a near optimal estimate of states as well. The IMM approach has been applied to spacecraft autonomy of failure detection and identification for sensors (gyros, star tracker) and actuator failures. Similar work by Eide³⁷ uses a bank of Kalman filters modeled to match a particular hypothesis of real world.

2.2.2. Knowledge Based Methods

Control theory and artificial intelligence (AI) strive to harness mathematics and logic for problem solving however control theory finds its origin in dynamics and electronics while the AI springs from biology, psychology and computer science. Knowledge based methods offer a new dimension to the fault detection methods where analytical methods fail because of unavailability of accurate models that represent the physical system. Therefore the knowledge based methods offer alternatives to analytical based methods or may complement it. AI techniques like Neural Networks and Fuzzy logic have been extensively used for failure detection by many researchers lately. Neural networks are motivated by the input-output relationships and learning properties of living neurons. Initially due to limited computational capacity of computers, the training of neural networks were found to be unworkable^{34,35}. Artificial NNs consists of nodes that simulate the neurons and weighing factors that simulate the synapse of living neurons. There are many advantages of NNs in performing a variety of fault tolerant systems because of their speed, ability to model non-linear functions, ease of modeling multidimensional problems and capability of learning with experience.

In order to formulate failure detection logic where mathematical modeling is hard to realize various knowledge based methods are used. Although mathematical models that describe the dynamic behavior of faults is present, yet the problem-solving actually used by pilots and human operators are fairly more than quantitative. According to Handelman²⁶ there is a need for incorporating a human like reasoning ability in flight control systems. The artificial intelligence concepts like neural networks and fuzzy logic strive to emulate the human thought process by computer and therefore provided good alternative to mathematical techniques. There have been used extensively among researchers because of ease of modeling uncertainties and non-linearities by expert systems. The problem is handled by Neural Networks^{52 53 54 55 56} or fuzzy logic⁴⁹ or a combination in the form of Neuro-fuzzy systems^{60 61 62}. The work by Chiang⁶¹ approaches the problem with a Neuro-fuzzy logic in which the neural network is capable of approximating a non-linear function if there are sufficient number of neurons. The fuzzy logic augments the system by developing a physical understanding by setting linguistic

rules. The global nature of the system is handled by the fuzzy system while neural network handles the local variables in real time. Similar work on a Neuro fuzzy system by Al-Jarrah ⁶² uses neural network for online learning and fuzzy system to handle uncertainties in the system. This is compared with work by Polycarpou⁶⁰ and has a better tracking error performance and approximations of the functions.

A FDII technique requires knowledge of system behavior by the FCS be available at any operating point at a heavy computational and storage costs. NNs have gained interest because they maintain the save level of FD capability without increasing the computational burden. The mapping property of the NN architecture is attractive when the input-output are related to completely or partially unknown dynamics. The NN implementation in the design on the estimator can be offline (frozen architecture) or online. The online learning is because of the advantageous as it provides flexibility to deal with variable dynamics and failure conditions.

The actuator FDII have been extensively researched using neural networks ^{8:52-53} and some of them have been flight tested as well. Due to availability of redundant sensors on an aircraft system however sensor FDII hasn't gained attention comparable to actuator failures. Notable work in the area of FDII using neural networks is by Napolitano, Perhinschi in which they have demonstrated NNs efficacy in actuator FDII, sensor FDII and a combination of both. The scheme utilized two sets of NNs which monitor the angular rates in the aircraft. The cross-correlations of these rates indicate any anomaly in the aircraft system. In most of the work the fixed threshold approaches have been used for actuator^{48:54} and Integrated^{56:58} (sensor/actuator) FDII by them. From simulation results it was concluded that NNs were very helpful in modeling failure detection algorithms.

Perhinschi ⁵⁷ and others have extended the actuator failure FDII using an adaptive threshold which reduces the ratio of false alarms considerably. The use of Autoregressive Moving Average Filter (ARMA) in their approach of floating Limiter (FL) for thresholds of cross-correlations rates is the main source of inspiration of this research work. The adaptive threshold offers more robustness because it is based on an auto regressive filter

and therefore takes into account any disturbance induced by pilot. In the fixed threshold it is noticed that there are many false detection flags if a pilot is included in the loop. These approaches have been extended for an integrated sensor/actuator FDII and have a good performance in terms of failure detection accuracy and speed of detection. A similar work for sensor failure⁹ is done using an adaptive threshold concept. This FL approach is extended for an integrated sensor/actuator FDII scheme in this research work. The integrated scheme developed is also tested in a motion flight simulator to monitor pilot activity in case of the failure. The simulation capability of a motion simulator in terms of more realistic motion and visual cues are leveraged in this experimental study. The MOTUS simulator interface with the F-15 FDII schemes offers a new extension to existing simulation capabilities at WVU. It provides a more complex simulation environment to study pilot behavior and monitor the performance in a simulation environment comparable to a real flight.

Chapter 3. Theoretical Background

3.1. Description of Gen-2 Scheme

The NASA Intelligent Flight Control System (IFCS) F-15 program²⁷ is aimed at development and testing of new control approaches using self learning neural networks that can optimize aircraft performance in both nominal and failure conditions. The explicit use of artificial intelligence techniques and on-line system identification algorithms have been used effectively in recent years for adaptive control systems. One of the advantages of these techniques over conventional control approaches is that they eliminate gain scheduling and also can handle a variety of control surface deflections. The control surface failures change the aircraft stability and control characteristics and render conventional control schemes ineffective. The general scheme architecture shown in Fig. 3-1 is taken from [46], named *Gen-2* within the IFCS program, is based on an adaptive neural controller canceling the errors associated with the dynamic inversion of the model. In order to reduce the level of computational effort the control strategy^{38, 39, 40} has been selected to provide consistent handling qualities and eliminate the need of gain-scheduling or system identification. Initially, constant values of aerodynamic stability and control derivatives for a fixed condition in the flight envelope are used for model inversion. In addition, desired handling qualities are achieved with *ad hoc* reference models

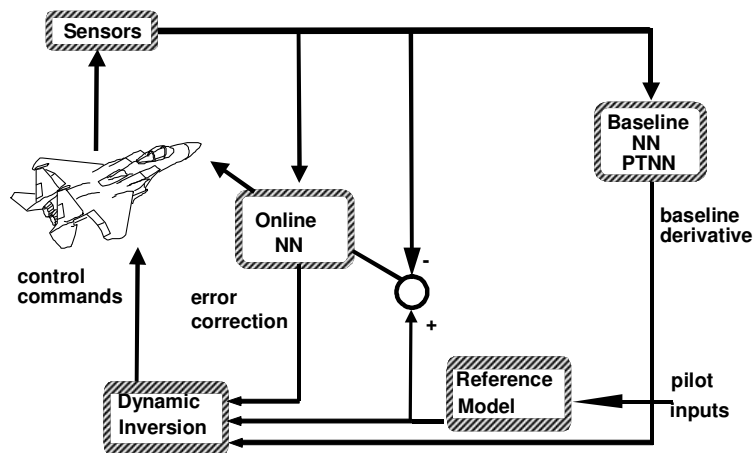


Fig. 3-1-General Block Diagram for IFCS Gen-2 Controller Scheme [46]

3.1.1. NLDI scheme

The mathematical model for *Gen-2*³⁸ is based on a Non-Linear Dynamic Inversion (NLDI) which uses pre-trained neural networks (PTNN) to obtain the stability and control derivatives. The scheme developed at WVU uses the stick commands as input to the NLDI controller. The displacement from longitudinal, lateral and directional stick (δ_{lon_stick} , δ_{lat_stick} , δ_{dir_pedal}) commands are converted into corresponding roll, pitch, and yaw rate commands (p_{com} , q_{com} , r_{com}). A reference model provides filtered rate commands (p_{ref} , q_{ref} , r_{ref}) and acceleration commands (\dot{p}_{ref} , \dot{q}_{ref} , \dot{r}_{ref}) using first order roll rate and second order pitch and yaw rate transfer functions. The inputs to dynamic inversion (\dot{p}_c , \dot{q}_c , \dot{r}_c) is a function of augmentation commands generated by NNs and pseudo control commands and is computed using the expression:

$$\begin{bmatrix} \dot{p}_c \\ \dot{q}_c \\ \dot{r}_c \end{bmatrix} = \begin{bmatrix} U_p \\ U_q \\ U_r \end{bmatrix} - \begin{bmatrix} U_{pad} \\ U_{qad} \\ U_{rad} \end{bmatrix} \quad (1)$$

In order to neutralize the effect of errors due to dynamic inversion, the augmentation commands (U_{pad} , U_{qad} , U_{rad}) are generated by adaptive NNs from the difference of reference and the aircraft actual angular rates. The errors (e_p , e_q , e_r) are used to create pseudo control acceleration commands (U_p , U_q , U_r) are computed with the following expressions:

$$U_p = \left(K_{pp} + \frac{K_{ip}}{s} \right) \cdot e_p + s \cdot p_{ref} \quad (2)$$

$$U_q = \left(K_{pq} + \frac{K_{iq}}{s} \right) \cdot e_q + s \cdot q_{ref} \quad (3)$$

$$U_r = \left(K_{pr} + \frac{K_{ir}}{s} \right) \cdot e_r + s \cdot r_{ref} \quad (4)$$

Where K_p and K_i are, respectively, proportional and integral constants

The necessary control surface deflections (δ_a , δ_e , δ_r) are obtained by NLDI and control surface commands ($\delta_{a_{com}}$, $\delta_{e_{com}}$, $\delta_{r_{com}}$) are obtained with the following equation:

$$\begin{bmatrix} \delta_{a_{com}} \\ \delta_{e_{com}} \\ \delta_{r_{com}} \end{bmatrix} = B^{-1} \begin{bmatrix} \dot{p}_c - L_I \\ \dot{q}_c - M_I \\ \dot{r}_c - N_I \end{bmatrix} \quad (5)$$

Where B is the state space system control matrix and the terms $(\dot{p}_c - L_I, \dot{q}_c - M_I, \dot{r}_c - N_I)$ are the differences between input acceleration commands and actual plant acceleration contributions (L_I, M_I, N_I) . These plant contributions are function of inertial and geometric characteristics, aerodynamic derivatives, angular rates, and aerodynamic angles. Using first and second order transfer functions, the actual control surface actual deflections are computed.

3.1.2. Neural Network Algorithms

The NLDI scheme features three different types of adaptive NNs and this study is carried out for comparing the performance of these three different NN augmentations. These NNs serve the purpose of canceling the dynamic inversion errors. The first method Sigma-Pi, is based on a two-layer Sigma-Pi NN^{41,42} for each angular acceleration $(\dot{p}, \dot{q}, \dot{r})$. These NNs use proportional and integral acceleration errors $(U_{p_error}, U_{q_error}, U_{r_error})$ for on-line learning purpose. The pseudo control acceleration commands (U_p, U_q, U_r) , bias terms, and sensor feedback are inputs to the neural networks. For each channel three terms C_1, C_2 and C_3 are computed as functions of input variables and previous-step network outputs $(U_{pad}, U_{qad}, U_{rad})$. The neural network gets its name because the neurons are summed and multiplied with each other. The neural network output is an additional control command for compensating the inversion errors and is calculated from Eq. (6).

$$U_{ad} = W^T f(C_1, C_2, C_3) \quad (6)$$

where f is computed from each signal inputs using a nested Kronecker product. The network weights W are determined by an adaptation law:

$$\dot{W} = -G(U_{error} \cdot f + L|U_{error}|W) \quad (7)$$

where G and L are user selected specific gains.

The EMRAN⁴³ algorithm is an extension of the standard MRAN⁶³ and is noted for its selective selection criterion of the neurons. The network features growing and pruning mechanisms and is based on the “survival of the fittest” strategy and updates parameters for the most activated neurons. This strategy allows only the parameters of the most activated neurons to be updated, while all the others are left unchanged. This strategy is suitable for online applications⁶⁴ as it reduces the number of parameters to be updated and thus reduces computational burden.

For Gaussian basis function used for EMRAN is given by the expression:

$$\hat{y}(x, \theta) = \sum_{i=1}^M w_i e^{-\left(\frac{|x-\mu_i|^2}{2\sigma_i^2}\right)} \quad (8)$$

Where x is the input vector, θ is the set of parameters to be tuned by the learning algorithm including the weight w , the Gaussian center positions μ , and the variances σ . The EMRAN NN is activated once the criteria for the initiation of NN are met. The neuron is activated as soon as estimation error, windowed estimation error and distance of inputs from center cross its threshold. Once it is activated the center, variance, and weight of the new neuron at iteration k are given by, respectively:

$$\begin{aligned} \mu_{M+1}(k) &= x(k) \\ \sigma_{M+1} &= \lambda \inf_{j=1}^M \|x(k) - \mu_j(k)\| \\ w_{M+1}(k) &= e(k) = y(k) - \hat{y}(k) \end{aligned} \quad (9)$$

If the above criteria is not met the parameters are updated using the eq(10)

$$\theta(k+1) = \theta(k) - \eta \left. \frac{\partial \hat{y}(k)}{\partial \theta(k)} \right|_{(k)} \cdot e(k) \quad (10)$$

Where $e(k)$ is the estimation error and η is the learning rate

To achieve an optimum performance in the presence of non-linearities and to minimize the computational burden on the operating areas, a combination of ADALINE⁴⁸ and an EMRAN network (A+EMRAN) working in parallel has been implemented on each of the three channels.

The Single Hidden Layer^{38,39} (SHL) NN The output of the network is given by the relationship:

$$y_i = \sum_{j=1}^m \left[w_{ij} \sigma \left(\sum_{k=1}^n v_{jk} x_k + \theta_{vj} \right) + \theta_{wi} \right], \quad i=1,2,\dots,p \quad (11)$$

where w_{ij} are the interconnection weights between the hidden layer and the output layer, v_{jk} are the interconnection weights between the input layer and the hidden layer, and θ_{vj} , θ_{wi} are bias terms. The activation potential ‘a’ is used to compute the activation function of the form:

$$\sigma(\xi) = \frac{1}{1 + e^{-a\xi}} \quad (12)$$

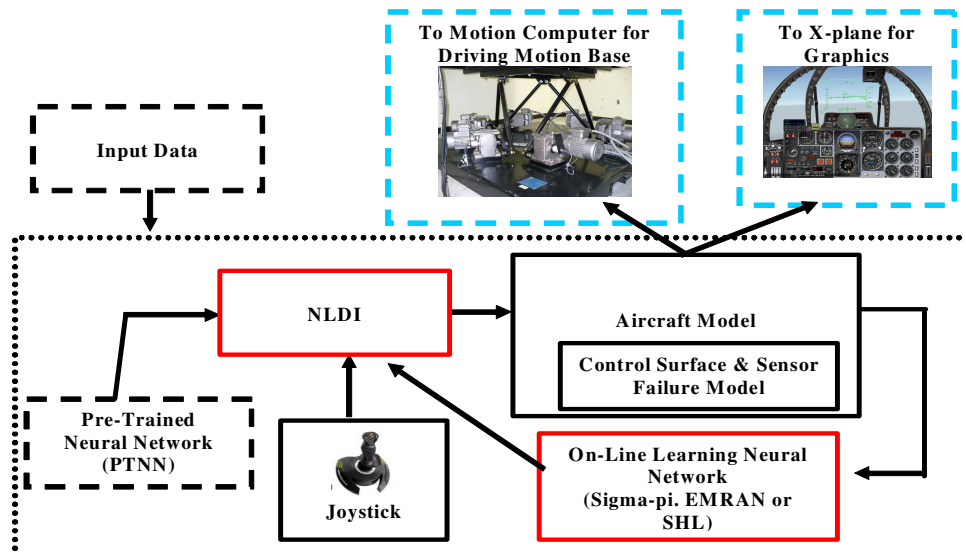


Fig. 3-2- The integration of NLDI based F-15 Controller with MFS

3.2. Aerodynamic Modeling of the failure

The actuator failure modeling has been developed and applied in recent literature for longitudinal, lateral and directional control surface locking⁴⁸. On the advent of a failure the stability and control derivatives changes and leads to a reduced “control surface effectiveness”. This is used to model the partial effectiveness once a control surface failure occurs. The sensor failure implies a bias induced in the sensor measurements which go in the feedback loop.

3.2.1. Failure Detection and Identification concepts

Since it is practically not feasible to have multiple channels of redundant primary control surfaces therefore actuator failures are major threats to safety of the flight. Each failure in an aircraft has a unique dynamic signature and hence it is not possible to pre-train pilots extensively for such situations. State estimation methods performance degrades in the presence of non-linearities and uncertainties. The augmentation of neural networks in FDII schemes handles modeling error⁵²⁻⁵⁶ and have been used in detection and identification criteria for primary control surfaces^{54,55}. In this effort, this approach has been extended to an integrated actuator failures to detect and identify failures of the locked primary surfaces (stabilator, aileron and rudder) as well sensor failures for the roll, pitch and yaw rate gyros. The experimental set-up of the West Virginia University’s (WVU) MOTUS Flight simulator as a simulation test bed is used to evaluate the FDII schemes. The failures considered in research work are actuator locking and sensor bias type failures.

3.2.2. The Adaptive Threshold Concept for FDII

While the fixed threshold provides a FDII scheme capable of fairly accurate detection and identification for actuator and sensor failures^{54, 58}, they are however susceptible to false alarms and wrong identifications when pilot is included in the loop. The adaptive threshold approach for FDII offers advantage over a fixed threshold FDII because it eliminates any need for parameter scheduling with different flight conditions, thereby reducing the design effort. In addition to this the Floating Limiter approaches⁵⁷ have demonstrated better performance in terms of detection delays and ratio of false alarms. The fixed threshold approach has to take into account the rate of variation of cross-

correlation parameters and therefore has higher detection, isolation or identification time as some parameters require certain time for the signal to build up and cross the fixed threshold. On the other hand the FL limiter is crossed when ever there is an abrupt rise in parameters being monitored. The FDII scheme for integrated actuator/sensor failure is built on an adaptive ARMA (Auto Regressive Moving Average) filter termed Floating Limiter (FL) which fluctuates with an upper and lower bound around the signal. The signal experiences a sudden jump at failure which crosses the FL bounds and hence a failure alarm is triggered.

The FL filter used in the study is shown below

$$D(Z) = \frac{z^3 + z^2 + z + 1}{40(z^3 - 0.3z^2 + 0.3z - 0.3)} \quad (15)$$

The FL limiter is used to evaluate the variable thresholds for the parameters monitored for FDII scheme. The soft and hard Upper and Lower bounds are computed as shown in Eq. (16). The respective bounds are shown in Figure 3-3.

$$\begin{aligned} X_{HUB} &= \bar{X} + \beta_{HUB} \cdot \sigma(X) + b_{HUB} \\ X_{HLB} &= \bar{X} - \beta_{HLB} \cdot \sigma(X) - b_{HLB} \\ X_{SUB} &= \bar{X} + \beta_{SUB} \cdot \sigma(X) + b_{SUB} \\ X_{SLB} &= \bar{X} - \beta_{SLB} \cdot \sigma(X) - b_{SLB} \end{aligned} \quad (16)$$

Where

\bar{X} = Average of signal computed over the time window

σ = Standard deviation of the signal

β_{HUB} = Bound factor for Hard Upper Limit

β_{HLB} = Bound Factor for Hard Lower Limit

β_{SUB} = Bound factor for Soft Upper Limit

β_{SLB} = Bound Factor for Soft Lower Limit

b_{HUB} = Bias for Hard Upper Bound

b_{HLB} = Bias for Hard Lower Bound

b_{SUB} = Bias for Soft Upper Bound

b_{SLB} = Bias for Soft Lower Bound

A typical FL signal in a non-failure mode will drift constantly with the signal and the main signal will remain within bounds. On the advent of a failure the signal abruptly rises and cross the FL bounds. This is used as an indication of a possible failure.

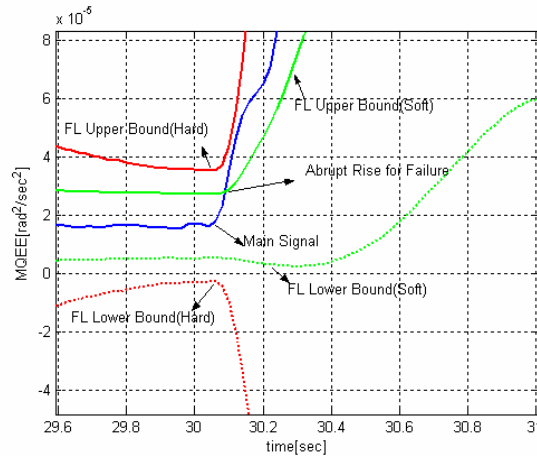


Figure 3-3- A FL approach to FDII showing the detection with MQEE

3.2.3. The FL architecture for Integrated Sensor/Actuator Failure

The concept of adaptive threshold in failure detection algorithms for actuator⁵⁷ and sensor⁹ failures is extended to an integrated S/AFDII scheme using the FL approach. The scheme is based on monitoring the outputs of Floating Limiter (FL)⁶⁵⁻⁶⁶ bounds for the signal in consideration. The signals are mostly the cross-correlation coefficients of angular rates by the aircraft. The FL fluctuates around the signal and drift with a rate of the signal but less than an imposed limit. The block diagram for the FL is shown in Figure 3-4. The FL consists of two low pass filters in which one of them is Infinite impulse response and the other is finite impulse response. The filter allows the signal to move within bounds which drift with the rate of signal. The FL Simulink block is shown in Figure 3-5.

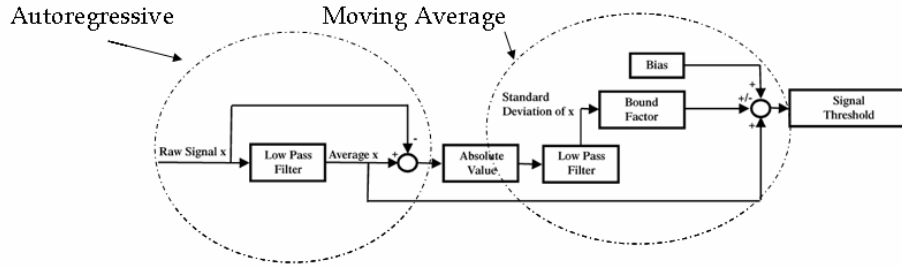


Figure 3-4- Block Diagram of Floating Limiter

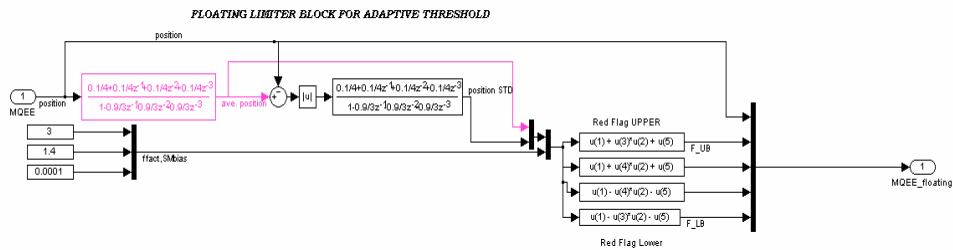


Figure 3-5- Simulink block showing Bounds of Floating Limiter

3.2.4. The Adaptive Threshold Parameters for S/AFDII Scheme

The FDII scheme is divided into three main steps mainly

Detection: The failure of an unspecified kind (Actuator/Sensor) is detected;

Isolation: The detected failure is differentiated between an actuator and a sensor failure.

Identification: The isolated failure is identified to be a stabilator, aileron, or rudder failure if there is actuator failure and a roll, pitch, yaw gyro if there is sensor failure.

The FDII scheme rests on two different sets of NNs; Main Neural Networks (MNNs) and the Decentralized Neural Networks (DNNs) for monitoring the FDII parameters. These parameters are basically cross-correlation functions of the aircraft angular rates. The outputs of the MNNs are the estimates of the angular rates, roll, pitch, and yaw (p_{MNN}, q_{MNN} & r_{MNN}) at time 'k', using measurements from time instant 'k-1' to 'k-m'. The inputs to the MNNs may include the respective gyro measurements. The outputs of the DNNs are individual estimates of the angular rates (p_{DNN}, q_{DNN} & r_{DNN}) and does not include measurements from sensors.

The estimates from the MNNs are compared with the actual measurements at time 'k' ($p_{actual}, q_{actual}, r_{actual}$) to define the main quadratic estimation error ($MQEE$) parameter:

$$MQEE = \frac{1}{2} \left[(p_{actual} - p_{MNN})^2 + (q_{actual} - q_{MNN})^2 + (r_{actual} - r_{MNN})^2 \right] \quad (17)$$

A NN output quadratic estimation error ($OQEE$) parameter is defined by comparing the estimates from the MNNs and DNNs:

$$OQEE = \frac{1}{2} \left[(p_{DNN} - p_{MNN})^2 + (q_{DNN} - q_{MNN})^2 + (r_{DNN} - r_{MNN})^2 \right] \quad (18)$$

Both MQEE and OQEE are used in the detection phase.

The isolation of the sensor/actuator failure is based on two main parameters which are

$$\bar{R}_{pq}(k) = \sum_{i=k-n_R}^k R_{pq}(i) \quad (19)$$

$$\bar{R}_{rr}(k) = \sum_{i=k-n_R}^k R_{rr}(i) \quad (20)$$

The cross-coupling parameters \bar{R}_{pq} induces a strong perturbation in case of an actuator failure particularly stabilator and aileron. Since a sensor failure has very slow effect on cross-coupling so the rise is not abrupt and the time interval can be used for isolation. Similarly \bar{R}_{rr} increase is abrupt for a rudder failure compared to a yaw sensor failure and can isolate the failure.

The other is an angular rate dominance parameter representing the weighted difference of the roll and pitch rate floating averages:

$$\omega_{pq} = |p| - \mu|q| \quad (21)$$

A stabilator failure induces both p and q perturbations and hence ω_{pq} is significantly lower than for an aileron failure which doesn't have an appreciable pitch rate. There is roll rate variation in aileron failure without affecting the pitch rate and hence ω_{pq} rises fast in comparison to stabilator failure. This can be used to distinguish between stabilator/aileron failures.

3.2.5. The detection phase

The occurrence of a primary control surface failure produces a perturbation of the normal dynamic response of the aircraft. This translates into large errors between the measured angular rates and the MNN estimates, in particular large values of $MQEE$, and/or large values of $OQEE$. Therefore, the condition for a control surface failure to be declared is:

$$\begin{aligned} MQEE &\geq MQEE_{FL} \\ \text{or} \\ OQEE &\geq OQEE_{FL} \end{aligned} \quad (22)$$

It should be noted that actuator failure corresponds to a large and a faster variation in $MQEE$ while sensor failures have a large and faster variation of $OQEE$

3.2.6. The Isolation phase

Once a failure is detected it is essential to differentiate between an actuator and a sensor failure. In general, the actuator failures induces strong cross coupling in angular rates as opposed to sensor failures. A failure to any of the actuators will induce a coupling between the longitudinal and lateral or directional channel. In turn, this implies large values of the roll-pitch rate cross-correlation function. Therefore, the criterion for preliminary identification is shown in eq. (23)

$$\begin{aligned} R_{pq} &\geq R_{pq_FL} \vee R_{rr} \geq R_{rr_FL} \quad \& \Delta t \geq \Delta t_1 \Rightarrow \text{Actuator Failure} \\ \text{else} \\ R_{pq} &\leq R_{pq_FL} \vee R_{rr} \leq R_{rr_FL} \quad \& \Delta t \geq \Delta t_1 \Rightarrow \text{Sensor Failure} \end{aligned} \quad (23)$$

It should be noted that R_{pq} varies slower for a sensor failure; therefore, a time interval up to Δt_1 should be allowed before a final preliminary failure isolation.

3.2.7. Failure Identification for Actuator Isolated

Three types of actuator failures are addressed here. Once the actuator failure is isolated it has to be differentiated as a stabilator, aileron or rudder failure.

$$\begin{aligned}
 \omega_{pq} \geq \omega_{pq_FL} \quad \& \Delta t \geq \Delta t_2 \Rightarrow \text{Aileron Failure} \\
 \omega_{pq} \leq \omega_{pq_FL} \quad \& \Delta t \geq \Delta t_2 \Rightarrow \text{Stabilator Failure} \\
 R_{rr} \geq R_{rr_FL} \Rightarrow \text{Rudder Failure}
 \end{aligned} \tag{24}$$

In case of isolation, a time interval up to Δt_2 should be allowed for ω_{pq} to grow

3.2.8. Failure Identification for Sensor Isolated

On the advent of sensor failures MNN starts learning biased angular rates as it have the sensor values as inputs while DNN still calculates angular rates based on the flight parameters except the sensor values. The difference of actual angular rates and the respective rates from the DNNs trigger failure flags and are identified by monitoring the outputs of DNNs. In an advent of failure the *DQEEs* cross their respective FL bounds and issues an identification flag which are given by eq. (25).

$$\begin{aligned}
 DQEE_p \geq DQEE_{p_FL} \quad \Rightarrow \text{Roll Sensor Failure} \\
 DQEE_q \geq DQEE_{q_FL} \quad \Rightarrow \text{Pitch Sensor Failure} \\
 DQEE_r \geq DQEE_{r_FL} \quad \Rightarrow \text{Yaw Sensor Failure}
 \end{aligned} \tag{25}$$

The logical diagram of the FDII scheme for an actuator and sensor failure with above discussed criteria is shown in Figure 3-6.

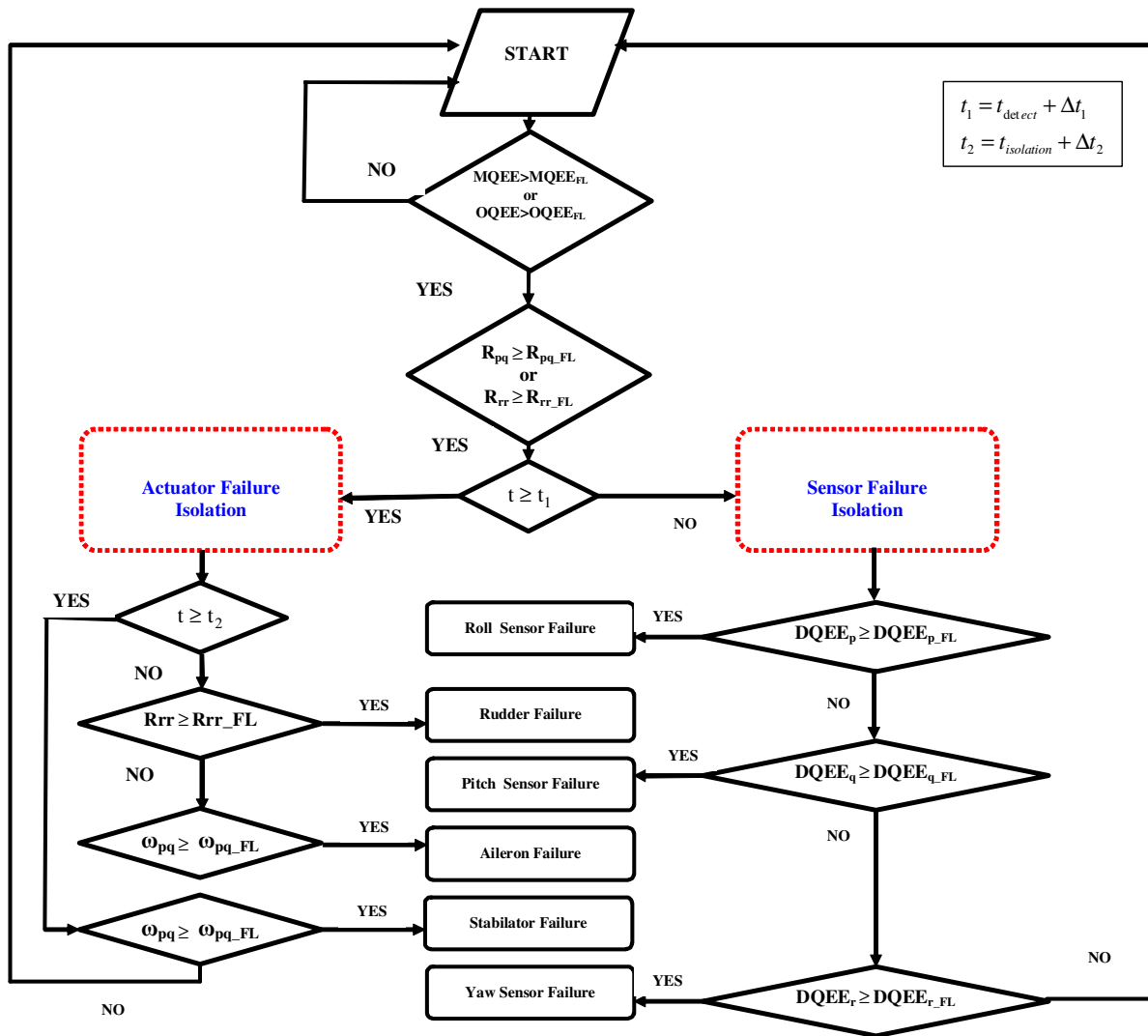


Figure 3-6 Logical Diagram for FDII

The FL parameters are tuned by changing the bound factor and bias by running the failure simulations till the number of false alarms is reduced. The tuned FL parameters are shown in Table 3-1. These were obtained by running the simulation for a number of times for different failures till the number of false detections was reduced to minimum.

Floating Limit Parameter $X_{\text{Bound}} = \bar{X} + \beta \times \sigma(X) + b$	β_{SB}	b_{SB}	β_{HB}	b_{HB}
$MQEE_{floating}$	1.4	0.00001	3.0	0.00001
$OQEE_{floating}$	1.2	0.00005	6.0	0.00005
$R_{pq_floating}$	1.5	0.001	6.0	0.001
$R_{rr_floating}$	1.5	0.001	3.0	0.001
$\omega_{pq_floating}$	1.2	0.003	3.0	0.003
$DQEE_{p_floating}$	1.5	0.002	3.0	0.002
$DQEE_{q_floating}$	2.0	0.00001	6.0	0.00001
$DQEE_{r_floating}$	1.5	0.00001	3.0	0.00001

Table 3-1 Floating Limiter Tuning Parameters for Integrated FDII Scheme

3.3. Integrated Sensor/Actuator S/AFDII Scheme with FL

The Simulink scheme shown in Fig. 3-7 is developed and the FDII block is incorporated in *Gen-2* controller scheme and it issues failure warnings using the GUI. The inputs to the FDII scheme is the sensors values and they are processed separately by the MNNs and DNNs to monitor the correlations coefficients such as $MQEE, DQEE, R_{pq}, \omega_{pq}$. The FDII makes use of the logic shown in Figure 3-6 to detect, isolate and identify the failure. The output of the FDII block is also channeled to the X-plane visuals to send the same warning into the cockpit to warn the pilot about the failure.

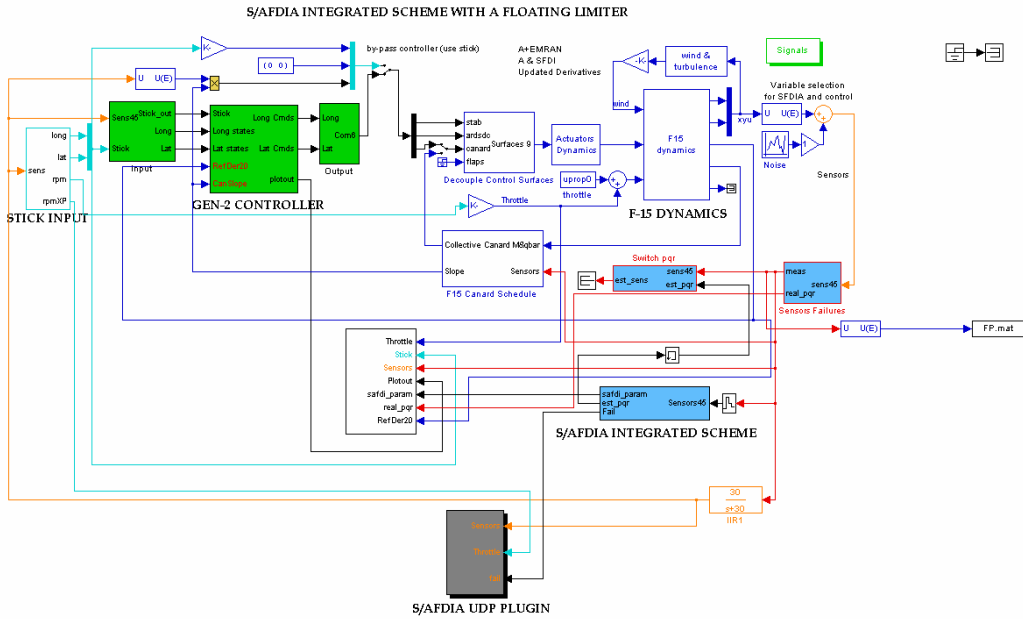


Fig. 3-7-Integrated Simulink scheme for S/AFDIA with Floating Limiter and MFS Interface

The detailed FDII block is shown in Fig. 3-8 and the outputs from MNNs and DNNs are processed to detect, isolate and identify the failure. The FDII block consists of a display warning system for the failure.

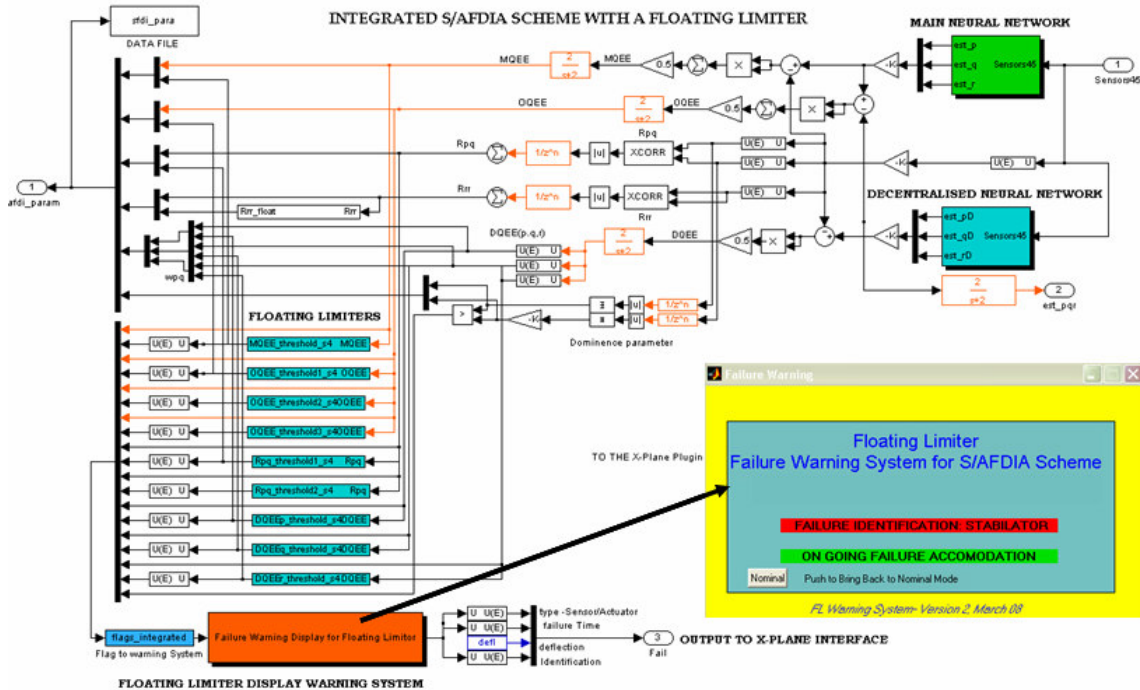


Fig. 3-8-S/AFDIA Scheme incorporating the Floating Limiters

The typical failure warning window in the Simulink scheme is shown in Figure 3-9. It pops up when the failure is initially detected. Similarly Figure 3-10 is a snapshot of GUI pop up when the actuator failure is identified. At the same time the outputs of the failure warning block are sent to the UDP block to display a text message warning the pilot about the failure once the detection takes place within the Simulink scheme. A typical warning received in cockpit is shown in Figure 3-11.

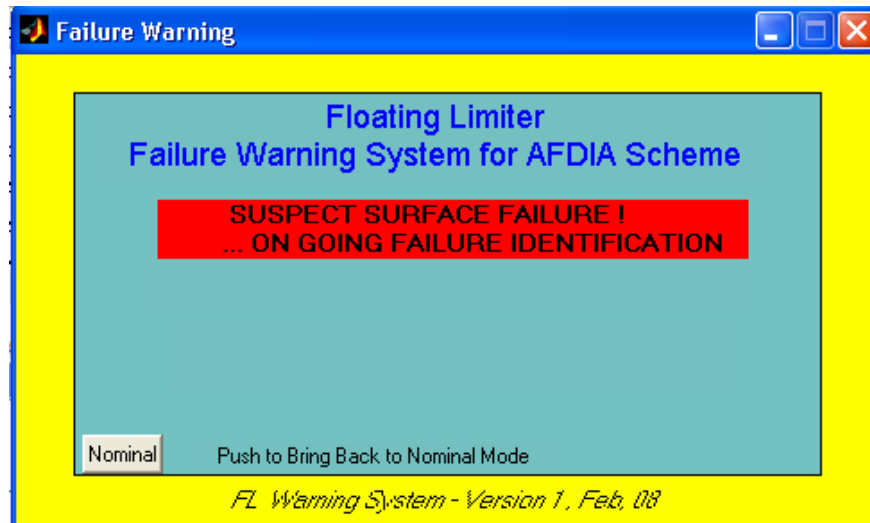


Figure 3-9 A warning window with Failure Detection

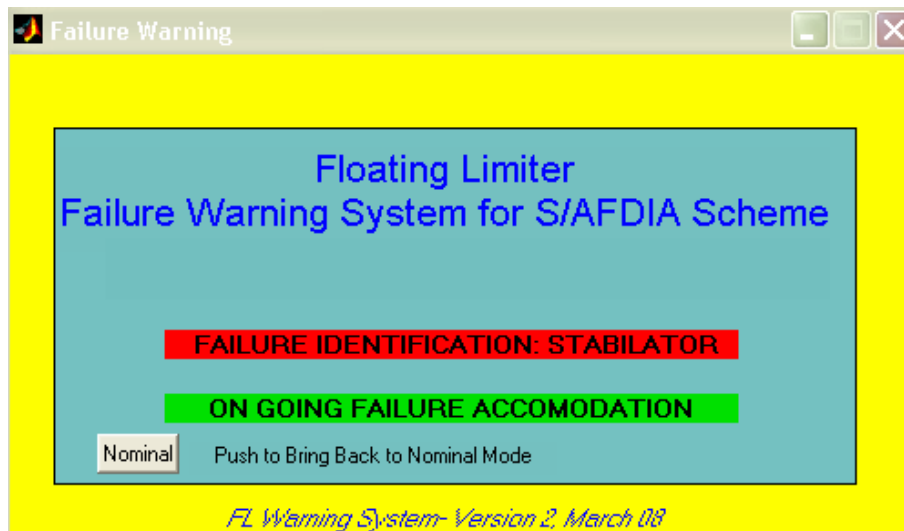


Figure 3-10 GUI for Actuator Failure Identification



Figure 3-11 Typical Warning received at lower left corner of pilot facing visual display

These warnings in the cockpit provide the pilot a notification of the failure and augment his response in handling the failure. Thus in addition to the aural, visual and motion cues these help the pilot to make a best judgment in compensation.

Chapter 4. Experimental Procedures

4.1. Experimental Environment Setup

The experiments are set up by interfacing the Flight Simulation Research Computer (FSRC) with the MOTUS flight simulator (MFS). The F-15 *Gen-2* schemes run on the FSRC and it is interfaced with MFS. A conventional joystick shown in Fig. 4-1 is used for all piloted experiments and it is used as an input device to the Simulink scheme. The cockpit controls of the flight simulator are replaced by this joystick for sake of convenience and also because the joystick is more realistic for flying a fighter aircraft. The joystick is provisioned to be placed in the cockpit by making a fitment shown in Fig. 4-2. This fitment ensures that the joystick remains secure in the cockpit and eliminates any need for the pilot to hold it in hands or lap. Additionally, in order to monitor the variables that are sent to the motion base from Simulink scheme an external monitor is hooked to one of the VGA outputs of the simulator mainframe. It is shown in Figure 4-3.



Fig. 4-1- Joystick Used for Piloted Experiments



Fig. 4-2-Fitment Placed in Cockpit to Place the joystick



Figure 4-3 Externally Hooked Monitor to inspect the Motion Base Data

4.1.1. Description of MOTUS Flight Simulator

The MOTUS flight simulator is a state of the art flight simulator consisting of five computers communicating with each other and synchronizing information by means of a central computer called the server computer. X-plane is the flight simulation software running simultaneously on all five computers and the communication is through intranet among the computers. The sever computer controls the other 4 computers as well as the Motion Base Computer (MBC). The MBC converts the accelerations into the inputs for motion base actuator motors which provide the desired motion cues. It should be noted that the algorithm for MBC is proprietary software by Fidelity Flight Inc. and there is no direct method to change the software parameters. The following are the main systems of the Motus Simulator

Computer 1 – This is *visual display No. 1* which forms the side left view.

Computer 2 – This is *Visual Display No. 2* which forms the forward view.

Computer 3 - This is *Visual Display No. 3* which forms the side right view.

Computer 4- This is the *Server* and displays the cockpit view showing instrument Panels

Computer 5- This is *Instructor Console* and used for changing the flight Conditions and loading different aircrafts etc

Motion Base Control Box: This is used to bring the motion base to kneel or flying mode and is shown in Figure 4-4.



Figure 4-4- Motion Base Control Box

4.1.2. Description of X-plane Software

X-plane software⁶⁷ is the heart of the MOTUS visual and aural flight simulation environment. It is developed by Laminar Research and features good graphics, an external user interface and ability to design and custom the aircrafts and scenery. The software has features to input and output the data pertaining to the flight simulation and it features a Software Development Kit (SDK) that allows users to set up communication with X-plane software by means of User Datagram Protocol (UDP). The UDP data can be sent /received by X-plane by means of writing external plugins which are essentially Dynamically Linked Libraries (DLLs). All plugins are managed by X-plane Plugin Manager (XPLM) which controls the data handling within the software. This feature is effectively utilized for this research effort and the Simulink based schemes are modified to communicate with X-plane and Motion Base Computer (MBC) in the MOTUS flight simulator by writing suitable plugins.

4.2. Interface of Simulink Scheme with Motion Simulator

The prime objective of the Simulink[®] interface with MFS is to create an efficient test bed to simulate advanced F-15 *Gen-2* schemes. The basic idea behind using the simulator was to obtain motion and visual cues for these Simulink schemes and create an advanced simulation environment than desktop simulation. The schematic of the interface is shown in Figure 4-5 and it can be seen that the joystick inclusion in the cockpit creates a feedback loop. The instructor console set-up is shown in Figure 4-6 which shows the research computer and the The foremost part of the interface was to identify the parameters necessary to obtain the necessary visual and motion cues for the Simulink[®] scheme. The parameters required to produce the desired visual cue were identified based on understanding the X-plane simulation set-up. The X-plane loads the scenery based on latitude and longitude information on a spherical earth. The aircraft is placed on the earth based by specifying the Euler angles and altitude and therefore the visual cues are obtained from the X-plane based on the location of the aircraft in terms of altitude, latitude and longitude and it defines the position of the aircraft on the globe. The deflections of the primary control surfaces need to be specified and they formed other important parameters in the visual cues. Thirdly the spatial orientations i.e. Euler angles are transformed to Quaternion in the X-plane. The parameters that are used to obtain the visual cues from X-plane and motion cues from MBC are shown in Table 4-1 and Table 4-2.

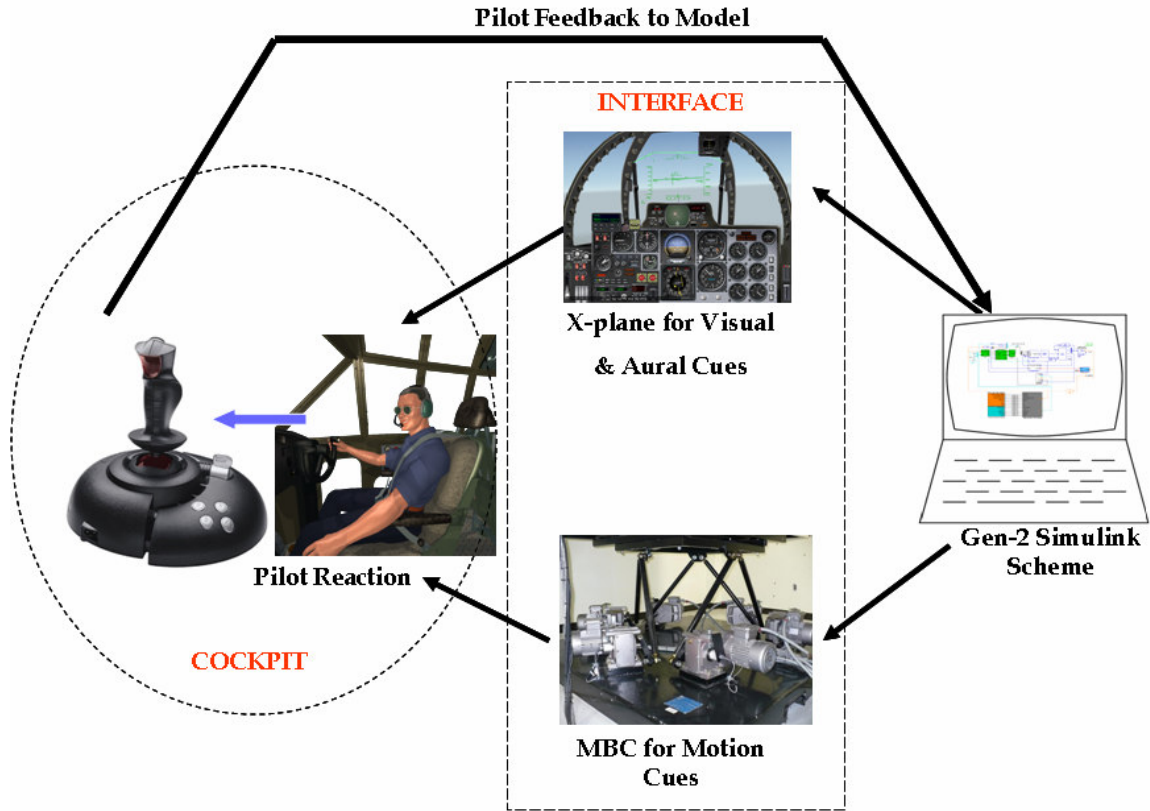


Figure 4-5 A schematic of the MFS interface with the Gen-2 Simulink Scheme for experiments



Figure 4-6 The instructor console station at MFS

Variable	X-Plane DataRef
Lat, Long, Elev.	sim/flightmodel/position/q
V_x	Sim/flightmodel/position/local_vx
V_y	sim/flightmodel/position/local_vy
V_z	Sim/flightmodel/position/local_vz
Roll rate	sim/flightmodel/position/P
Pitch rate	sim/flightmodel/position/Q
Yaw rate	sim/flightmodel/position/R
Throttle	Sim/flightmodel/engine/ENGN_thro_override
Aileron def	Sim/joystick/artstab_roll_ratio
Elevator def	Sim/joystick/artstab_pitch_ratio
Rudder def	sim/joystick/artstab_heading_ratio

Table 4-1 : Data Sent to X-Plane for Visual Cue

Variable	X-Plane DataRef
Total Airspeed	Sim/flightmodel//position/groundspeed
Angle of Attack	sim/flightmodel//position/alpha
Pitch Acceleration	sim/flightmodel//position/Q_dot
Yaw Acceleration	sim/flightmodel//position/R_dot
Roll Acceleration	sim/flightmodel//position/P_dot
Side Acceleration	Sim/flightmodel//forces/g_side
Normal Acceleration	Sim/flightmodel//forces/g_nrml
Axial Acceleration	Sim/flightmodel//forces/g_axil
Pitch Angle	Sim/flightmodel//position/theta
Yaw Angle	Sim/flightmodel//position/beta
Roll Angle	Sim/flightmodel//position/phi

Table 4-2- Data Sent to Motion base for Motion Cue

4.2.1. Motion Computer Data and Displays

In order to check whether correct data is sent to the motion base, an external monitor is hooked to the MBC. During the flight experiments the data on the monitor can be checked with the ones in the Simulink schemes. The format of the data is specified by the

manufacturer and should strictly adhere to the format shown in Table 4-3. These indicate the order in which data is received and can be compared with the Simulink data.

Message number 170:	
Item 0 : Motion override flag (should be set to 1)	
Item 1 : Total airspeed (in knots)	
Item 2 : Angle of attach (Degrees)	+Up
Item 3 : Pitch Acceleration (Deg/sec ²)	+Up
Item 4 : Yaw acceleration (Deg/sec ²)	+Right
Item 5 : Roll acceleration (Deg/sec ²)	+Right
Item 6 : Side acceleration (Gs)	+Right
Item 7 : Normal acceleration (Gs)	+Up
Message number 171:	
Item 0 : Axial acceleration (Gs)	+Forward
Item 1 : Pitch angle (Degrees)	+Up
Item 2 : Yaw angle (Degrees)	+Right
Item 3 : Roll angle (Degrees)	+Right
Item 4 : not used	
Item 5 : not used	
Item 6 : not used	
Item 7 : not used	

Table 4-3- Motion Base Computer Data Format for Receiving via UDP

The motion data is sent through the Simulink scheme since the objective is to obtain the motion cues based on the *Gen-2* control algorithms. The software designed by Fidelity Flight Inc within MFS for driving motion base automatically adjusts the acceleration received from the X-plane within the prescribed limits but it is not the case with us. Thus we have placed a hard limit on the accelerations limits as prescribed by the manufacturer as shown in Table 4-4. A couple of tests were carried out to ensure the accelerations experienced for the tests are within the limits. The accelerations can be monitored while running the experiments while they are sent to the motion base. A generic external monitor used and connected to VGA output of MBC. The data received is shown in a zoomed view in Figure 4-7.

Acceleration	Limits
Pitch Acceleration	$\pm 30 \text{ deg/ sec}^2$
Yaw Acceleration	$\pm 30 \text{ deg/ sec}^2$
Roll Acceleration	$\pm 30 \text{ deg/ sec}^2$
Side Acceleration	$\pm 3 g$
Normal Acceleration	$\pm 3 g$
Axial acceleration	$\pm 3 g$

Table 4-4- Limits on acceleration for Motion base



Figure 4-7 The external monitor showing the Motion Base data

This interface is divided into two main blocks.

1. X-plane visual UDP interface
2. Motion Base UDP interface

The 2 interfaces are s-functions custom built for obtaining the visual and motion cues for the Simulink model.

4.2.2. UDP interface and IP address settings for Communications

The main objective was to carry out simulation experiments for the neurally augmented failure detection schemes in the flight simulator. To accomplish this, two different S-function were written in the Simulink scheme to send UDP data to the X-plane and the motion computer. Figure 4-8 shows the Simulink scheme with the UDP interface block for this particular study. Figure 4-9 shows the UDP interface block in detail. The visual UDP block asks for user to input the IP address of the server computer in order to make a communication with X-plane. When the user clicks on the visual block, a window shown in Figure 4-10 prompts the user to specify the IP address. Similarly, a pop up block prompts user to input the IP address for Motion data interface and is shown in Figure 4-11.

In addition to these, two plugins for X-plane whose objective is to receive the data sent by Simulink scheme via s-function discussed before were created. These plugins communicate with the central plugin called X-plane Plugin Manager (XPLM). The X-plane software doesn't allow any communication directly and therefore all communications are channeled through the XPLM. The syntax and procedure for writing customized plugin for X-plane can be found in Software Development Kit (SDK) at the manufacturer website.

There are few things which must be taken into account while running the experiments using this interface. Foremost the Simulink scheme should be running before X-plane is started. This is necessary to open UDP ports for the X-plane plugins. Also, it clears previous buffer from the memory of X-plane. The X-plane dial for the indicated airspeed uses knots for display. We have adjusted it to display speed in m/sec for our convenience and for consistency with the Simulink model.

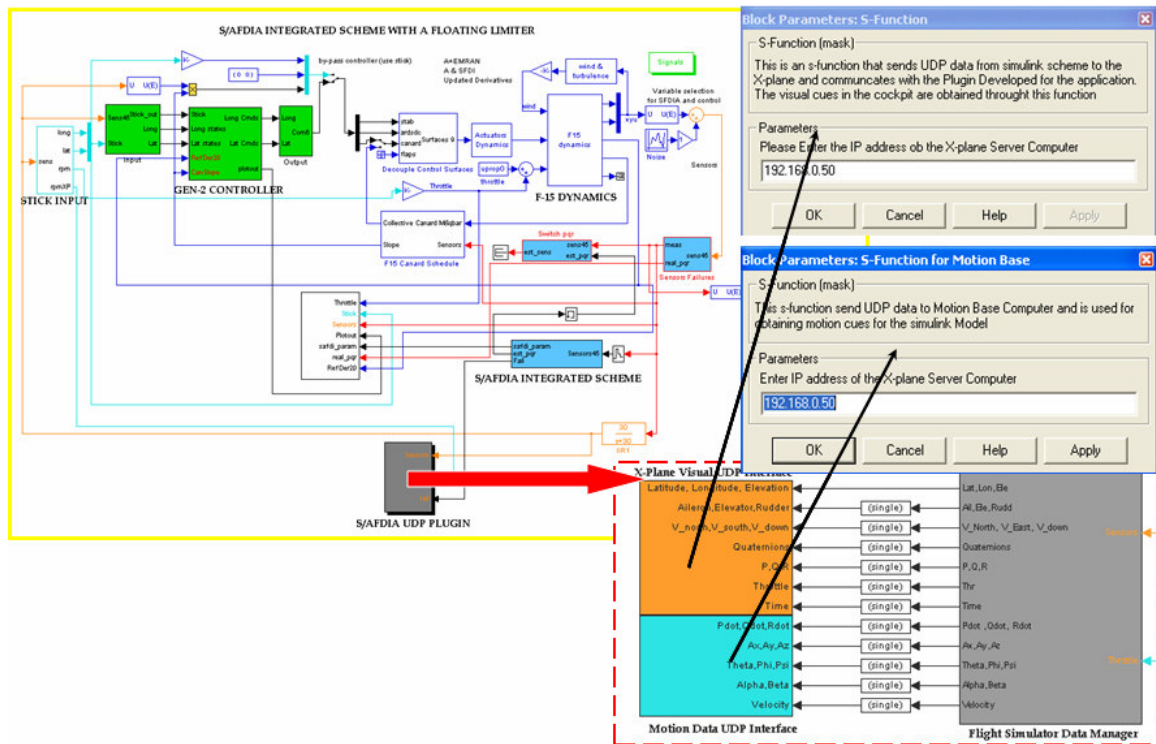


Figure 4-8- Simulink Scheme integrated with the UDP interface

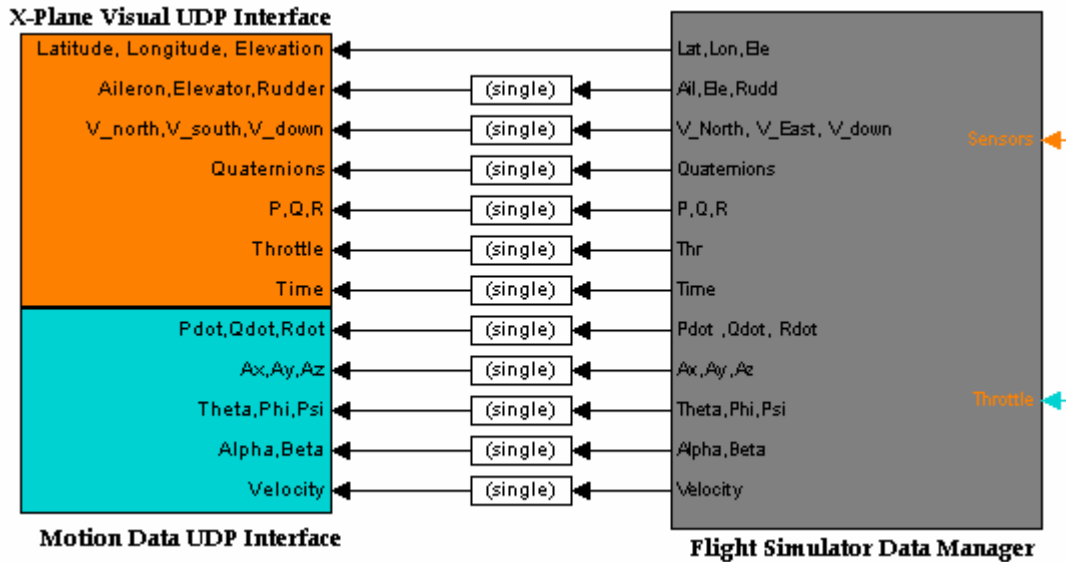


Figure 4-9- UDP interface Block in the F-15 Simulation Scheme

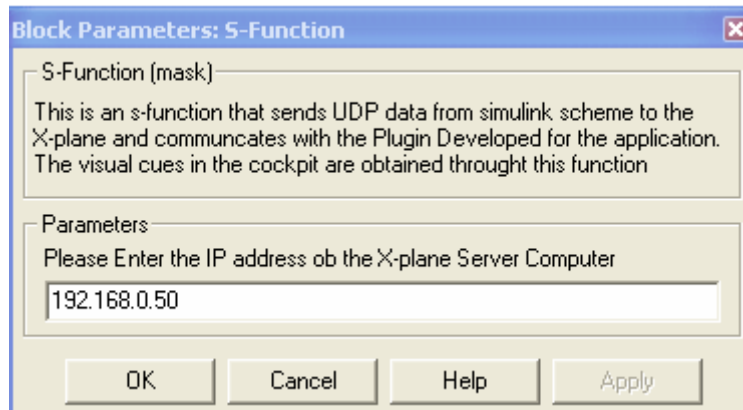


Figure 4-10- POP up window to specify IP address of the Motus Server machine for Visual UDP interface

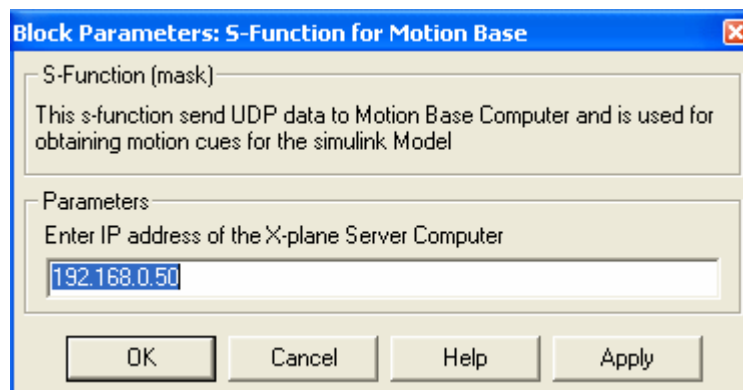


Figure 4-11 POP up window to specify IP address of the Motus Server machine for Motion UDP interface

As mentioned earlier the five computers interact with each other through Ethernet connections. The server houses the IP addresses of the other machines. The visual display

computers should have IP address of the server. The snapshot of the server computer should have for the IP address settings is shown in Figure 4-12. These should be strictly selected as shown in figure else it could desynchronize the visual and instructor console with the server computer.

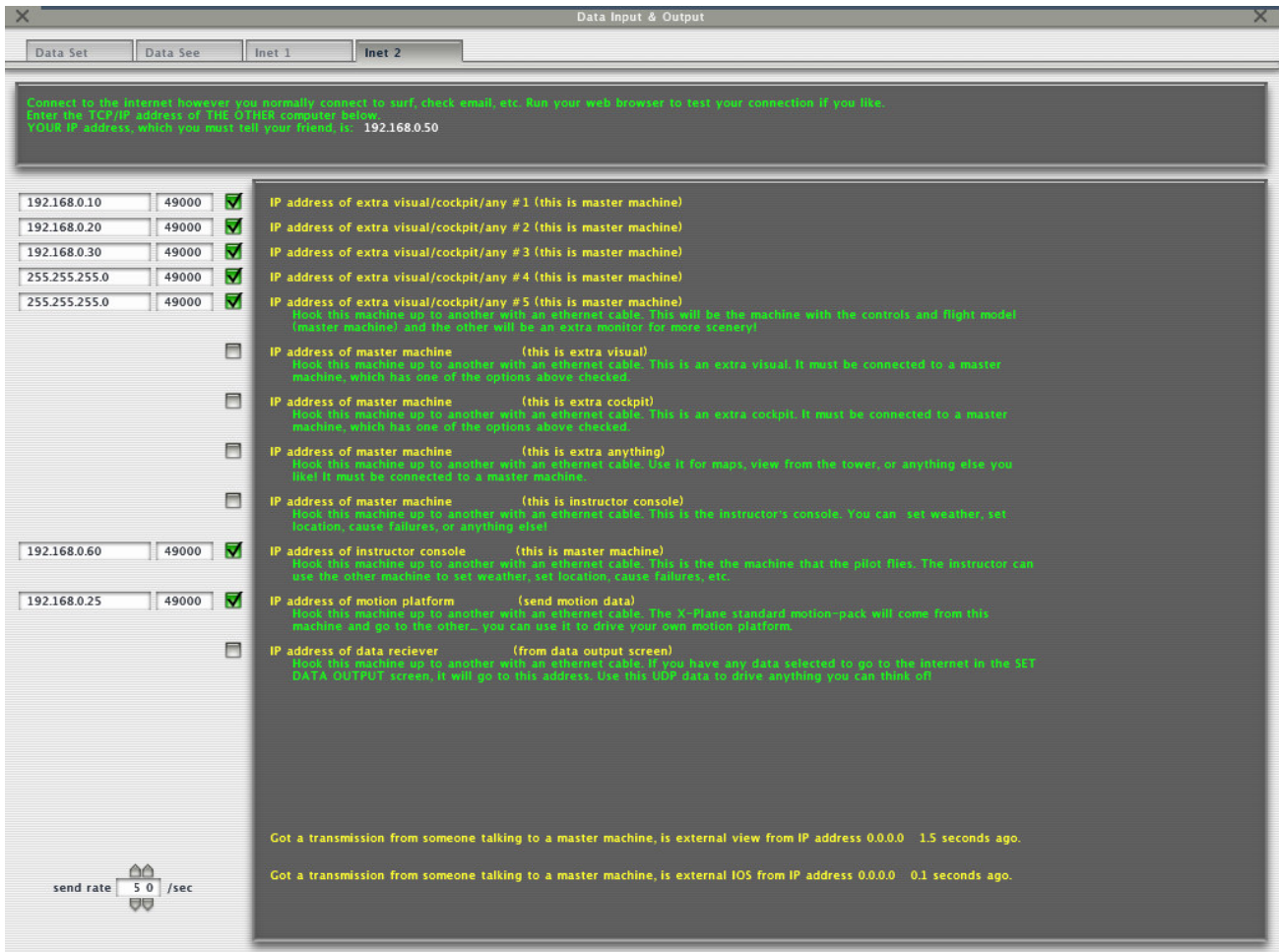


Figure 4-12 IP settings and UDP setting for Server Computer

4.2.3. Pilot interaction and Warning Systems in the Cockpit

The pilot is included in the loop by placing the joystick in the cockpit as shown in Figure 4-13 and the flight experiments are conducted in a controlled fashion and many a times it is necessary to warn the pilot about the type of work he is expected to do. It can be anything from notifying in advance about a maneuver about to hit or to instruct him to leave the stick or fly with the stick. An example of such a warning is shown in Figure 4-14



Figure 4-13 Joystick Placed in the Cockpit



Figure 4-14-Pilot warning about an oncoming maneuver

Since the objective is pilot in the loop assessment therefore a warning system is also developed to display failure messages in the cockpit. As soon as the simulink FDII block

detects and identifies a failure it is communicated to the pilot by way of a text warning as shown in Figure 4-15 on the bottom left corner. These warning can be enabled/disabled by using F1/F2 keys from the server computer. The instructor console computer is used to visually observe the flight experiments and Figure 4-16 shows a chase view of the simulation.



Figure 4-15- A snapshot of warning message received in the cockpit



Figure 4-16- Instructor Console Snapshot for monitoring the flight experiments

4.3. Description of Simulink software

A desktop based simulation environment⁴⁶ is developed in Simulink for the F-15 *Gen-2* control scheme. The model was originally derived from a high performance fighter aircraft non-linear code distributed at the 1990 AIAA GNC Design Challenge⁴⁷ by NASA. The model efficiently embedded in the Simulink environment and the Graphic User Interface (GUI) to create a user friendly simulation environment. Two types of

control surface failure have been modeled. The first failure type corresponds to an actuator mechanism failure and results in a locked surface; in fact, at the failure occurrence, the control surface remains fixed in the current or in a user prescribed position. Any of the individual six control surfaces may be subject to a failure: left or right stabilator, aileron, and rudder. The second failure types correspond to induced drifting biases of the angular rate sensor output and are implemented as sensor failures⁵⁶. The user can select between different transients as well as different sizes of the bias for a total of six different sensor failures.

4.3.1. The Simulink graphic user interface

Figure 4-17 shows the typical interface of the Simulink Graphic User Interface (GUI) scheme for user friendly selection menu for choosing the flight scenarios. The left hand side of Figure 4-17 enables the user to choose the type of flight scenario. Once it is selected then it is required to choose the simulation scenario which has different options like nominal, actuator failure without FDI, actuator failure with FDI etc.

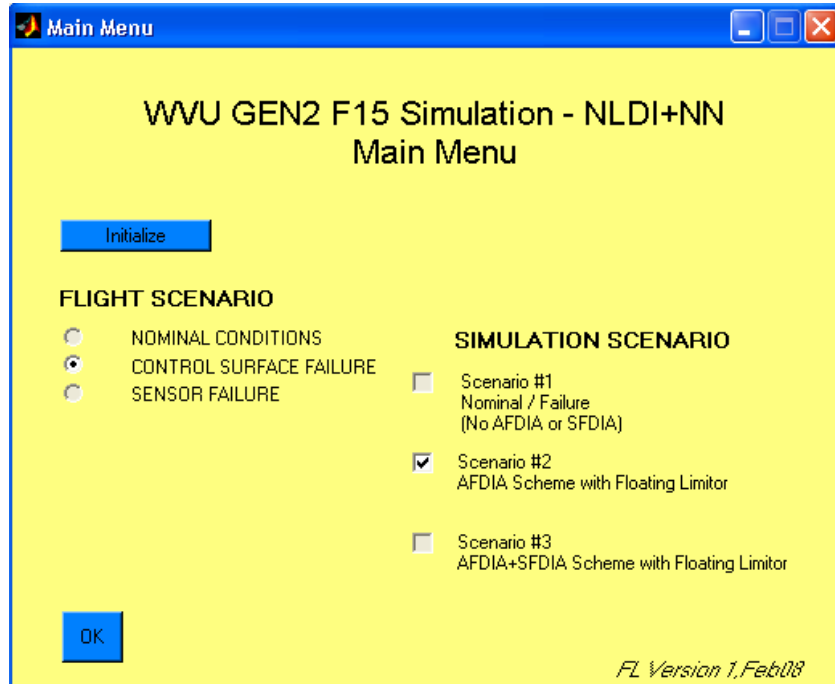


Figure 4-17- Main GUI window for Flight Scenario Selection

After selecting the desired simulation scenario the next window shown in Figure 4-18 gives user the option to select the mode of pilot input. The input can be all generated by

joystick, pre-recorded or a combination of both. If the pre-recorded input is chosen it must be included in the parent directory.

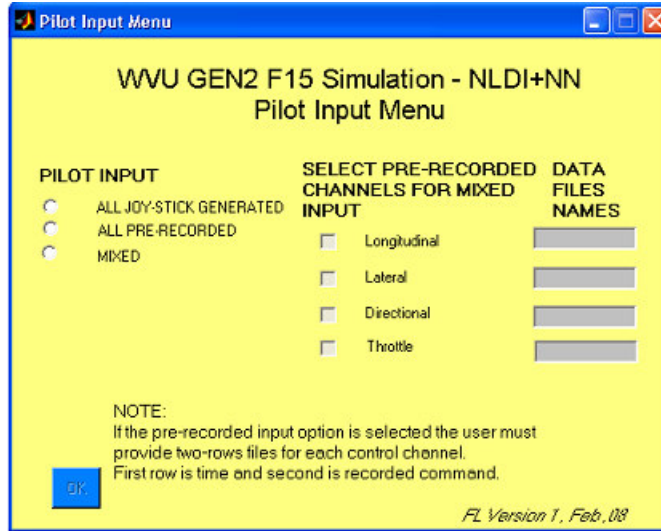


Figure 4-18- Input command selection window

With the selection of inputs the 3rd GUI window shown in Figure 4-19 pops up which prompt the user to select the type of NN for the scheme. Figure 4-20 shows the GUI pop up window to define the actuator failure parameters. The user has option to introduce a particular failure at a particular time instant and deflection. Similarly Figure 4-21 shows the selection of sensor failure scenarios.

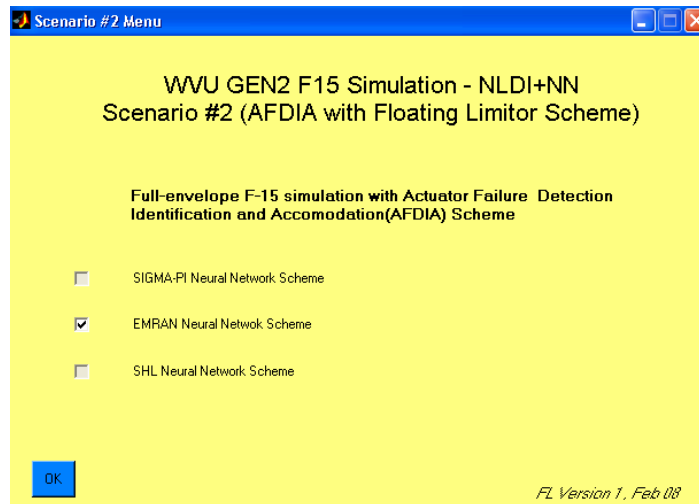


Figure 4-19-Selection of Neural Network Scheme

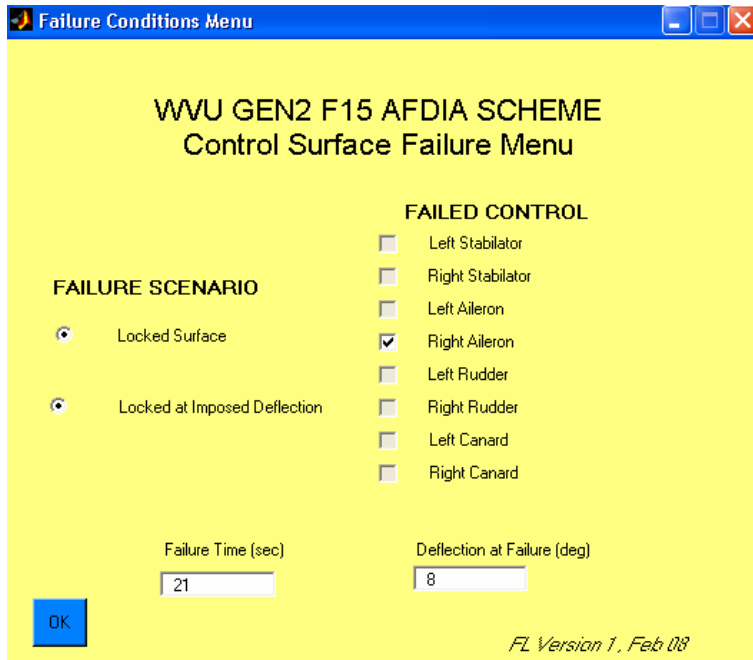


Figure 4-20- Defining the actuator failure parameters

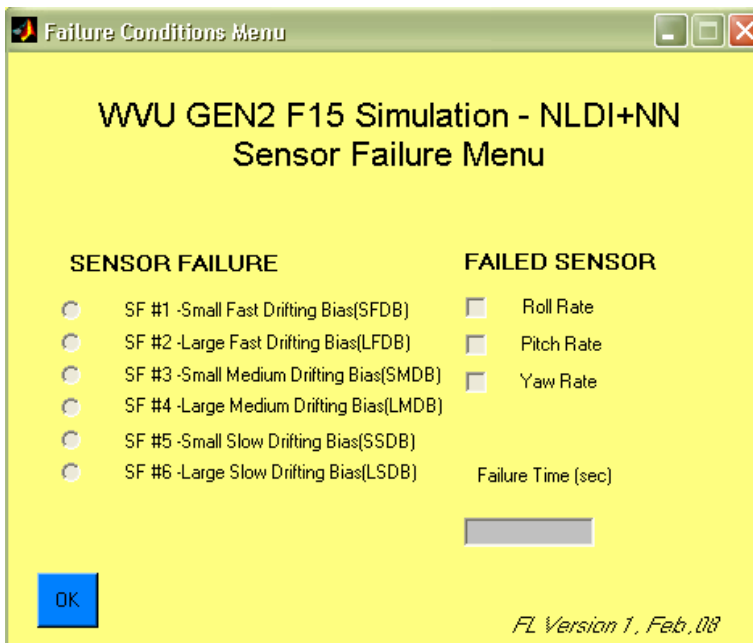


Figure 4-21-Defining the Sensor Failure Parameters

Chapter 5. Results and Evaluation

5.1. Overview of the experiments

The goal of this research effort is the evaluation of the pilot-in-loop performance of the fault tolerant flight control schemes under different flight conditions (nominal/failure) as well as for different class of failures (actuator/sensor). There were primarily two broad objectives of the study. Firstly, to compare the performance of three NN augmented schemes and secondly to evaluate the adaptive FDII schemes with a pilot in the loop in the flight simulator. The different phases of the experiments can be divided into the following scenarios:

1. Nominal Condition
2. Failure Conditions with no FDII
3. Failure Condition with Adaptive FDII scheme

5.2. Comparison of different NN schemes with Actuator failure (NO FDII)

To begin with baseline experiments were first conducted for nominal flight conditions. Pre-generated doublet maneuver along the longitudinal and lateral channel as shown in Fig. 5-1 is used as a baseline test estimate the performance of the NN schemes. At this point there is no pilot in the loop and the test served to validate the prior established performance.

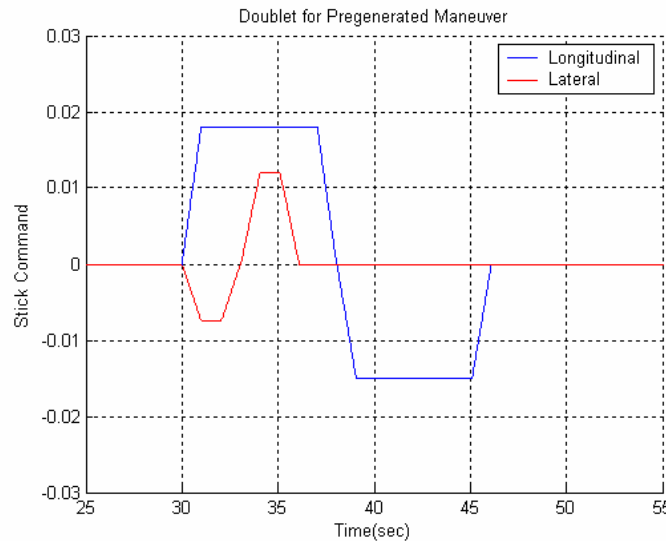


Fig. 5-1-Pre-generated Doublet for longitudinal and lateral channel

5.2.1. Tracking Error Performance for Nominal Condition (Baseline Doublets)

The TE statistics for comparing the NN schemes for the longitudinal doublet maneuver shown in Table 5-1

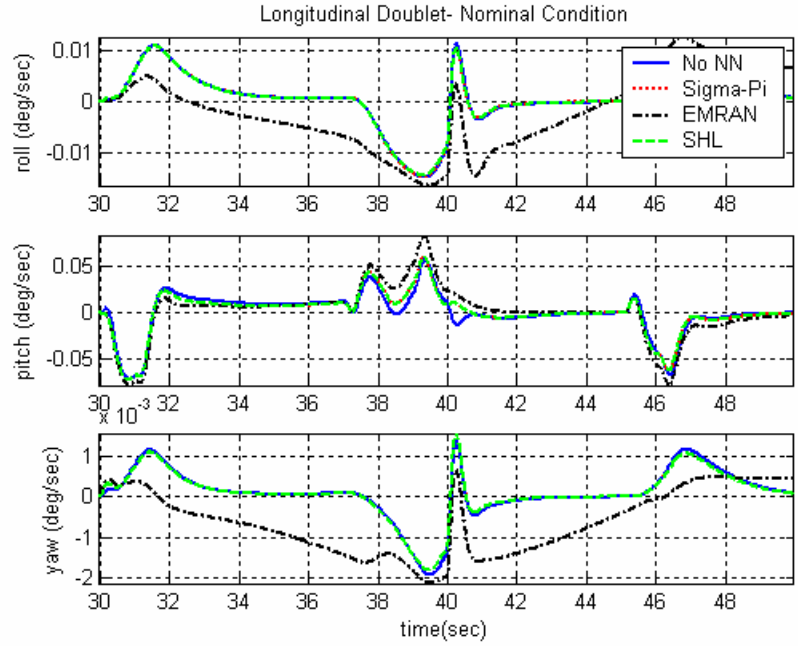


Figure 5-2-Tracking Error for Longitudinal Doublet

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e-4	4.7978	4.8227	-1.4091	4.7296
Mean Pitch x10e-4	-1.1241	-1.4045	1.1829	-1.2762
Mean Yaw x10e-5	6.7871	6.7671	-5.2985	6.7599
SD Roll x10e-3	4.5470	4.5443	4.7688	4.5454
SD Pitch x10e-2	2.8178	2.8278	3.1351	2.8124
SD Yaw x10e-4	4.1754	4.1672	4.9050	4.1739

Table 5-1 Longitudinal Doublet TE – (Nominal Condition)

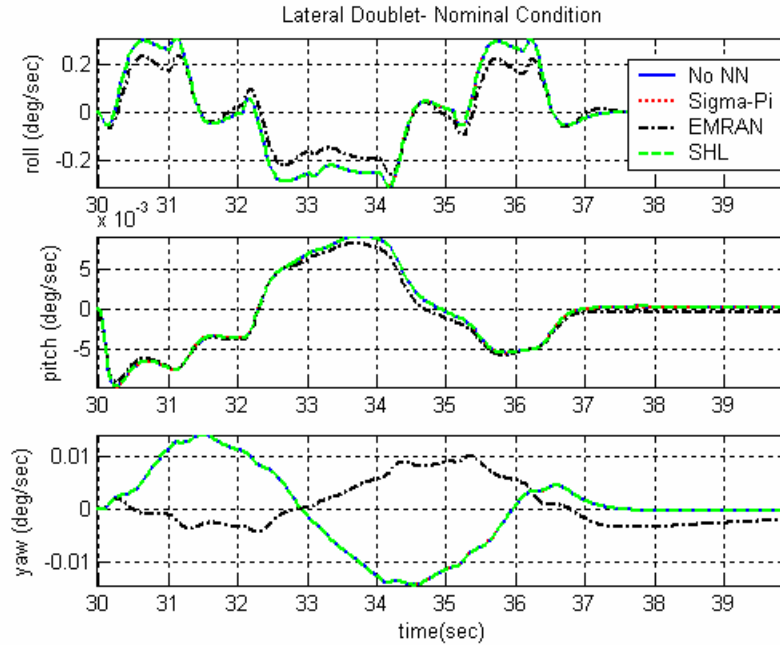


Figure 5-3- Tracking Error for Lateral Doublet

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e-4	1.5575	1.5916	-0.25313	1.5552
Mean Pitch x10e-3	-2.8190	-2.8747	-1.7086	-2.8110
Mean Yaw x10e-3	-1.2636	-1.2633	-0.3650	-1.2650
SD Roll x10e-1	2.3504	2.3541	1.7207	2.3487
SD Pitch x10e-2	1.5643	1.5895	1.4898	1.5623
SD Yaw x10e-2	1.0075	1.0098	0.35934	1.0094

Table 5-2 Latéral Doublet TE Statistics (Nominal Condition)

The tracking error (TE), defined as the difference between the rates generated by the reference model and the actual aircraft rates is an important indicator of neural networks performance. The TE along the pitch channel for a longitudinal maneuver and the pitch and roll channels for a lateral maneuver are considered for analysis in the following sections. For the purposes of this study, the TE statistics on each channel is computed over a 20 sec time window around the pre-generated longitudinal maneuvers and 10 second window for the lateral maneuver, starting from the instant the maneuver occurs. Figs. 7-8 show the comparison of the TE for the pre-generated maneuvers along the longitudinal and lateral channels, with and without NN compensation and the corresponding statistics are tabulated in Table 5-1 and Table 5-2. It can be seen that the

effect of NN augmentation is negligible, and may be attributed to small inversion errors and periodic updating of aerodynamic coefficients.

5.2.2. Analysis of Actuator Failures (No pilot- No FDII)

Following the nominal condition experiments with pre-generated maneuvers, failures on the left stabilator and right aileron, as shown in Table 5-3 and Table 5-4, were simulated and the performance of the control laws evaluated (no pilot in loop). For failure cases, the TE statistics on each channel is computed over a 20 sec time window around the failure instant, beginning 5 seconds before the failure.

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e0	-0.6215	-0.4274	-0.2719	-0.6217
Mean Pitch x10e0	0.2409	0.1865	0.1155	0.2410
Mean Yaw x10e-2	-5.1404	-5.0736	-5.7683	-5.1420
SD Roll x10e0	2.4756	2.3646	2.5454	2.4756
SD Pitch x10e0	1.0592	1.0462	1.0386	1.0592
SD Yaw x10e0	0.1997	0.2075	0.2211	0.1997

Table 5-3 -Stabilator Failure TE Statistics (No Pilot)

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e0	0.2420	0.2172	0.1420	0.2420
Mean Pitch x10e-4	-2.7736	-2.8866	-2.3183	-2.7616
Mean Yaw x10e-4	-3.6941	-4.0079	-1.2083	-3.6968
SD Roll x10e0	1.5420	1.5309	1.5969	1.5420
SD Pitch x10e-2	1.1089	1.1282	1.1239	1.1089
SD Yaw x10e-3	5.7585	5.7924	7.5842	5.7584

Table 5-4 Aileron Failure TE Statistics (No Pilot)

Table 5-3 shows the TE statistics for the stabilator failure scenario (no pilot in the loop). The EMRAN performed better in comparison to the other NNs, reducing the mean of pitch tracking error by nearly 50% as compared to a no NN case. Table 5-4 shows the TE statistics for the aileron failure case. As in the case of the stabilator failure, the EMRAN performs better, reducing the mean of roll TE by nearly 40% TE and mean of pitch TE by 15%. The analysis also shows that the neural augmentation using the SHL NN has an adverse effect on the TE performance.

5.2.3. Analysis of Actuator Failures (With Pilot- No FDII)

The same failure conditions were now simulated with 3 different pilots that were selected from among the students in the WVU flight simulation course. Prior to conducting the experiments, the pilots were briefed about the nature of the failures and also instructed to maintain wings level flight after the failure. A specific objective of this phase of the study was to compare the level of pilot activity/compensation in the absence and presence of NN augmentation and to analyze the associated TE statistics. With Pilot 1 in the loop, for the stabilator failure, it can be seen in Figure 5-4 that the EMRAN is the best performer, with the mean TE on the pitch and roll channels reduced as compared to the no NN case. The mean TE statistics is reduced by 50-55%, and is shown in Table 5-5.

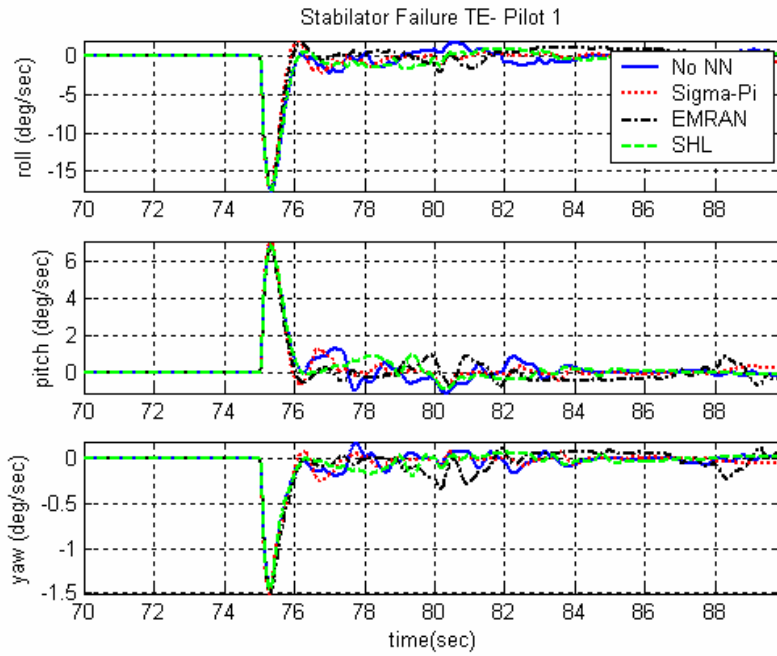


Figure 5-4 TE with different NN (Stab. Failure, Pilot1)

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e0	-.5928	-.5445	-.2469	-.6156
Mean Pitch x10e0	.2177	.2143	.1199	.2317
Mean Yaw x10e-2	4.7926	-5.0836	-5.6346	-4.9166
SD Roll x10e0	2.5489	2.3546	2.4624	2.5094
SD Pitch x10e0	1.0545	1.0244	1.0076	1.0295
SD Yaw x10e0	1.9724	2.0122	2.2005	1.9538

Table 5-5 Stabilator Failure TE Statistics (Pilot 1)

Similarly, for an aileron failure, EMRAN outperforms the other networks, effectively reducing the mean TE in the pitch channel by nearly 77% and roll channel by about 58%. The corresponding results are tabulated in Table 5-6 and shown in Figure 5-5. It can also be seen that the SHL network has negligible effect on the tracking error statistics in both the pitch and roll channels for both failure cases.

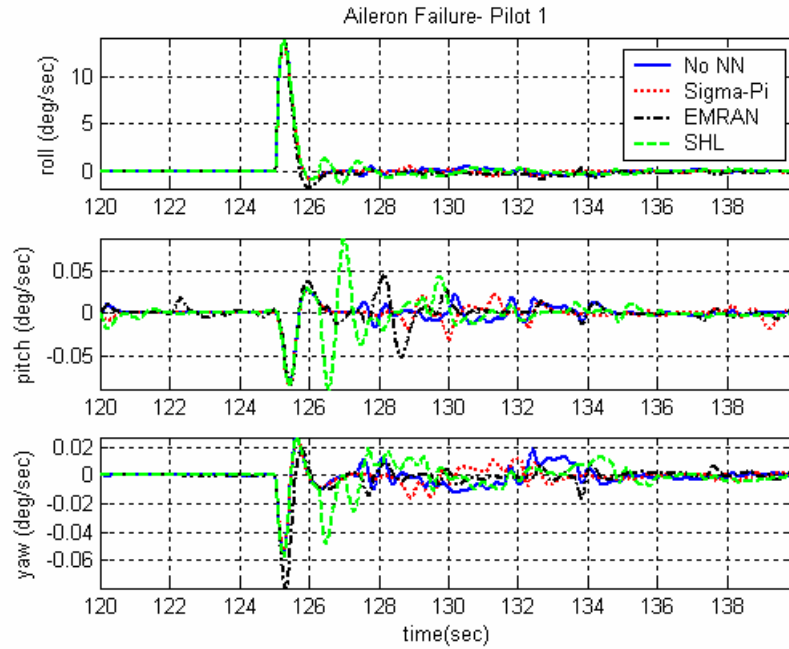


Figure 5-5- TE with different NN (Ail. Failure, Pilot1)

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e-1	2.8866	2.5512	1.2169*	2.8644
Mean Pitch x10e-4	-2.1266	-7.7528	1.3627*	-12.885
Mean Yaw x10e-3	-1.0142	-1.0213	-1.6379	-10.229
SD Roll x10e0	1.7463	1.7031	1.7454	1.7585
SD Pitch x10e-2	1.1651	1.2283	1.4529	1.8851
SD Yaw x10e-3	8.0870	7.4486	9.4841	10.103

Table 5-6 Aileron Failure TE Statistics (Pilot 1)

With Pilot 2 in the loop, for the stabilator failure, Sigma-Pi and EMRAN NNs are comparable in TE performance in the stabilator failure with a 40% decrease in pitch TE. On the other hand, the Sigma-Pi outperforms the EMRAN along the roll channel. The tracking error statistics are shown in

and Figure 5-6. The SHL NN actually degrades the performance in pitch TE, by 25%.

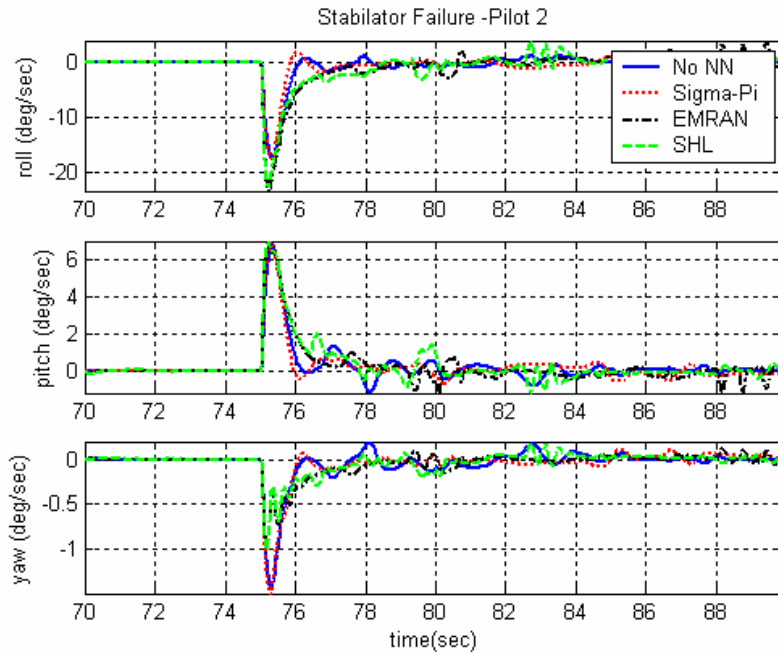


Figure 5-6- Stabilator Failure TE with different NN (Pilot 2)

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e0	-.5845	-.4564	-.9027	-.8390
Mean Pitch x10e0	.2196	.1914*	.1956	.2716
Mean Yaw x10e-2	-4.5483	-4.6421	-4.5490	-3.6780
SD Roll x10e0	2.5092	2.3910	3.3773	3.1885
SD Pitch x10e0	1.0231	.9995	1.1218	1.1479
SD Yaw x10e-1	1.9522	1.9547	1.3337	1.2997

Table 5-7 Stabilator Failure TE Statistics (Pilot 2)

On the other hand, for the aileron failure, the EMRAN performs the best with the mean of TE reduced considerably as compared with a no NN case. It shows a 65% reduction in mean TE along the roll channel and approximately 80% reduction in mean TE along pitch. The statistics are listed in Table 5-8 and the TE performance shown in Figure 5-7.

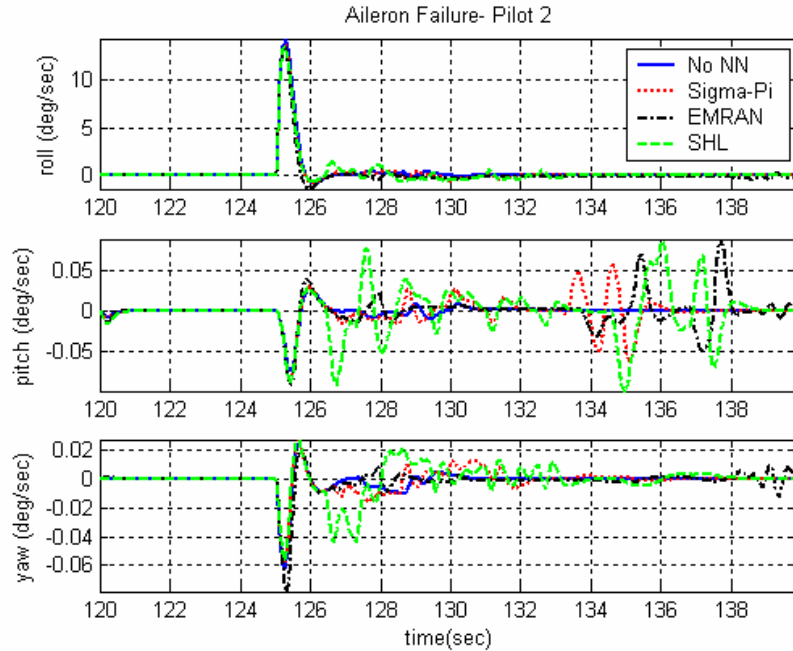


Figure 5-7 Aileron Failure TE with different NN (Pilot 2)

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e-1	3.0171	2.5672	1.0449	2.6877
Mean Pitch x10e-4	-7.5015	-7.9599	1.3001	-9.6403
Mean Yaw x10e-3	-116.20	-1.0067	-1.6330	-9.8127
SD Roll	1.8405	1.7336	1.7007	1.7359
SD Pitch x10e-3	9.7979	17.073	18.816	28.404
SD Yaw x10e-3	7.2739	7.8356	8.9207	1.1077

Table 5-8 Aileron Failure TE Statistics (Pilot 2)

With Pilot 3 in the loop, a similar trend is observed for the stabilator failure. The contributions from all the three networks result in a decrease in TE, with EMRAN reducing it by 58% in pitch and 82% in roll. The statistics are given in Table 5-9 and shown in Figure 5-8.

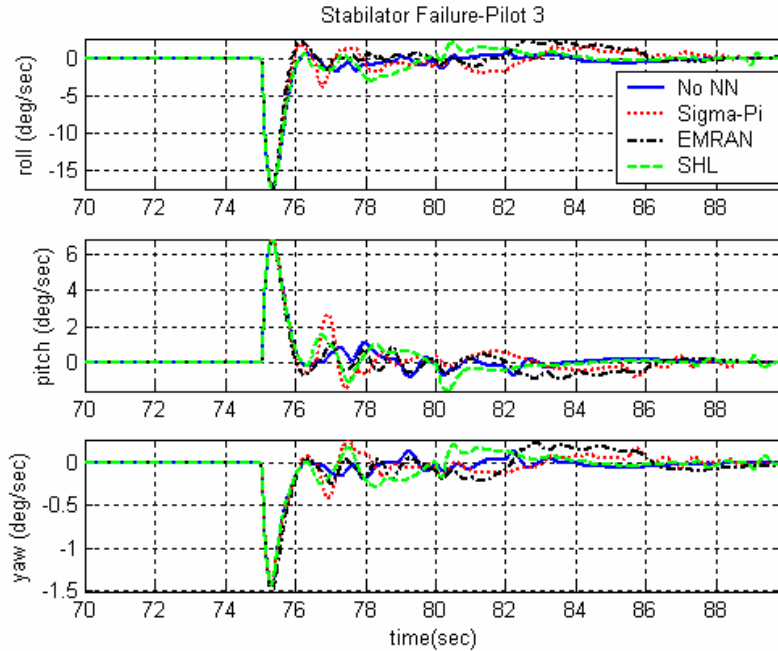


Figure 5-8- Stabilator Failure TE with different NN (Pilot 3)

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e0	-.6622	-.5030	-.1190	-.5570
Mean Pitch x10e0	.2422	.1983	.1001	.2114
Mean Yaw x10e-2	-5.2567	-4.4448	-4.9201	-4.4573
SD Roll x10e0	2.4889	2.5544	2.5513	2.6015
SD Pitch x10e0	1.0085	1.0678	1.0434	1.0599
SD Yaw x10e-1	1.9158	2.0732	2.3196	2.0742

Table 5-9 Stabilator Failure TE Statistics (Pilot 3)

In the case of aileron failure, as shown in Figure 5-9, the TE performance is degraded for all the networks on the pitch channel. However, in the roll channel, the Sigma-Pi network reduces the mean TE by 6%, while the EMRAN significantly degrades the performance. The results are tabulated in Table 5-10.

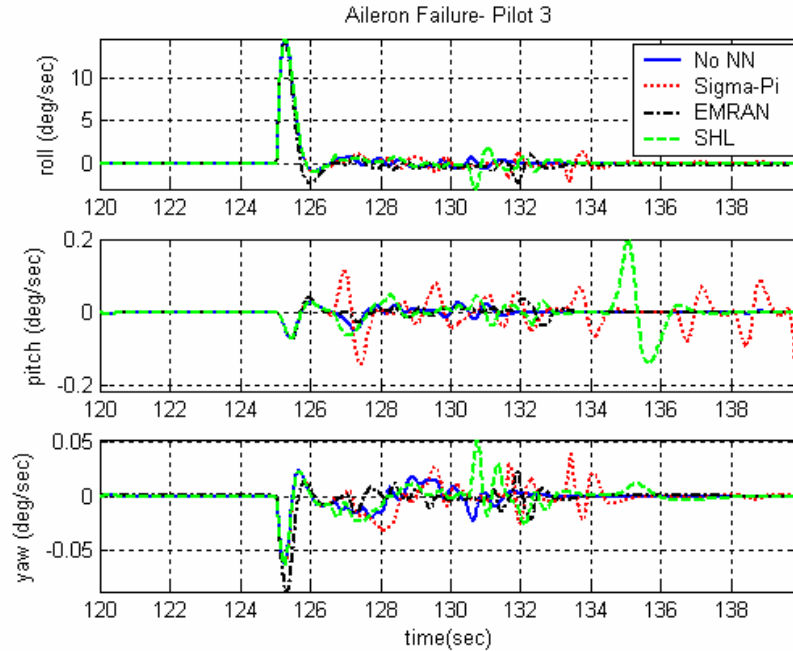


Figure 5-9-Aileron Failure TE with different NN (Pilot 3)

TE	No NN	Sigma-Pi	EMRAN	SHL
Mean Roll x10e0	.3079	.2883	.8474	.3101
Mean Pitch x10e0	-.66960	-2.3203	-1.6484	-5.2381
Mean Yaw x10e-2	-1.2423	-1.4342	-2.0495	-1.1713
SD Roll x10e0	1.8760	1.9125	1.8644	1.9196
SD Pitch x10e0	1.2384	3.8596	1.2828	3.6064
SD Yaw x10e-1	0.90192	1.1441	1.1084	1.1101

Table 5-10- Aileron Failure TE Statistics (Pilot 3)

With the conclusion of preliminary actuator failure tests in the simulator it was established that the MOTUS simulator served as a good test best for carrying out analysis of NN augmented control schemes. It extends the simulation capability to a next level of realism than the desktop simulation. Also, the motion and visual cues facilitate appropriate compensation in the advent of failure.

The next step was to develop an adaptive failure detection scheme which is capable of detection and distinguishing an actuator failure from a sensor failure. The earlier work done on the integrated S/AFDIA scheme with constant thresholds is found to be susceptible to disturbances and the threshold is crossed from disturbances arising out of

stick input from pilot as well. The adaptive threshold scheme is extended for an integrated (sensor and actuator) FDII scheme and one broad objective of carrying out these piloted experiments was to evaluate the robustness of the adaptive FDII scheme which has substantial pilot activity. Normally in a desktop simulation there are fewer amounts of false alarms because of no pilot in the loop and as a result the scheme looks robust. In the advent of a pilot in the loop cross-correlations are induced even when there is no failure. Moreover, as soon as a failure is detected the pilot reacts and may result in perturbations of monitored signals and might lead to false isolations and detections. In a desktop simulation there is no such response and the scheme gives good results. In case of real flight this is not the case so the FDII scheme should be tolerant for pilot activity arising due to sudden failure and identify the failure correctly between the time of detection and identification. Moreover wrong identification may lead to a wrong reconfiguration of the controller and might perplex the pilot further.

5.3. Evaluation of Integrated FDII Scheme with adaptive FL threshold

As discussed in chapter 3, the FDII scheme developed for actuator and sensor failures were subjected to piloted experiments in the MFS. The pilot was instructed to maintain wings level flight and maintain an altitude within 200 ft and the failure was induced at 30 sec from the start of simulation. Also the failures were thrown randomly so that pilot responds according to his best judgment without having a prior indication of the type of failure. The results are divided into two sections covering actuator and sensor failure cases. The FDII scheme result summary is expressed in terms of the detection delay, isolation delay, identification delay, correct isolation, correct identification and false alarms. The actuator failures are categorized as soft (2 deg), medium (5 deg) and hard (8 deg). The sensor failure have been formulated by having a small (4 deg/sec) and large (8 deg/sec) sensor bias. Also each of these biases is subject to a slow drift (2 sec), medium drift (6 sec) and fast drifts (10 sec). These six different cases of sensor failures shown in Figure 5-10 are used for the simulator tests. In order to reduce the number of experiments a test was carried out with an experienced pilot to choose the best NN for upcoming studies. An initial comparison of the Sigma-Pi and EMRAN was done based on simulating a hard stabilator failure (8 Deg) to choose among the two networks for the

next phase of the study. This was necessary to decrease the number of simulation tests. SHL was not used as its performance was poor in most of the earlier cases.

Run-1	No NN	Sigma-Pi	EMRAN	Sigma-Pi(% Inc/Dec)	EMRAN(% Inc/Dec)
Mean Roll	-4.00E-01	-3.26E-01	-3.83E-02	18.49	90.41
Mean Pitch	1.55E-01	1.29E-01	7.59E-02	16.54	51.01
Mean Yaw	-3.11E-02	-2.93E-02	-3.09E-02	5.56	0.53
SD Roll	2.29E+00	2.22E+00	1.46E+00	3.27	36.39
SD Pitch	9.04E-01	8.91E-01	7.83E-01	1.38	13.39
SD Yaw	1.82E-01	1.74E-01	1.44E-01	4.17	20.94

Run-2	No NN	Sigma-Pi	EMRAN	Sigma-Pi(% Inc/Dec)	EMRAN(% Inc/Dec)
Mean Roll	-4.29E-01	-6.17E-02	-3.03E-02	85.62	92.95
Mean Pitch	1.61E-01	9.35E-02	8.51E-02	42.06	47.31
Mean Yaw	-3.38E-02	-3.39E-02	-3.12E-02	-0.45	7.68
SD Roll	2.23E+00	1.96E+00	2.00E+00	12.37	10.29
SD Pitch	8.96E-01	9.56E-01	9.83E-01	-6.67	-9.62
SD Yaw	1.77E-01	1.78E-01	1.92E-01	-0.84	-8.64

Run-3	No NN	Sigma-Pi	EMRAN	Sigma-Pi(% Inc/Dec)	EMRAN(% Inc/Dec)
Mean Roll	-3.64E-01	-3.50E-01	-3.50E-02	3.69	90.37
Mean Pitch	1.38E-01	1.37E-01	8.10E-02	0.55	41.15
Mean Yaw	-2.80E-02	-2.99E-02	-2.97E-02	-6.89	-6.29
SD Roll	2.22E+00	2.17E+00	1.94E+00	2.27	13.02
SD Pitch	8.95E-01	9.12E-01	9.40E-01	-1.86	-5.04
SD Yaw	1.77E-01	1.77E-01	1.69E-01	0.24	4.59

Table 5-11 Comparison done for finalizing NN for further studies

Based on these tests it can be seen that EMRAN has more improvement when compared with a no NN case and hence it was frozen to be used for next set of tests on the actuator failures. The sensors failures will be carried out for the EMRAN NN alone as the objective is to test the FDII scheme. The analysis of failure cases are done in terms of TE along the roll, pitch and yaw channels. It is an indication of controller to track desired inputs. The other parameter for analysis is the pilot compensation (PC) which is the sum of absolute stick activity over the time under consideration. In this analysis of FDII algorithms we have considered a time window of 20 sec for monitoring the PC immediately after failure. Since it is desirable to have a TE and PC as little as possible so the Composite Parameter (CP) denoted by η , which is the absolute of the product of TE and PC is the third parameter for a meaningful observation and desirable to be minimum for best performance.

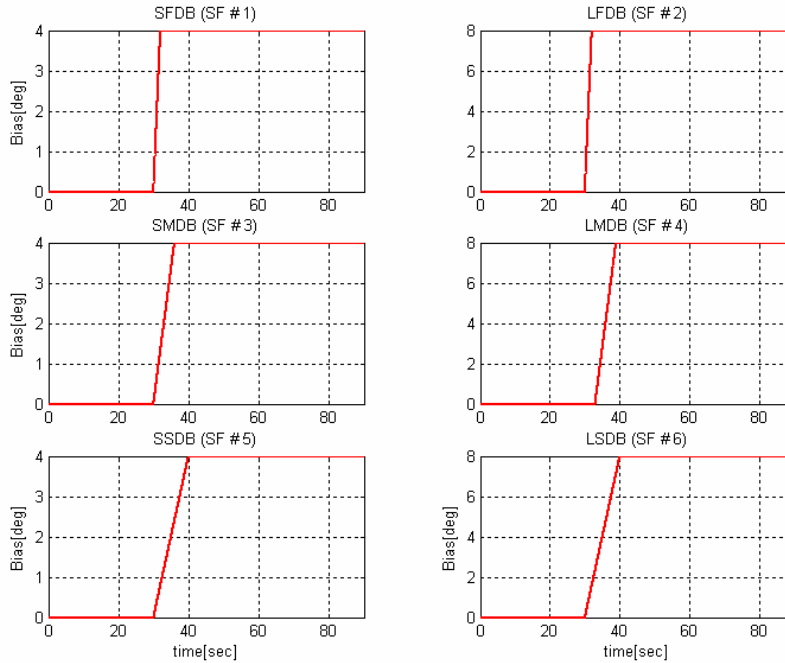


Figure 5-10 Bias for Sensor Failures

5.3.1. Analysis of Actuator Failure Cases (Stabilator Failure)

Two locked types of failures (stabilator/aileron) were selected for conducting the FDII flight experiments in the flight simulator. The goal of this study was two folded. Firstly, to have a quantitative evaluation of pilot compensation in advent of a failure and secondly, to test the FL incorporated FDII scheme in the flight simulator. The study provided a means of testing the FDII scheme in a near realistic way in the flight simulator. Any pilot activity during the flight experiments induces cross-correlation rates and thus helped in evaluating the insensitivity of FDII to disturbances induced. In an ideal situation the FDII should be least effected by a pilot activity in failure warning systems. Since FDII scheme is built on monitoring cross-correlations rate which cannot be isolated when a pilot is in the loop. Any pilot activity is bound to induce coupling and thus FDII scheme is susceptible to wrong detection. The FL parameters are tuned appropriately to reduce the effect of pilot input. The actuator failure cases were repeated 5 times to ensure data consistency and repeatability.

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	2 Deg Left Stab(NO NN)	30.1	31.16	30.22	T
2	5 Deg Left Stab (NO NN)	30.08	30.1	30.2	T
3	8 Deg Left Stab (NO NN)	30.06	30.08	30.16	F(Rudder)
4	2 Deg Right Ail (NO NN)	30.14	30.2	30.24	T
5	5 Deg Right Ail (NO NN)	30.1	30.24	30.26	T
6	8 Deg Right Ail (NO NN)	30.08	30.1	30.16	T
7	2 Deg Left Stab(EMRAN)	30.1	30.14	30.22	T
8	5 Deg Left Stab (EMRAN)	30.08	30.1	30.2	T
9	8 Deg Left Stab (EMRAN)	30.06	30.08	30.18	T
10	2 Deg Right Ail (EMRAN)	30.12	30.34	30.36	T
11	5 Deg Right Ail (EMRAN)	30.1	30.24	30.26	T
12	8 Deg Right Ail (EMRAN)	30.18	30.24	30.26	T

Table 5-12- Actuator Failure FDII results with Adaptive Threshold (Run1)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	2 Deg Left Stab(NO NN)	30.1	30.12	30.22	T
2	5 Deg Left Stab (NO NN)	30.06	30.08	30.18	T
3	8 Deg Left Stab (NO NN)	30.06	30.06	30.16	F(Rudder)
4	2 Deg Right Ail (NO NN)	30.14	31.14	33.34	F(Pitch Sensor)
5	5 Deg Right Ail (NO NN)	30.12	30.14	30.24	T
6	8 Deg Right Ail (NO NN)	30.08	30.24	30.26	T
7	2 Deg Left Stab(EMRAN)	30.1	30.12	30.22	T
8	5 Deg Left Stab (EMRAN)	30.08	31.1	30.18	T
9	8 Deg Left Stab (EMRAN)	30.06	30.08	30.18	T
10	2 Deg Right Ail (EMRAN)	30.12	0.24	30.26	T
11	5 Deg Right Ail (EMRAN)	30.1	30.2	30.22	T
12	8 Deg Right Ail (EMRAN)	30.08	30.16	30.18	T

Table 5-13- Actuator Failure FDII results with Adaptive Threshold (Run2)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	2 Deg Left Stab(NO NN)	30.1	30.14	30.22	T
2	5 Deg Left Stab (NO NN)	30.08	30.1	30.2	T
3	8 Deg Left Stab (NO NN)	30.06	30.08	30.18	T
4	2 Deg Right Ail (NO NN)	30.14	31.14	33.34	F(Pitch Sensor)
5	5 Deg Right Ail (NO NN)	30.12	30.28	30.3	T
6	8 Deg Right Ail (NO NN)	30.08	30.24	30.26	T
7	2 Deg Left Stab(EMRAN)	30.1	30.12	30.22	T
8	5 Deg Left Stab (EMRAN)	30.08	30.1	30.2	T
9	8 Deg Left Stab (EMRAN)	30.06	30.1	30.18	T
10	2 Deg Right Ail (EMRAN)	30.12	30.2	30.22	T
11	5 Deg Right Ail (EMRAN)	30.1	30.18	30.2	T
12	8 Deg Right Ail (EMRAN)	30.08	30.16	30.18	T

Table 5-14- Actuator Failure FDII results with Adaptive Threshold (Run3)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	2 Deg Left Stab(NO NN)	30.1	30.14	30.22	T
2	5 Deg Left Stab (NO NN)	30.08	30.1	30.2	T
3	8 Deg Left Stab (NO NN)	30.06	30.08	30.18	T
4	2 Deg Right Ail (NO NN)	30.12	30.28	30.3	T
5	5 Deg Right Ail (NO NN)	30.1	30.2	30.22	T
6	8 Deg Right Ail (NO NN)	30.08	30.18	30.2	T
7	2 Deg Left Stab(EMRAN)	30.1	30.12	30.22	T
8	5 Deg Left Stab (EMRAN)	30.08	30.1	30.2	T
9	8 Deg Left Stab (EMRAN)	30.06	30.08	30.18	T
10	2 Deg Right Ail (EMRAN)	30.18	30.26	30.28	T
11	5 Deg Right Ail (EMRAN)	30.1	30.18	30.2	T
12	8 Deg Right Ail (EMRAN)	30.08	30.16	30.18	T

Table 5-15- Actuator Failure FDII results with Adaptive Threshold (Run4)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	2 Deg Left Stab(NO NN)	Undetected	Undetected	Undetected	NO DETECTION
2	5 Deg Left Stab (NO NN)	30.08	30.1	30.2	T
3	8 Deg Left Stab (NO NN)	30.06	30.1	30.18	T
4	2 Deg Right Ail (NO NN)	30.14	30.28	30.3	T
5	5 Deg Right Ail (NO NN)	30.1	30.22	30.24	T
6	8 Deg Right Ail (NO NN)	30.08	30.16	30.18	T
7	2 Deg Left Stab(EMRAN)	30.1	30.12	30.18	T
8	5 Deg Left Stab (EMRAN)	30.08	30.1	30.2	T
9	8 Deg Left Stab (EMRAN)	30.06	30.08	30.18	T
10	2 Deg Right Ail (EMRAN)	30.12	30.24	30.26	T
11	5 Deg Right Ail (EMRAN)	30.1	30.18	30.2	T
12	8 Deg Right Ail (EMRAN)	30.08	30.16	30.18	T

Table 5-16- Actuator Failure FDII results with Adaptive Threshold (Run5)

As shown with Run 1 in Table 5-12, only one case is incorrectly identified. Two cases are incorrectly identified in Run2, one in Run 3, none in Run 4 and one case is undetected in Run 5. It should be noted that prior to running the piloted experiments, the FDII scheme was run on desktop without any pilot and there were no cases of false FDII. The pilot input in the experiments has triggered false alarms and detections in these cases. Table 5-17 shows the summary of FDII tests for actuator failure cases.

Tests	Total Failures Tested	Correctly FDII	False Detections	False Isolations	False Identifications	No Detections
Stabilator failure	30	27	0	2	2	1
Aileron failure	30	28	0	2	2	0

Table 5-17-Summary of Piloted Tests of Adaptive FDII scheme for Actuator Failure

In the subsequent sections a performance evaluation of the FDII schemes is carried out in terms of Tracking Error (TE), amount of required pilot activity to accomplish the mission profile.

5.3.1.1. Tracking Error Analysis of Stabilator Failures

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	-2.39E-01	-6.87E-02	-4.47E-01	-1.62E-01	-4.47E-01	-6.87E-02
Mean Pitch	9.07E-02	4.43E-02	1.80E-01	9.97E-02	3.06E-01	4.43E-02
Mean Yaw	-1.75E-02	-2.00E-02	-3.45E-02	-3.82E-02	-6.11E-02	-2.00E-02
SD Roll	9.50E-01	8.03E-01	1.67E+00	1.63E+00	3.03E+00	8.03E-01
SD Pitch	4.17E-01	3.38E-01	6.76E-01	6.70E-01	1.20E+00	3.38E-01
SD Yaw	1.70E-01	1.68E-01	1.95E-01	2.06E-01	2.72E-01	1.68E-01

Table 5-18- TE statistics for Left Stabilator (Run1)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	-2.09E-01	-6.38E-02	-4.45E-01	-1.47E-01	-4.45E-01	-6.38E-02
Mean Pitch	9.37E-02	3.10E-02	2.04E-01	7.59E-02	3.53E-01	3.10E-02
Mean Yaw	-1.71E-02	-1.54E-02	-3.66E-02	-3.40E-02	-6.56E-02	-1.54E-02
SD Roll	8.22E-01	7.97E-01	1.66E+00	1.68E+00	3.06E+00	7.97E-01
SD Pitch	4.01E-01	3.23E-01	7.95E-01	6.70E-01	1.37E+00	3.23E-01
SD Yaw	1.75E-01	1.70E-01	2.12E-01	2.11E-01	2.96E-01	1.70E-01

Table 5-19- TE statistics for Left Stabilator (Run2)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	-2.02E-01	-7.51E-02	-4.35E-01	-1.57E-01	-4.35E-01	-7.51E-02
Mean Pitch	9.12E-02	3.73E-02	1.93E-01	8.50E-02	4.13E-01	3.73E-02
Mean Yaw	-1.70E-02	-1.75E-02	-3.60E-02	-3.74E-02	-7.69E-02	-1.75E-02
SD Roll	9.20E-01	7.89E-01	1.65E+00	1.69E+00	2.87E+00	7.89E-01
SD Pitch	4.19E-01	3.16E-01	7.69E-01	6.82E-01	1.27E+00	3.16E-01
SD Yaw	1.77E-01	1.70E-01	2.10E-01	2.13E-01	2.79E-01	1.70E-01

Table 5-20- TE statistics for Left Stabilator (Run3)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	-2.01E-01	-1.18E-01	-4.31E-01	-2.21E-01	-4.31E-01	-1.18E-01
Mean Pitch	8.13E-02	5.03E-02	1.75E-01	1.04E-01	3.22E-01	5.03E-02
Mean Yaw	-1.61E-02	-2.08E-02	-3.44E-02	-4.10E-02	-6.33E-02	-2.08E-02
SD Roll	7.93E-01	7.73E-01	1.65E+00	1.61E+00	2.92E+00	7.73E-01
SD Pitch	3.43E-01	3.13E-01	6.62E-01	6.40E-01	1.17E+00	3.13E-01
SD Yaw	1.67E-01	1.69E-01	1.97E-01	2.06E-01	2.70E-01	1.69E-01

Table 5-21- TE statistics for Left Stabilator (Run4)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	-8.69E-02	-9.09E-02	-4.90E-01	-1.96E-01	-4.90E-01	-9.09E-02
Mean Pitch	3.31E-02	4.04E-02	1.94E-01	9.58E-02	3.03E-01	4.04E-02
Mean Yaw	-6.99E-03	-1.76E-02	-3.79E-02	-3.91E-02	-6.04E-02	-1.76E-02
SD Roll	3.54E-01	7.72E-01	1.51E+00	1.63E+00	3.16E+00	7.72E-01
SD Pitch	1.78E-01	3.18E-01	6.02E-01	6.61E-01	1.26E+00	3.18E-01
SD Yaw	1.58E-01	1.69E-01	1.91E-01	2.07E-01	2.87E-01	1.69E-01

Table 5-22- TE statistics for Left Stabilator (Run5)

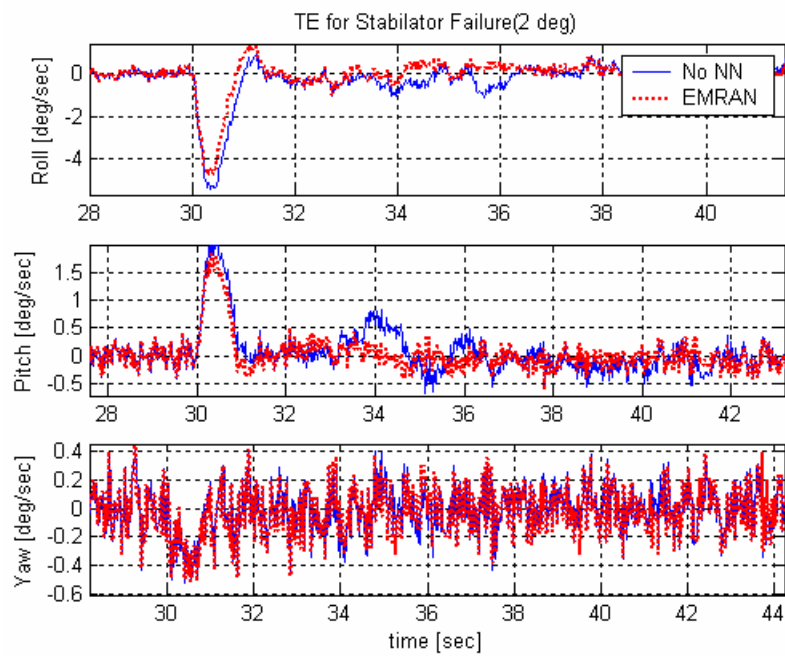


Fig. 5-11-TE for Left Stabilator soft failure (Run1)

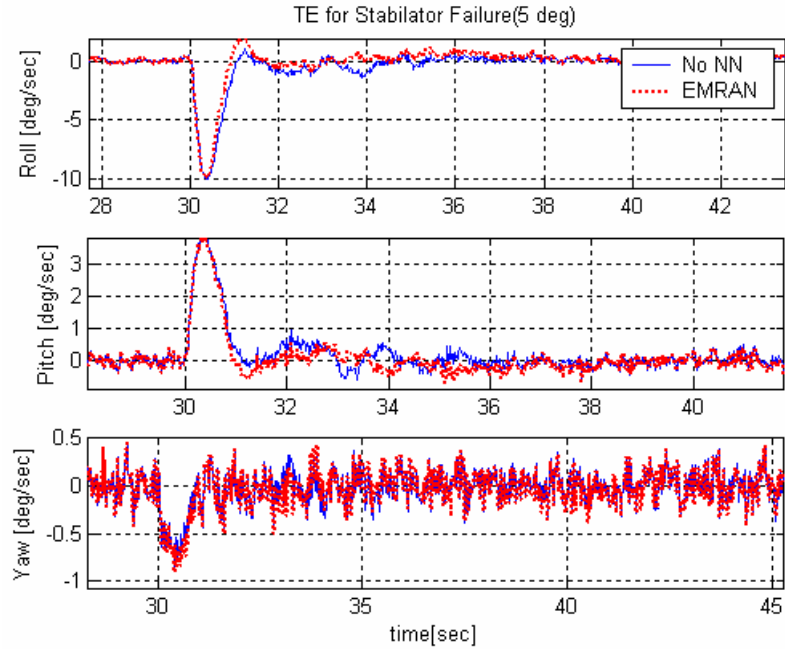


Fig. 5-1 TE for Left Stabilator Medium failure (Run1)

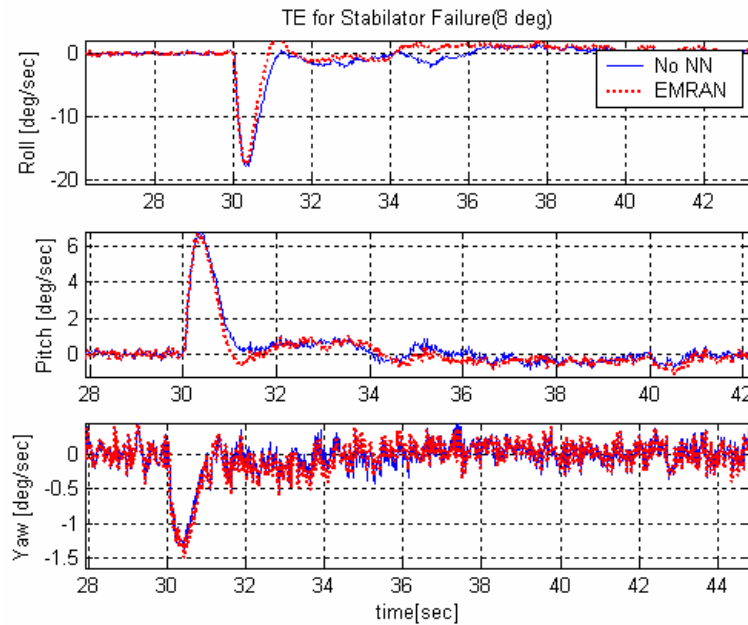


Fig. 5-2 TE for Left Stabilator hard failure (Run1)

The stabilator failure induces a coupling in roll and pitch with little effect on yaw channel. Since the pilot was instructed for a same mission profile therefore the TE results can be averaged for the 5 runs for any meaningful conclusion. These are tabulated in Table 5-23 for the soft, medium and hard failure. As can be seen for a soft failure (2deg), the mean of TE along pitch channel for EMRAN NN is 48% lower than for a no NN

case. For the pitch channel it is a 56% lesser than a no NN case. Along the yaw channel there is however increases of 22% in the TE compared to a no NN case. Similarly, in case of the medium stabilator failure (5 deg), the mean of TE shows a decrease of 52% along pitch and 61% along roll channel. The yaw TE increase with EMRAN compared to no NN case. For a hard failure (8 deg), however the mean of TE along pitch, roll and yaw channel shows a decrease of 88%, 81% and 72% respectively from a no NN case.

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	-1.88E-01	-8.33E-02	-4.50E-01	-1.77E-01	-4.50E-01	-8.33E-02
Mean Pitch	7.80E-02	4.06E-02	1.89E-01	9.22E-02	3.40E-01	4.06E-02
Mean Yaw	-1.49E-02	-1.83E-02	-3.59E-02	-3.80E-02	-6.54E-02	-1.83E-02
SD Roll	7.68E-01	7.87E-01	1.63E+00	1.65E+00	3.01E+00	7.87E-01
SD Pitch	3.52E-01	3.22E-01	7.01E-01	6.65E-01	1.25E+00	3.22E-01
SD Yaw	1.69E-01	1.69E-01	2.01E-01	2.09E-01	2.81E-01	1.69E-01

Table 5-23- Average of 5 runs for Stabilator failure

5.3.1.2. Pilot Activity Analysis

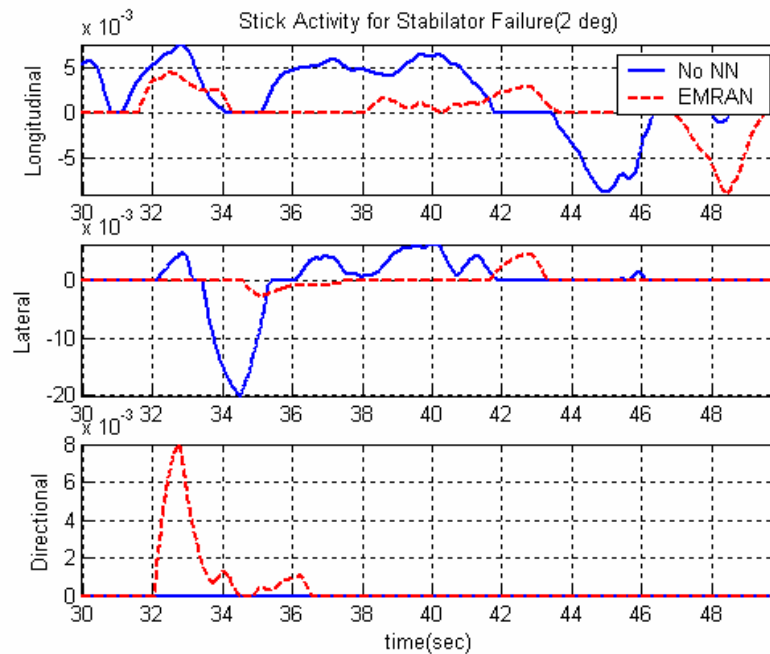


Fig 5-12-Pilot Activity for Left Stabilator Failure (Run1)

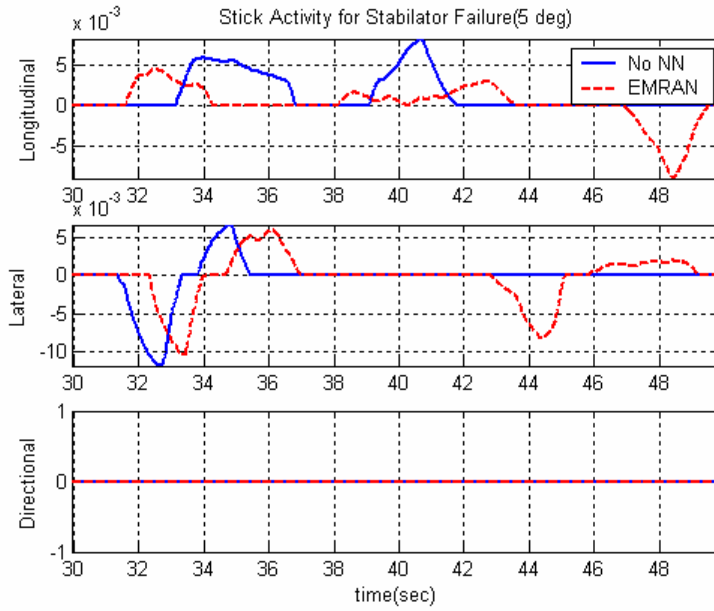


Figure 5-13 Pilot Activity for Medium Stabilator Failure (Run1)

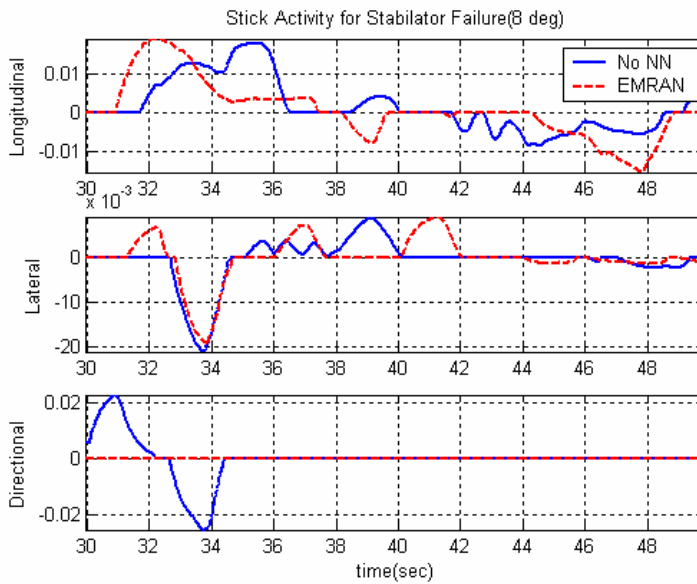


Fig. 5-14-Pilot Activity for Hard Stabilator Failure (Run1)

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	1.9914	0.4921	0.9489	0.3878	2.9470	3.0497
Longitudinal	1.9737	0.1797	0.9776	0.9531	2.0409	1.6669
Directional	0.0000	0.4323	0.0000	0.0000	2.7738	0.0000

Table 5-24- Pilot Activity for Left Stabilator Failure (Run 1)

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	1.2450	0.9914	0.1850	1.8606	5.8352	1.9231
Longitudinal	1.2853	0.8282	1.4829	0.7906	3.6638	0.4619
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-25- Pilot Activity for Left Stabilator Failure (Run 2)

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	1.3565	1.3148	1.5163	2.6622	4.1735	2.8421
Longitudinal	0.0798	0.1480	1.4335	1.3242	1.4388	2.0285
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-26- Pilot Activity for Left Stabilator Failure (Run 3)

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	1.0702	1.1366	2.3505	0.5165	3.8280	1.6925
Longitudinal	0.8165	0.4364	0.9862	0.3965	1.3527	0.7399
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-27- Pilot Activity for Left Stabilator Failure (Run 4)

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	0.6453	0.1718	1.5008	1.3012	4.6993	3.5295
Longitudinal	0.0000	0.5405	0.6548	1.2617	2.7355	1.2537
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-28- Pilot Activity for Left Stabilator Failure (Run 5)

The average pilot activity for 5 runs of the stabilator failure is shown in Table 5-29. The main effect of this type of failure is on the pitch and roll channel. It can be seen that for a soft failure EMRAN has 48% lower pilot activity in longitudinal channel than a no NN case. Similarly, it is 34% lower in lateral channel. For a medium failure, EMRAN shows a decrease in 14% compared to a No NN while mean TE increases along lateral channel by 3%. Similarly, for hard failure there is a reduction in pilot activity for EMRAN in comparison to a no NN case. It is 45% lower for longitudinal and 39% lower for lateral. In general there is a trend of reduced pilot activity and it was corroborated by pilot feedback as well. The pilot experienced a reduced workload with EMRAN NN. Yaw

channel pilot activity however is very minimal and cannot be concluded for any meaningful observation.

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	1.2617	0.8213	1.3003	1.3457	4.2966	2.6074
Longitudinal	0.8311	0.4265	1.1070	0.9452	2.2463	1.2302
Directional	0.0000	0.0865	0.0000	0.0000	0.5548	0.0000

Table 5-29 Average Pilot Activity for Stabilator Failure

Since the objective of design of the controller required a minimal amount of TE together with a little stick activity on behalf of the pilot in advent of failures. So a composite parameter (CP) denoted as η is an appropriate indication of the performance index of the schemes which is the product of the TE and the PC and is shown below.

$$\eta = TE \times PC$$

The CP for stabilator failures is shown in Table 5-30 for the soft, medium and hard failure. The CP has shown a remarkable reduction for an EMRAN augmented scheme than a No NN scheme. For example if we look at Run 1, EMRAN shows a reduction in η for the roll and pitch channels. The composite parameter η reduced by 86% along roll and 80% along pitch channel for the soft failure. EMRAN reduced η by 74% and 45% in roll and pitch channel for a medium failure. Similarly, for the hard failure EMRAN have reduced the η by 84% and 87% respectively along the roll and pitch channel. It is also observed that there is no appreciable improvement along the yaw channel for the soft and medium failures. However, for hard failures there is always an improvement in case of EMRAN NN.

Composite Parameter (Stabilator)- Run1						
	2 deg		5 deg		8 deg	
CP	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.7161	0.1025	0.8709	0.2250	1.7639	0.2783
Pitch	0.2696	0.0522	0.3552	0.1948	0.9313	0.1181
Yaw	0.0175	0.0286	0.0345	0.0382	0.2307	0.0200
Composite Parameter (Stabilator)- Run2						
CP	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.4699	0.1271	0.5271	0.4193	3.0404	0.1865
Pitch	0.2140	0.0566	0.5058	0.1360	1.6473	0.0453
Yaw	0.0171	0.0154	0.0366	0.0340	0.0656	0.0154
Composite Parameter (Stabilator)- Run3						
	2 deg		5 deg		8 deg	
CP	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.4750	0.1738	1.0941	0.5766	2.2495	0.2884
Pitch	0.0985	0.0428	0.4699	0.1975	1.0077	0.1128
Yaw	0.0170	0.0175	0.0360	0.0374	0.0769	0.0175
Composite Parameter (Stabilator)- Run4						
	2 deg		5 deg		8 deg	
CP	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.4162	0.2524	1.4450	0.3355	2.0822	0.3180
Pitch	0.1476	0.0723	0.3471	0.1458	0.7580	0.0875
Yaw	0.0161	0.0208	0.0344	0.0410	0.0633	0.0208
Composite Parameter (Stabilator)- Run5						
	2 deg		5 deg		8 deg	
CP	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.1430	0.1065	1.2260	0.4499	2.7941	0.4118
Pitch	0.0331	0.0622	0.3214	0.2166	1.1308	0.0910
Yaw	0.0070	0.0176	0.0379	0.0391	0.0604	0.0176

Table 5-30 Composite Parameter for Stabilator Failure

A lower value of η is desirable and is also demonstrated from the CP values of the stabilator failure results. An average of all the tests is tabulated in Table 5-31 and it can be concluded that there is a reduction in CP with EMRAN augmentation in all cases along the roll and pitch channel. There is a reduction of 64%, 59% and 87% along the roll channel for soft, medium and hard failure respectively. Similarly these values are 59%, 54% and 91% along the pitch channel. For the yaw channel there is an increase in CP i.e.32% and 6% for soft and medium failure respectively. For the hard failure EMRAN reduces it by 82%.

From the analysis of the CP it can be concluded that EMRAN has improved the performance consistently and η is an indication of reduced pilot workload in the advent of a failure. For hard failures which require more pilot effort to compensate and maintain a level flight EMRAN had reduced the pilot workload considerably.

CP	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.4244	0.1518	1.0342	0.4142	2.3814	0.3006
Pitch	0.1428	0.0580	0.3984	0.1793	1.1022	0.0906
Yaw	0.0149	0.0198	0.0359	0.0380	0.1018	0.0183

Table 5-31 Average CP for Stabilator Failure Cases

5.3.1.3. FDII Analysis of Stabilator Failures

The MQEE is the main parameter responsible for detection of actuator failure and it rises fast as compared to OQEE. Figure 5-15 shows the MQEE crossing the soft failure UB immediately after the failure. However, the hard UB is still not crossed. In case of medium and hard failures as shown in Fig. 5-16 and Fig. 5-17 the hard UB is crossed and thus a failure detection flag is triggered. The floating bounds on OQEE are however not reached to trigger any detection flag.

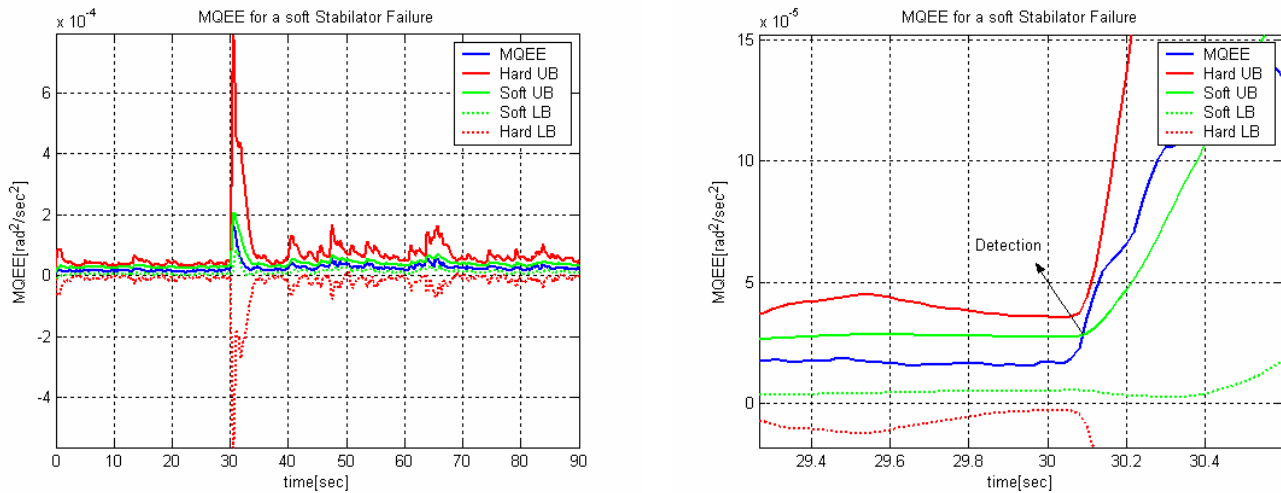


Figure 5-15 Soft Failure Detection with FL Soft Bound for Stabilator using MQEE

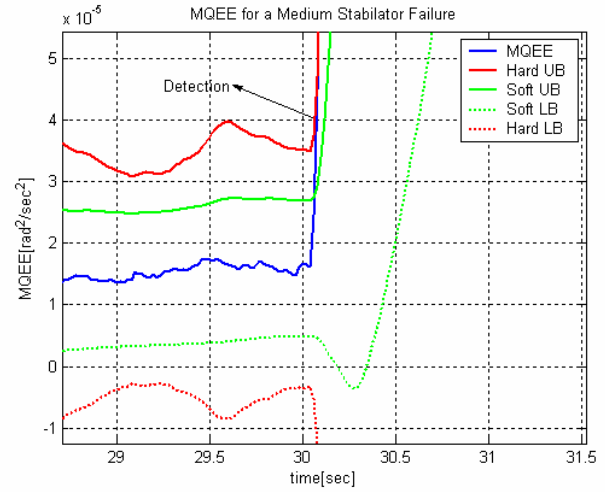
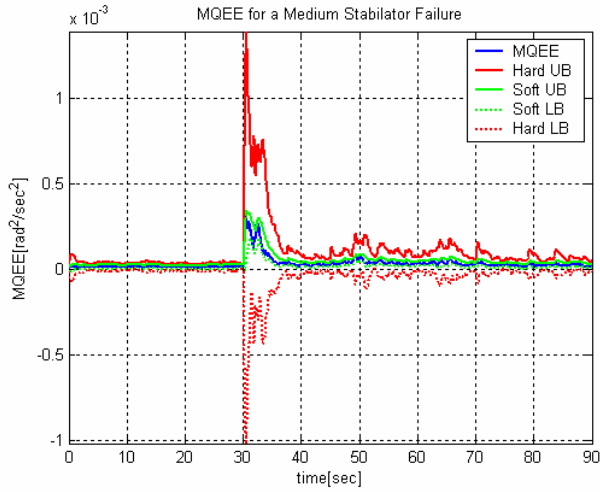


Fig. 5-16- Medium Failure Detection with FL Hard Bound for Stabilator using MQEE

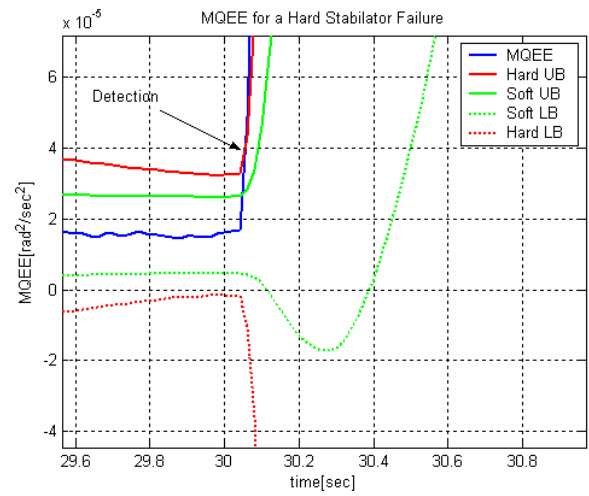
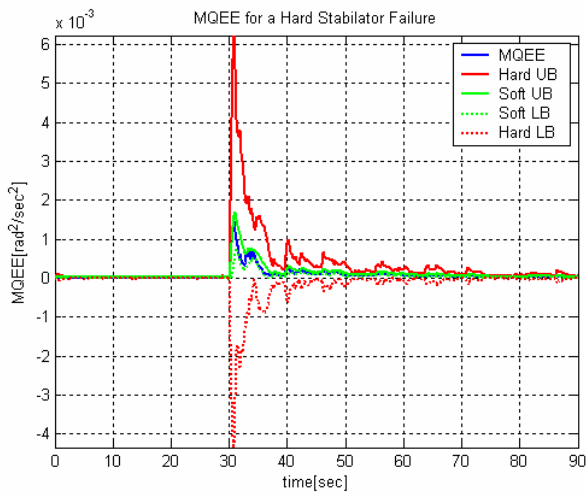


Fig. 5-17- Hard Failure Detection with FL Hard Bound for Stabilator using MQEE

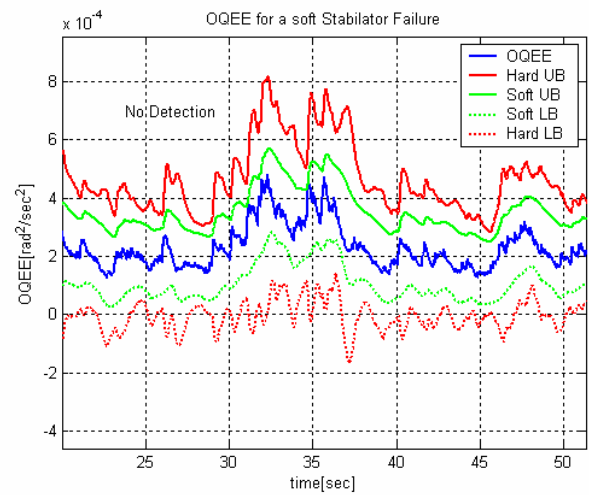
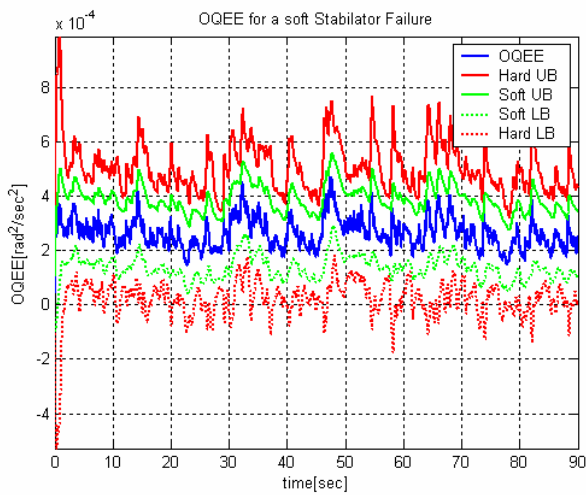


Fig. 5-18-No Detection with OQEE for Soft Stabilator Failure

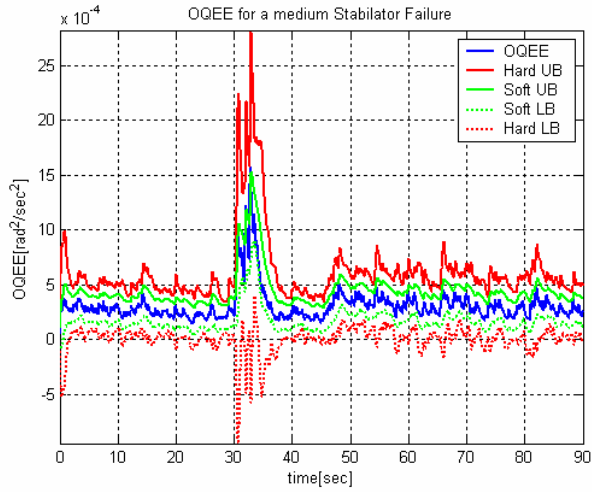


Fig. 5-19- No Detection with OQEE for Medium Stabilator Failure

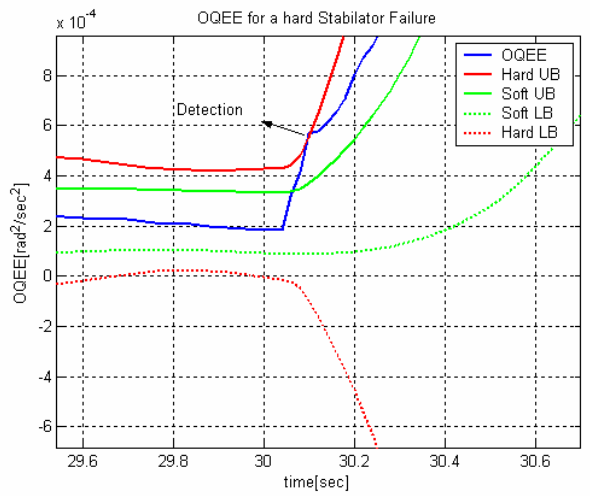
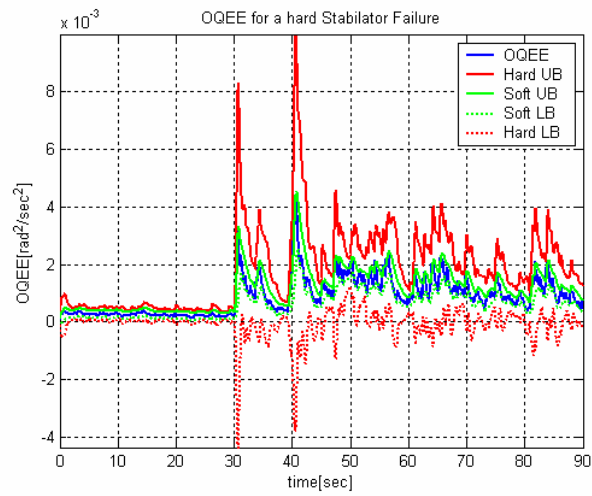


Fig. 5-20- Detection with OQEE for Hard Stabilator Failure

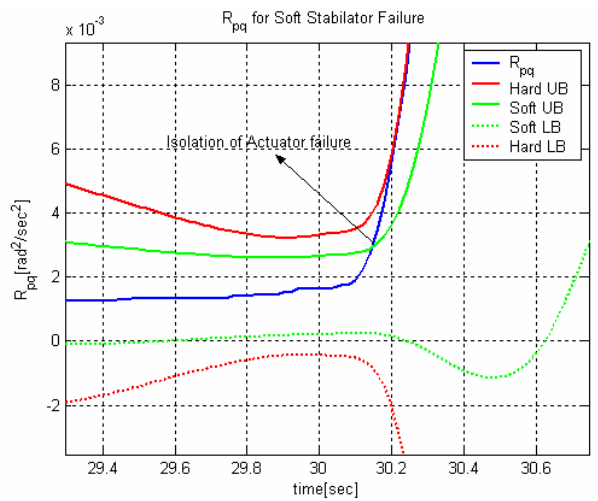
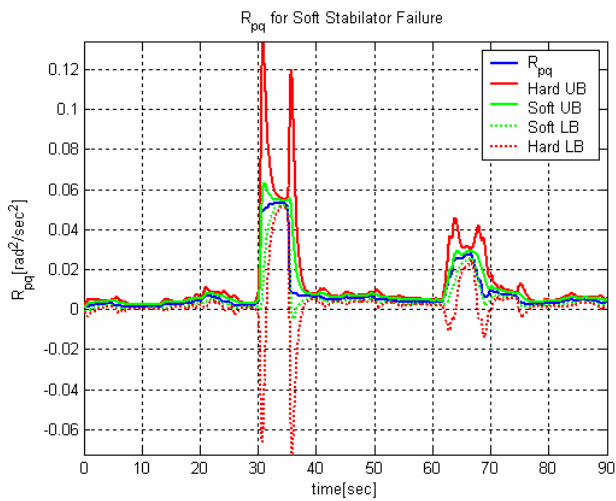


Fig. 5-21-Soft Failure Isolation with R_{pq}

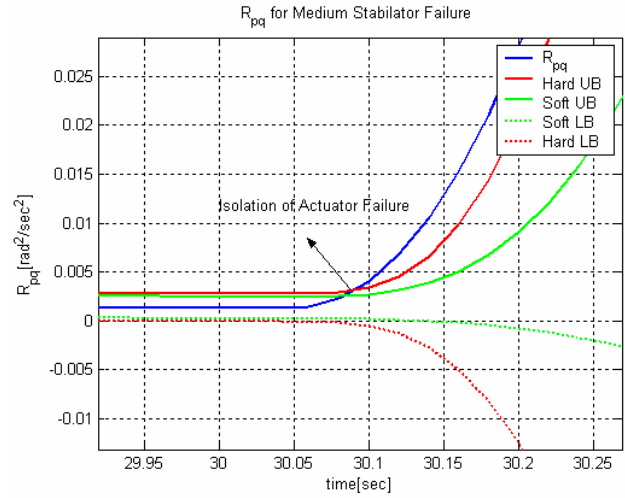
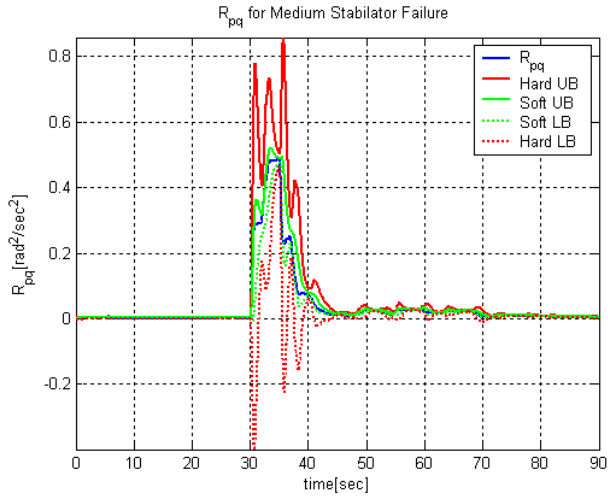


Fig. 5-22- Medium Failure Isolation with R_{pq}

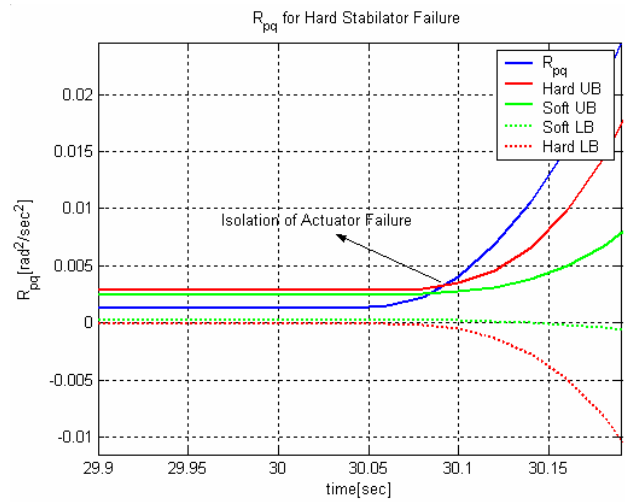
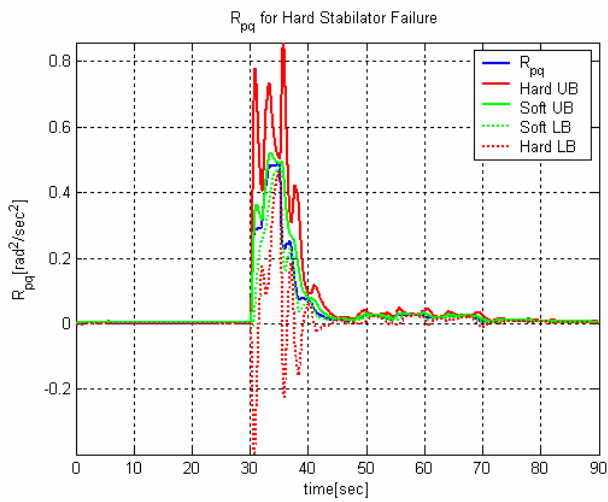


Fig. 5-23-Hard Failure Isolation with R_{pq}

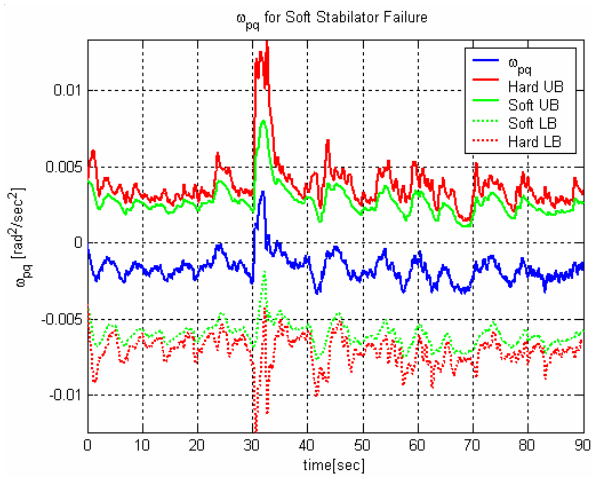


Fig. 5-24-Identification of Soft Stabilator Failure with ω_{pq}

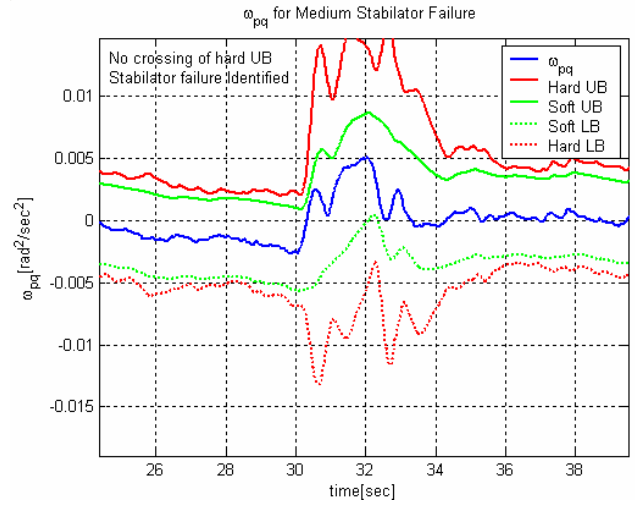
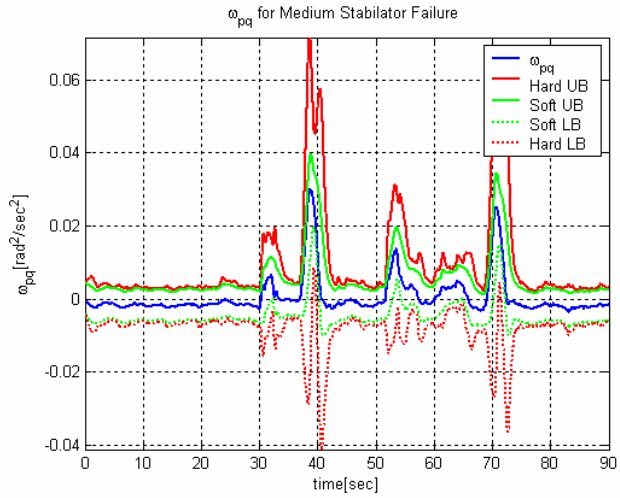


Fig. 5-25- Identification of Medium Stabilator Failure with ω_{pq}

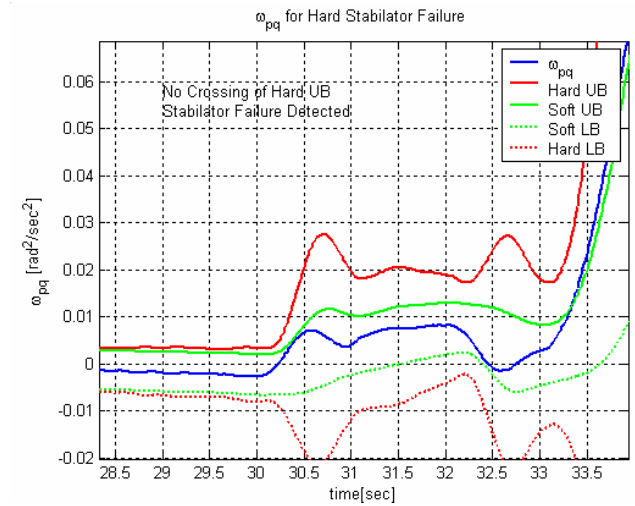
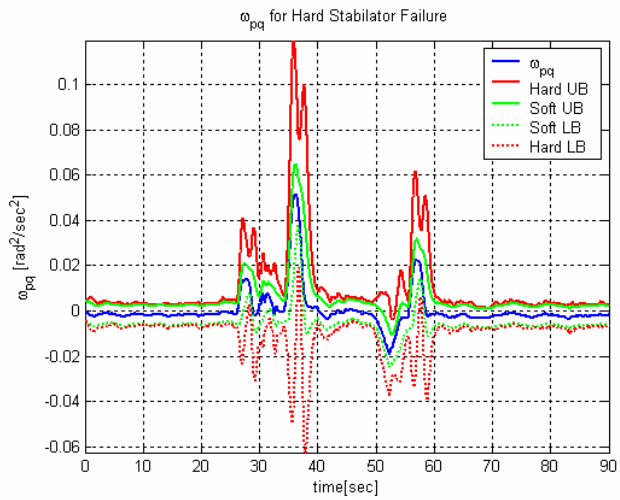


Fig. 5-26- Identification of Hard Stabilator Failure with ω_{pq}

5.3.2. Analysis of Actuator Failure Cases (Aileron Failure)

5.3.2.1. Tracking Error Analysis

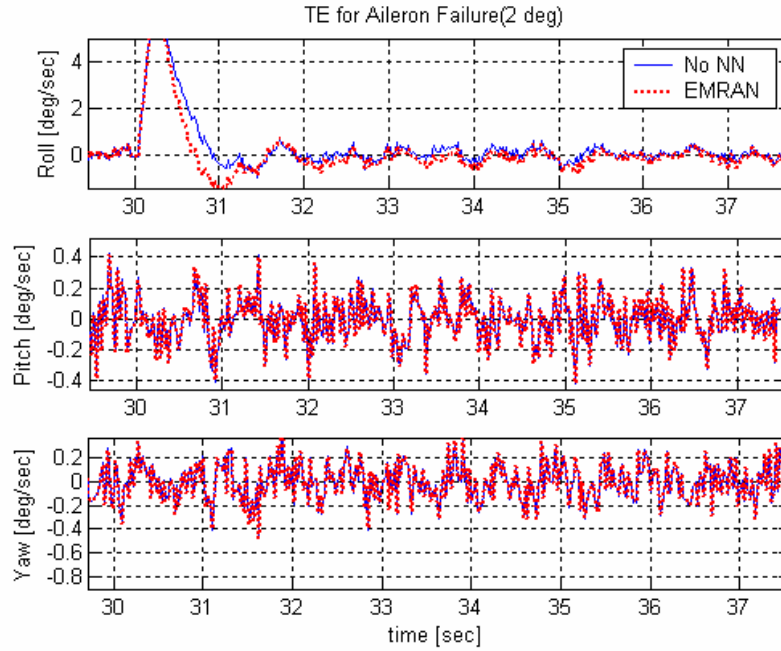


Fig. 5-27- TE for a Soft Aileron failure (Run1)

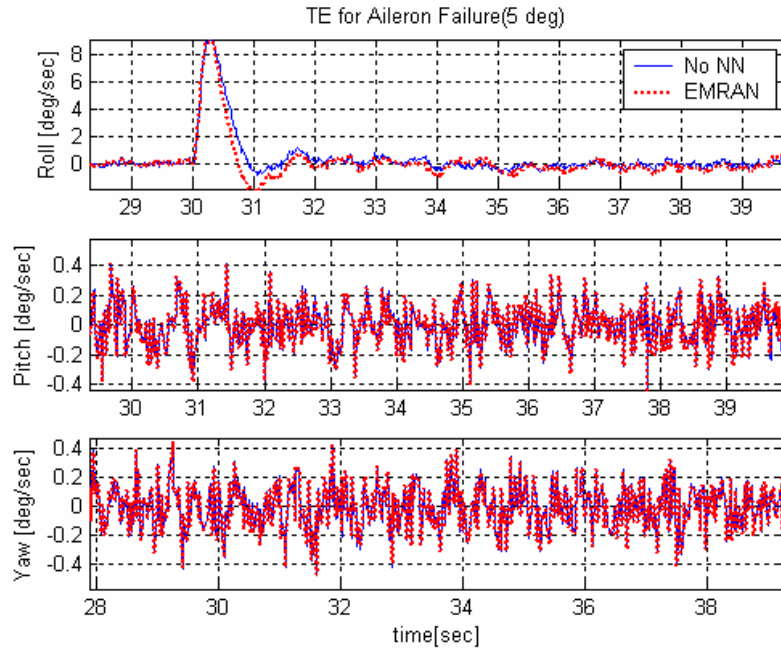


Fig. 5-28- TE for a Medium Aileron failure (Run1)

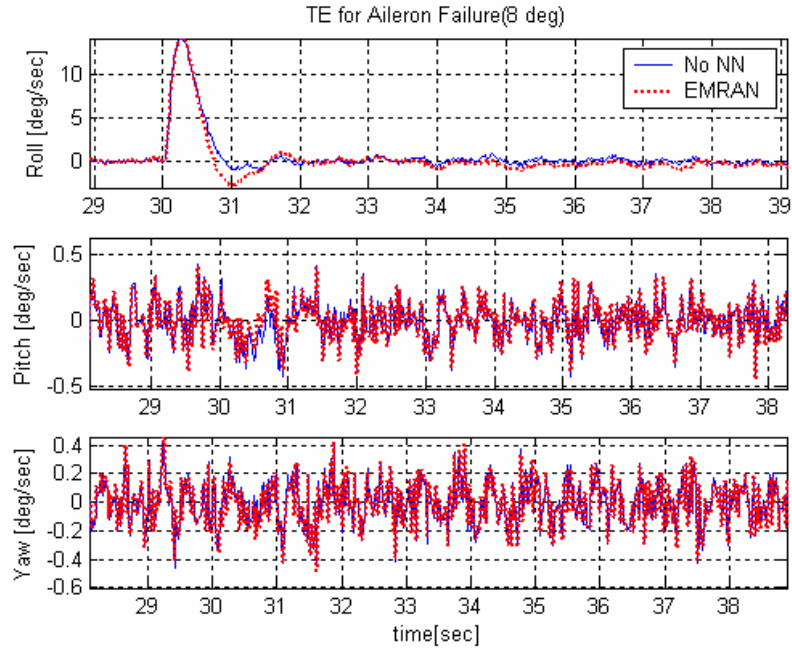


Fig. 5-29- TE for a Hard Aileron failure (Run1)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	1.74E-01	4.95E-02	2.66E-01	7.57E-02	2.66E-01	4.95E-02
Mean Pitch	6.29E-05	-1.49E-03	-1.16E-03	-1.82E-03	-3.71E-03	-1.49E-03
Mean Yaw	-6.10E-04	-3.43E-04	-9.82E-04	-1.70E-03	-1.80E-03	-3.43E-04
SD Roll	8.53E-01	8.19E-01	1.41E+00	1.37E+00	2.07E+00	8.19E-01
SD Pitch	1.43E-01	1.44E-01	1.43E-01	1.44E-01	1.48E-01	1.44E-01
SD Yaw	1.52E-01	1.53E-01	1.52E-01	1.53E-01	1.54E-01	1.53E-01

Table 5-32-TE statistics for aileron failure (Run 1)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	1.24E-01	4.68E-02	2.02E-01	1.90E-01	2.02E-01	4.68E-02
Mean Pitch	2.50E-04	-1.90E-03	4.79E-03	-2.64E-03	-7.11E-04	-1.90E-03
Mean Yaw	-4.56E-04	-1.02E-03	-6.82E-04	-2.37E-03	-9.01E-04	-1.02E-03
SD Roll	7.06E-01	8.18E-01	1.14E+00	2.10E+00	1.79E+00	8.18E-01
SD Pitch	1.48E-01	1.47E-01	1.52E-01	1.47E-01	1.48E-01	1.47E-01
SD Yaw	1.58E-01	1.55E-01	1.58E-01	1.55E-01	1.58E-01	1.55E-01

Table 5-33-TE statistics for aileron failure (Run2)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	1.23E-01	6.50E-02	2.01E-01	1.09E-01	2.01E-01	6.50E-02
Mean Pitch	2.81E-03	-3.62E-03	8.07E-04	-2.15E-03	-3.76E-04	-3.62E-03
Mean Yaw	-4.37E-04	-5.98E-04	-6.54E-04	-1.40E-03	-6.97E-04	-5.98E-04
SD Roll	7.09E-01	8.10E-01	1.13E+00	1.33E+00	1.79E+00	8.10E-01
SD Pitch	1.48E-01	1.47E-01	1.48E-01	1.47E-01	1.48E-01	1.47E-01
SD Yaw	1.59E-01	1.55E-01	1.58E-01	1.55E-01	1.58E-01	1.55E-01

Table 5-34-TE statistics for aileron failure (Run3)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	1.51E-01	3.28E-02	2.55E-01	9.64E-02	2.55E-01	3.28E-02
Mean Pitch	-2.56E-03	-5.25E-03	-2.29E-03	-2.08E-03	-3.42E-03	-5.25E-03
Mean Yaw	-3.87E-04	-2.53E-04	-1.17E-03	-1.33E-03	-1.69E-03	-2.53E-04
SD Roll	8.17E-01	5.78E-01	1.34E+00	1.33E+00	1.99E+00	5.78E-01
SD Pitch	1.47E-01	1.47E-01	1.46E-01	1.47E-01	1.49E-01	1.47E-01
SD Yaw	1.54E-01	1.55E-01	1.54E-01	1.55E-01	1.56E-01	1.55E-01

Table 5-35-TE statistics for aileron failure (Run4)

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	1.54E-01	6.53E-02	2.60E-01	1.15E-01	2.60E-01	6.53E-02
Mean Pitch	-2.44E-03	-2.02E-03	-3.06E-03	-2.22E-03	-4.64E-03	-2.02E-03
Mean Yaw	-6.58E-04	-5.12E-04	-1.11E-03	-1.43E-03	-1.95E-03	-5.12E-04
SD Roll	8.20E-01	8.06E-01	1.34E+00	1.32E+00	2.03E+00	8.06E-01
SD Pitch	1.46E-01	1.47E-01	1.46E-01	1.47E-01	1.49E-01	1.47E-01
SD Yaw	1.54E-01	1.55E-01	1.54E-01	1.55E-01	1.55E-01	1.55E-01

Table 5-36-TE statistics for aileron failure (Run5)

The average of 5 runs for the aileron failure is shown in Table 5-37 . For a soft failure, the mean roll TE for EMRAN is 64% lower than no NN case. For a medium failure it is 51% lower and for a hard failure it is 78% lower than No NN. The mean TE along pitch channel increase with EMRAN for the soft, medium and hard failures.

Tracking Error Statistics	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Mean Roll	1.45E-01	5.19E-02	2.37E-01	1.17E-01	2.37E-01	5.19E-02
Mean Pitch	-3.75E-04	-2.85E-03	-1.80E-04	-2.18E-03	-2.57E-03	-2.85E-03
Mean Yaw	-5.10E-04	-5.46E-04	-9.19E-04	-1.64E-03	-1.41E-03	-5.46E-04
SD Roll	7.81E-01	7.66E-01	1.27E+00	1.49E+00	1.94E+00	7.66E-01
SD Pitch	1.46E-01	1.46E-01	1.47E-01	1.46E-01	1.49E-01	1.46E-01
SD Yaw	1.55E-01	1.55E-01	1.55E-01	1.55E-01	1.56E-01	1.55E-01

Table 5-37-Average of 5 Runs for Aileron Failure

5.3.2.2. Stick Activity Analysis

The stick activity for 20 seconds after the failure is shown in Table 5-38- Table 5-42. As can be seen from Table 5-38 , EMRAN has reduced pilot activity for pitch channel.

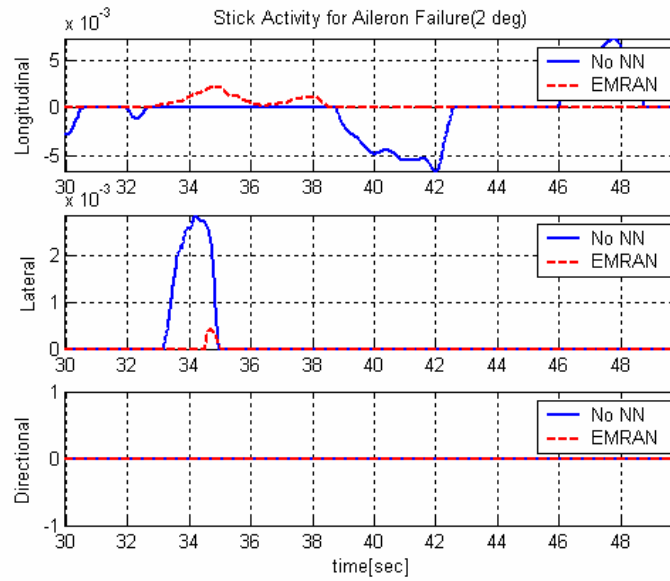


Figure 5-30 Pilot Activity for Soft Aileron Failure (Run 1)

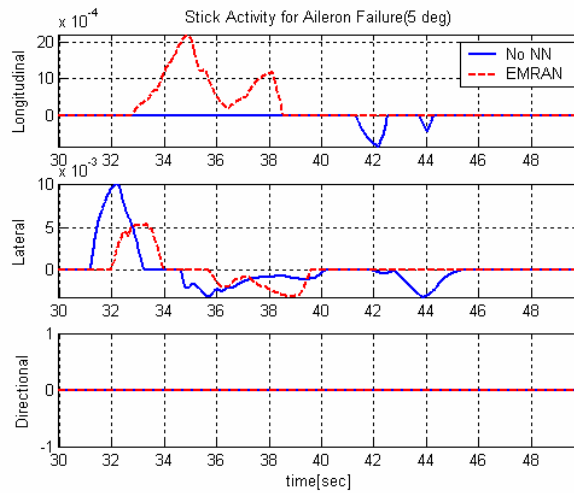


Figure 5-31 Pilot Activity for Medium Aileron Failure (Run 1)

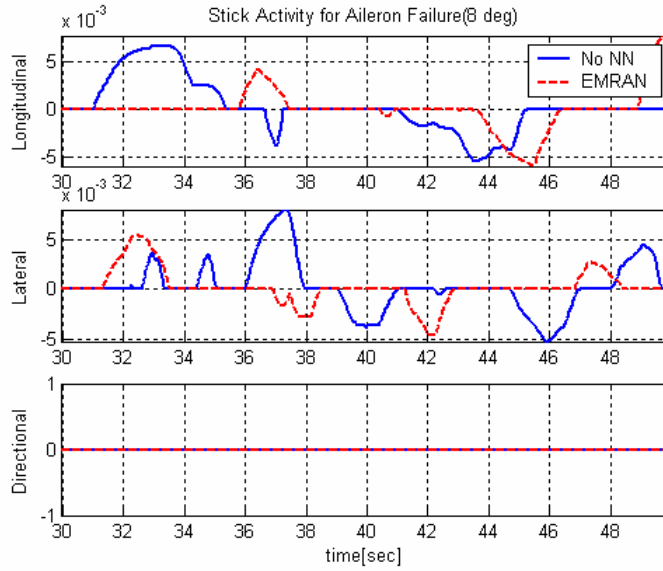


Figure 5-32 Pilot Activity for Hard Aileron Failure (Run 1)

	2 deg		5 deg		8 deg	
Stick Activity	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	0.24669	0.26045	0	0.85622	1.1329	0.8816
Longitudinal	0.17477	0.0066015	1.0182	0.741	0.43793	0.66156
Directional	0	0	0	0	0	0

Table 5-38- Pilot Activity for Right Aileron Failure (Run 1)

	2 deg		5 deg		8 deg	
Activity	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	0	0	1.0384	0	0	1.7698
Longitudinal	0	0.74725	0.1053	0.31267	0.13645	0.30788
Directional	0	0	0	0	0	0

Table 5-39- Pilot Activity for Right Aileron Failure (Run 2)

	2 deg		5 deg		8 deg	
Stick Activity	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	0.34459	1.34	0	0	0	0.31442
Longitudinal	0	0.071806	0.69882	0.3789	0.5996	0.70339
Directional	0	0	0	0	0	0

Table 5-40- Pilot Activity for Right Aileron Failure (Run 3)

	2 deg		5 deg		8 deg	
Stick Activity	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	0.0089399	0.049124	0.050194	0	0.84842	0
Longitudinal	0.57014	0.62955	0.49721	0.50048	0.35209	0.78907
Directional	0	0	0	0	0	0

Table 5-41- Pilot Activity for Right Aileron Failure (Run 4)

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	0.067996	0	0.3772	0	1.0083	0.20597
Longitudinal	0.68904	0.033492	0.19058	0.21471	0.77237	0.51599
Directional	0	0	0	0	0	0

Table 5-42- Pilot Activity for Right Aileron Failure (Run 5)

Considering the average pilot activity for the aileron failures as shown in Table 5-43, it can be seen that pilot activity is comparable for No NN and EMRAN in longitudinal channel. As a general trend EMRAN needs more pilot workload in roll channel. No pilot activity is in required in the yaw channel for aileron failure cases.

Stick Activity	2 deg		5 deg		8 deg	
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Lateral	0.1336432	0.3299148	0.2931588	0.171244	0.597924	0.634358
Longitudinal	0.28679	0.2977399	0.502022	0.429552	0.459688	0.595578
Directional	0	0	0	0	0	0

Table 5-43 Average Pilot Activity for Aileron Failure Cases

In a similar way as stabilator failure cases the CP is evaluated for the aileron failures and is shown in Table 5-44. The roll channel is predominant in aileron failures and much of the pilot activity is in this channel for maintaining the mission profile. The EMRAN augmentations have shown a definite trend of reducing the η compared to the No NN case. For example if we consider Run 1, there is a reduction of 71% for a soft failure, 47% for a medium failure and 83% for a hard failure along the roll channel. The similar trend is for the rest of the failures. An interesting observation is that the reduction is highest for a hard failure except Run2 and it gives a clear indication that EMRAN have helped the pilot in dealing with the failure better than when there is No NN. This was also subjectively corroborated by the feedback from the pilot while the experiments were being conducted. Thus CP is an important parameter in analysis of NN performance when there is a pilot in the loop.

Run1						
	2 deg		5 deg		8 deg	
CP	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.2167	0.0623	0.2663	0.1404	0.5679	0.0931
Pitch	0.0001	0.0015	0.0023	0.0032	0.0053	0.0025
Yaw	0.0006	0.0003	0.0010	0.0017	0.0018	0.0003
Run2						
Roll	0.1240	0.0468	0.4124	0.1899	0.2023	0.1297
Pitch	0.0003	0.0033	0.0053	0.0035	0.0008	0.0025
Yaw	0.0005	0.0010	0.0007	0.0024	0.0009	0.0010
Run3						
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.1658	0.1522	0.2007	0.1093	0.2007	0.0855
Pitch	0.0028	0.0039	0.0014	0.0030	0.0006	0.0062
Yaw	0.0004	0.0006	0.0007	0.0014	0.0007	0.0006
Run4						
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.1519	0.0344	0.2683	0.0964	0.4722	0.0328
Pitch	0.0040	0.0086	0.0034	0.0031	0.0046	0.0094
Yaw	0.0004	0.0003	0.0012	0.0013	0.0017	0.0003
Run5						
	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.1642	0.0653	0.3585	0.1147	0.5228	0.0788
Pitch	0.0041	0.0021	0.0036	0.0027	0.0082	0.0031
Yaw	0.0007	0.0005	0.0011	0.0014	0.0019	0.0005

Table 5-44 Composite Parameter for Aileron Failures

	2 deg		5 deg		8 deg	
CP	No NN	EMRAN	No NN	EMRAN	No NN	EMRAN
Roll	0.1645	0.0690	0.3065	0.1373	0.3787	0.0848
Pitch	0.0005	0.0037	0.0003	0.0031	0.0038	0.0046
Yaw	0.0005	0.0005	0.0009	0.0016	0.0014	0.0005

Table 5-45 Average Composite Parameter for Aileron Failures

Considering the average of the CP in Table 5-45 for all the 5 runs of the aileron failures, there is a reduction of η along roll channel by 58%, 55% and 77% respectively for a soft, medium and hard failure. Along the pitch channel, EMRAN have increased η for the soft, medium and hard failure though this increase is minimal for hard failure. For the yaw channel EMRAN shows an increase for soft and medium failure but a reverse trend for hard failure. It can be concluded that EMRAN greatest contribution is during hard

failures when pilot compensation is not sufficient to maintain the mission profile in a minimum time.

5.3.2.3. FDII Analysis of Aileron Failures

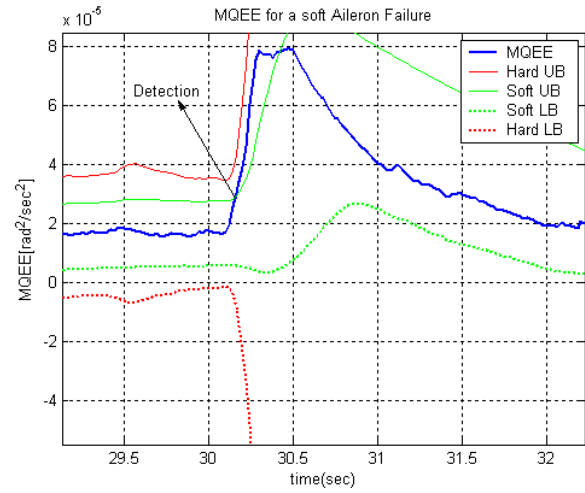
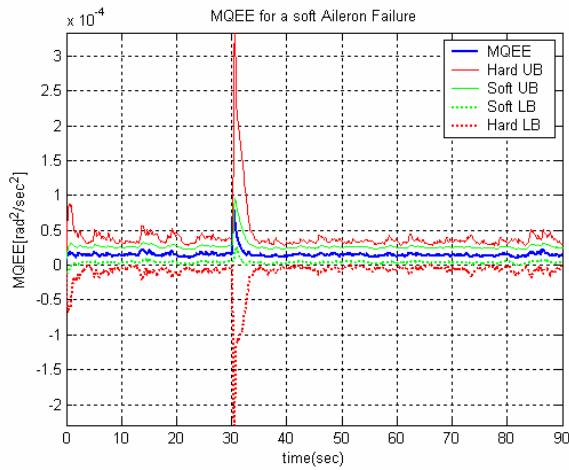


Figure 5-33- Detection of a soft aileron failure

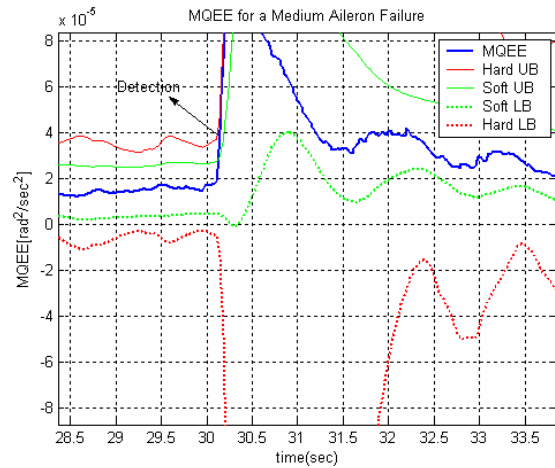
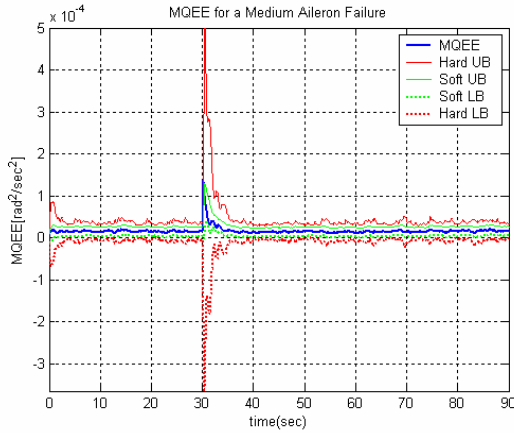


Figure 5-34-Detection of a Medium aileron failure

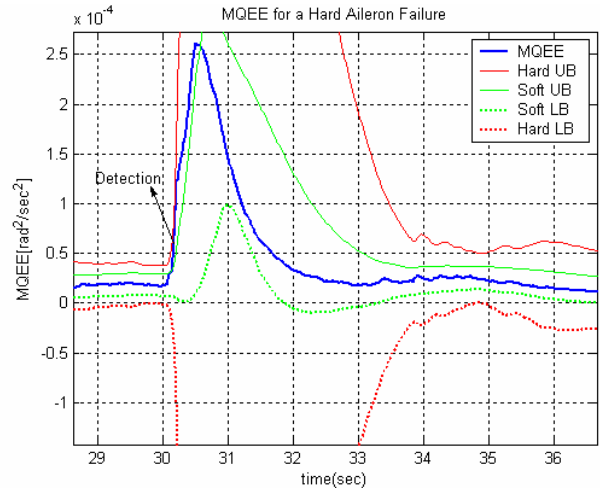
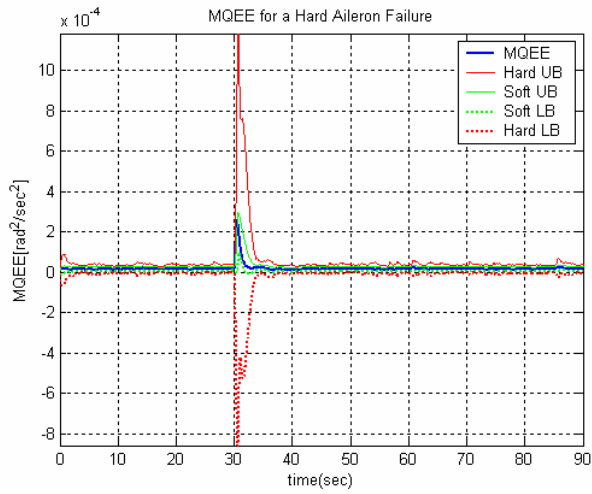


Figure 5-35- Detection of a soft aileron failure

Isolation of Aileron failure

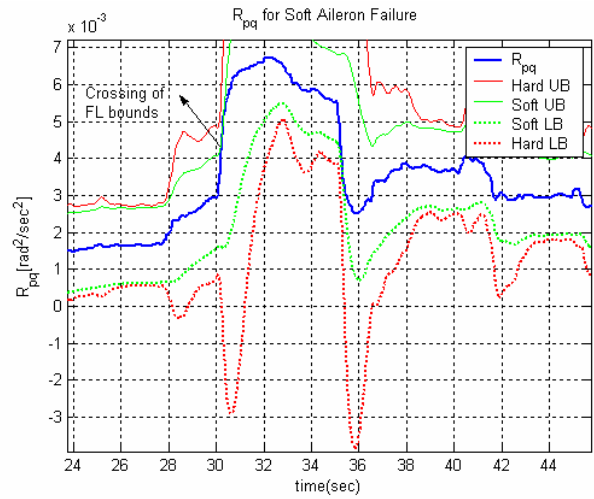
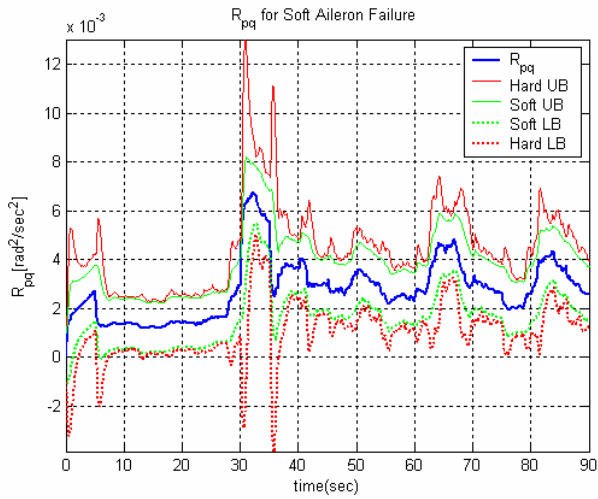


Figure 5-36- Isolation of a soft Aileron Failure

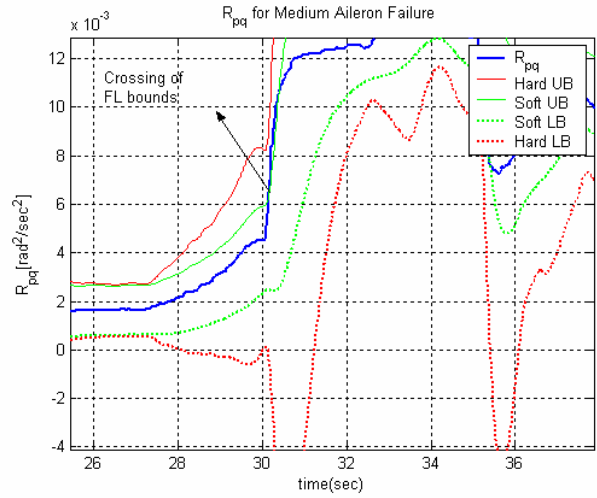
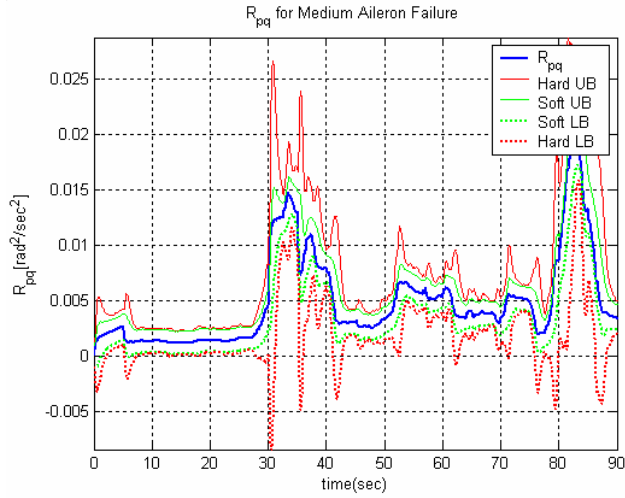


Figure 5-37- Isolation of a Medium Aileron Failure

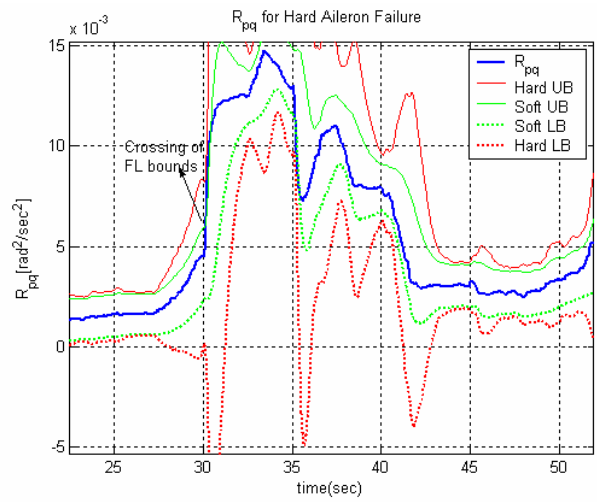
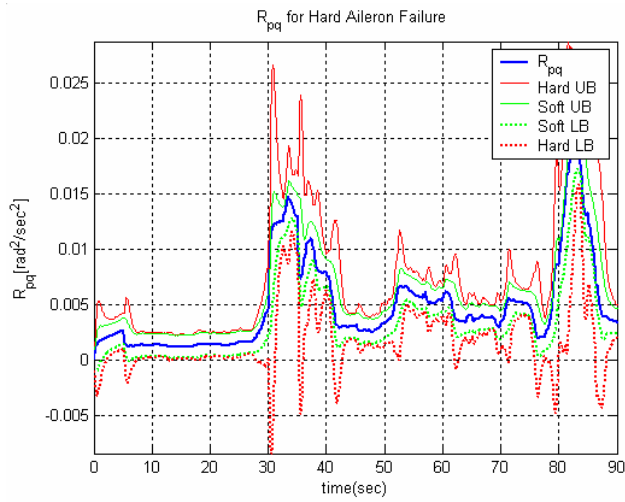


Figure 5-38- Isolation of a Hard Aileron Failure

Identification of Aileron failure

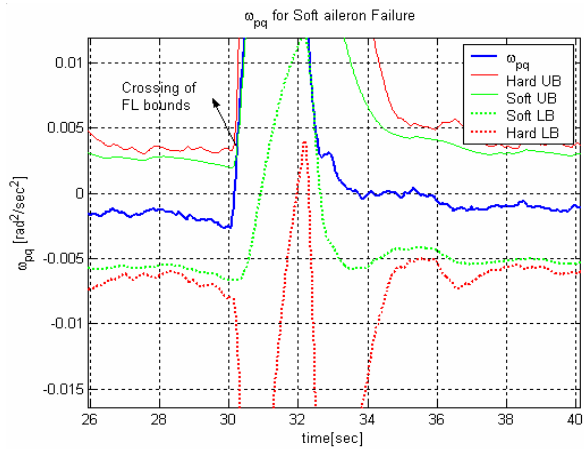
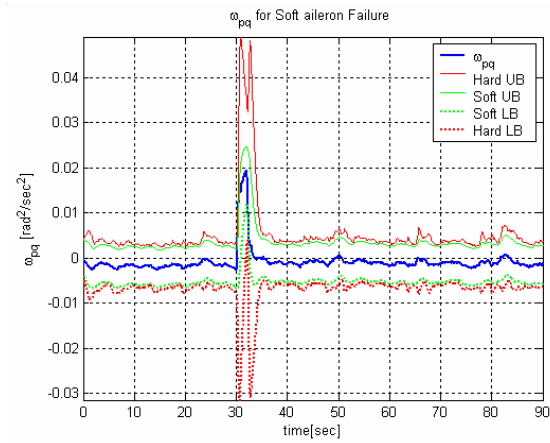


Figure 5-39- Identification of soft Aileron failure with ω_{pq}

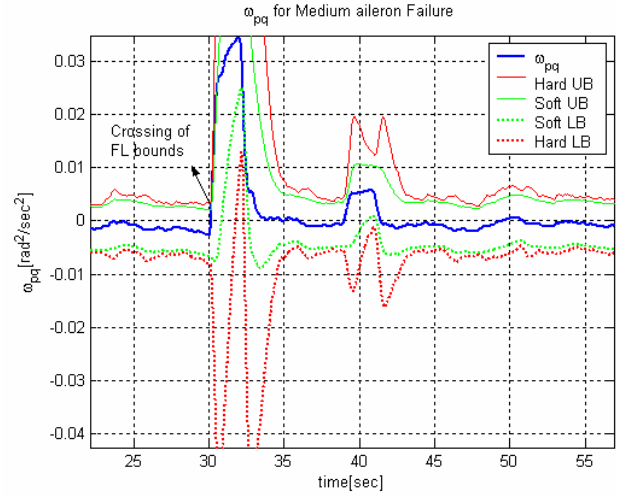
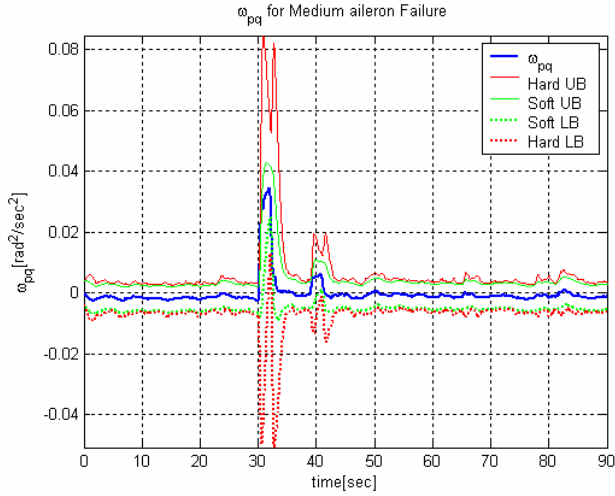


Figure 5-40- Identification of Medium Aileron failure with ω_{pq}

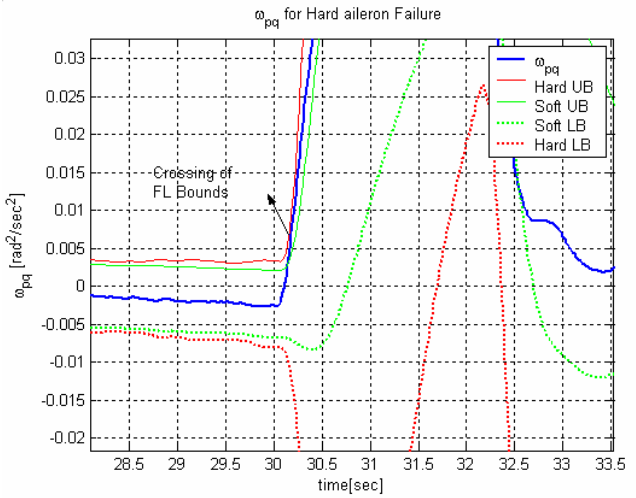
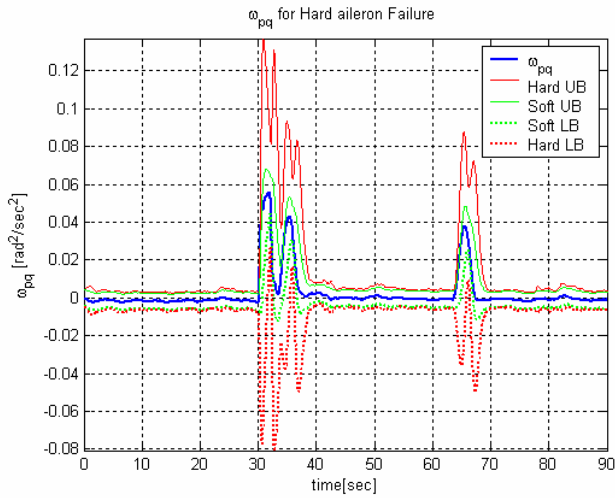


Figure 5-41- Identification of Hard Aileron Failure with ω_{pq}

5.3.3. Analysis of Sensor Failure Cases (Roll Sensor)

5.3.3.1. Tracking Error Analysis

The TE statistics for roll sensor failures for the 5 runs are tabulated below in Table 5-46.

Roll Sensor -Run1

	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Mean Roll	1.4544E-04	-1.7799E-02	-1.6920E-02	-1.5761E-02	-8.5170E-04	-1.6501E-02
Mean Pitch	-1.9340E-03	-1.8064E-03	-1.8212E-03	-1.7971E-03	-1.8371E-03	-1.7666E-03
Mean Yaw	9.8992E-04	7.6659E-04	7.8719E-04	9.4406E-04	6.0538E-04	8.8158E-04
SD Roll	3.7190E-01	6.7338E-01	3.7344E-01	3.9960E-01	5.1856E-01	3.9903E-01
SD Pitch	1.4654E-01	1.4662E-01	1.4648E-01	1.4643E-01	1.4646E-01	1.4643E-01
SD Yaw	1.5566E-01	1.5574E-01	1.5574E-01	1.5562E-01	1.5565E-01	1.5565E-01

Roll Sensor -Run2

Mean Roll	-1.3981E-02	-8.4069E-03	-8.0537E-03	-2.0159E-02	-1.3002E-02	-1.6081E-02
Mean Pitch	-1.8307E-03	-1.9189E-03	-2.0157E-03	-1.9060E-03	-1.9783E-03	-1.8779E-03
Mean Yaw	9.7828E-04	4.8118E-04	7.2537E-04	8.9779E-04	8.8327E-04	9.0807E-04
SD Roll	6.4619E-01	5.8936E-01	6.8990E-01	4.3785E-01	4.1023E-01	3.3896E-01
SD Pitch	1.4649E-01	1.4654E-01	1.4724E-01	1.4646E-01	1.4642E-01	1.4642E-01
SD Yaw	1.5569E-01	1.5580E-01	1.5618E-01	1.5569E-01	1.5568E-01	1.5563E-01

Roll Sensor -Run3

Mean Roll	-1.1165E-02	-1.4222E-02	-1.8597E-02	-2.2246E-02	-1.7915E-02	-7.3550E-03
Mean Pitch	-1.8173E-03	-1.7986E-03	-1.8074E-03	-1.8008E-03	-1.7877E-03	-1.9331E-03
Mean Yaw	9.5335E-04	9.7666E-04	7.2640E-04	7.7909E-04	7.4074E-04	5.2124E-04
SD Roll	3.8061E-01	4.2633E-01	4.4934E-01	4.5874E-01	4.5161E-01	4.5668E-01
SD Pitch	1.4644E-01	1.4650E-01	1.4645E-01	1.4643E-01	1.4644E-01	1.4639E-01
SD Yaw	1.5561E-01	1.5562E-01	1.5574E-01	1.5569E-01	1.5569E-01	1.5570E-01

Roll Sensor -Run4

Mean Roll	-2.6271E-02	-5.4341E-03	-1.6300E-02	-1.7626E-02	-2.1366E-02	-7.7890E-03
Mean Pitch	-1.8490E-03	-1.9416E-03	-1.8343E-03	-1.8264E-03	-1.8745E-03	-1.9378E-03
Mean Yaw	8.1100E-04	6.8564E-04	8.3262E-04	7.3165E-04	7.5095E-04	7.9093E-04
SD Roll	4.2492E-01	5.1931E-01	4.8534E-01	5.6613E-01	4.7266E-01	4.1448E-01
SD Pitch	1.4645E-01	1.4655E-01	1.4642E-01	1.4644E-01	1.4641E-01	1.4642E-01
SD Yaw	1.5571E-01	1.5566E-01	1.5570E-01	1.5584E-01	1.5587E-01	1.5569E-01

Roll Sensor -Run5

Mean Roll	-1.5021E-02	-1.7392E-02	-1.3256E-02	-1.2203E-02	3.3891E-03	-1.0074E-02
Mean Pitch	-1.7812E-03	-1.9111E-03	-1.8888E-03	-1.8888E-03	-1.8584E-03	-1.7533E-03
Mean Yaw	6.6255E-04	7.6744E-04	7.1986E-04	7.4291E-04	4.4932E-04	6.9476E-04
SD Roll	4.0513E-01	3.4697E-01	4.7920E-01	4.1507E-01	4.4880E-01	5.3449E-01
SD Pitch	1.4645E-01	1.4653E-01	1.4642E-01	1.4643E-01	1.4644E-01	1.4645E-01
SD Yaw	1.5564E-01	1.5566E-01	1.5567E-01	1.5561E-01	1.5565E-01	1.5570E-01

Table 5-46 Tracking Error (TE) statistics for Roll Sensor Failures**5.3.3.2. Stick Activity Roll Sensor Failures****Run1**

Stick Activity	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Lateral	0.9125	1.0010	0.9839	0.8945	0.9125	0.9219
Longitudinal	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Run2

Lateral	1.0770	0.9013	0.9307	0.9527	1.0770	0.9674
Longitudinal	0.0000	0.0000	0.2348	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Run3

Lateral	1.0781	1.1434	0.8290	1.0004	1.0781	0.7446
Longitudinal	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Run4

Lateral	1.0629	0.8617	0.8185	0.7821	1.0629	0.9983
Longitudinal	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Run5

Lateral	1.0358	0.8168	0.8349	0.8185	1.0358	1.0075
Longitudinal	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-47 Stick Activity for Roll Sensor Failures

Considering the average of stick activity for the roll sensor failures, there is little or no activity along the pitch and yaw channels. The pilot compensation is of the same order in all the test cases. Clearly, there is no significant workload on the pilot on the onset of failures however; this small activity is because of the instrument readings in the cockpit. The sensor failure induces a small shift in the horizon indicator and pilot thinks that it is a failure. There is no motion cue for the pilot in case of sensor failures but the visual cue change at the onset of sensor failure. In order to bring the instrument reading to level pilot does a little activity which is typically the stick activity shown in tables.

Stick Activity	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Lateral	1.0333	0.9448	0.8794	0.8896	1.0333	0.9279
Longitudinal	0.0000	0.0000	0.0470	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-48 Average Stick Activity for Roll Sensor Failures

5.3.3.3. FDII Analysis of Roll Sensor Failures

The piloted tests in the flight simulator were performed for a roll sensor failure and Table 5-49- Table 5-53 shows the FDII results in terms of detection time, isolation time and identification time. The FDII has shown to detect, isolate and identify the roll sensor failures accurately and there are no cases of false detections, isolations or identifications.

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Roll Sensor SFDB	31.92	32.92	33.98	T
2	Roll Sensor LFDB	31.3	31.3	36.3	T
3	Roll Sensor SMDB	30.38	31.38	31.92	T
4	Roll Sensor LMDB	31.88	32.88	33.98	T
5	Roll Sensor SSDB	30.38	31.38	31.92	T
6	Roll Sensor LSDB	31.88	32.38	33.98	T

Table 5-49 FDI results for Roll Sensor Failure (Run1)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Roll Sensor SFDB	31.92	32.92	33.98	T
2	Roll Sensor LFDB	31.3	32.3	33.9	T
3	Roll Sensor SMDB	30.38	31.38	31.92	T
4	Roll Sensor LMDB	31.9	32.92	33.98	T
5	Roll Sensor SSDB	30.38	31.38	31.90	T
6	Roll Sensor LSDB	30.38	31.38	31.92	T

Table 5-50 FDI results for Roll Sensor Failure (Run2)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Roll Sensor SFDB	31.92	32.92	33.98	T
2	Roll Sensor LFDB	31.3	32.3	45.26	T
3	Roll Sensor SMDB	30.38	31.38	31.92	T
4	Roll Sensor LMDB	31.9	32.92	33.98	T
5	Roll Sensor SSDB	30.38	31.38	31.92	T
6	Roll Sensor LSDB	31.88	32.88	33.98	T

Table 5-51 FDI results for Roll Sensor Failure (Run3)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Roll Sensor SFDB	31.92	32.92	33.98	T
2	Roll Sensor LFDB	31.3	32.3	33.82	T
3	Roll Sensor SMDB	30.38	31.38	31.92	T
4	Roll Sensor LMDB	31.9	32.92	33.98	T
5	Roll Sensor SSDB	30.38	31.38	31.92	T
6	Roll Sensor LSDB	31.88	32.88	33.98	T

Table 5-52 FDI results for Roll Sensor Failure (Run4)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Roll Sensor SFDB	31.92	32.92	33.98	T
2	Roll Sensor LFDB	31.3	32.3	33.82	T
3	Roll Sensor SMDB	30.38	31.38	31.92	T
4	Roll Sensor LMDB	31.9	32.92	33.98	T
5	Roll Sensor SSDB	30.38	31.38	31.92	T
6	Roll Sensor LSDB	31.08	32.88	33.96	T

Table 5-53 FDI results for Roll Sensor Failure (Run5)

The detection phase consists of monitoring $MQEE$ and $OQEE$. It can be seen from Figure 5-42 that there is no detection with $MQEE$ because of its low sensitivity to sensor failures. On the other hand $OQEE$ detects the roll sensor failure for all the six different cases and is shown in Figure 5-43. A roll sensor failure is identified in all the test cases.

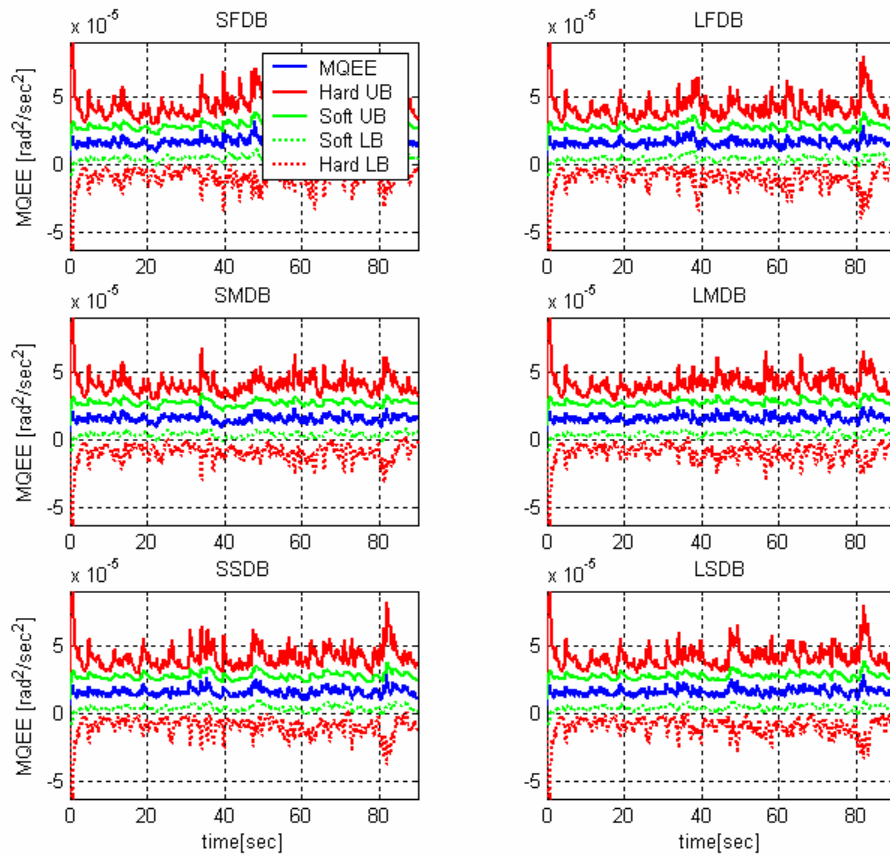


Figure 5-42- MQEE for a Roll Sensor Failure

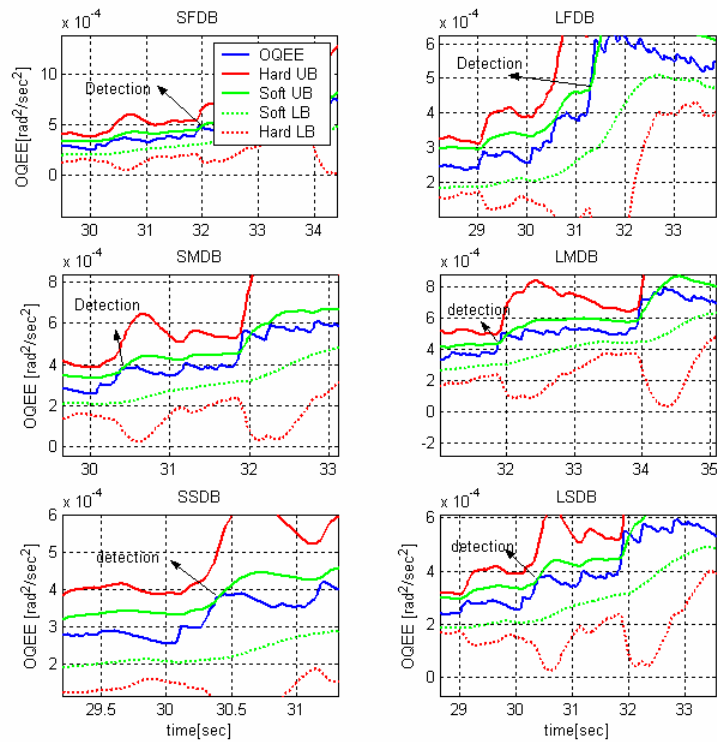


Figure 5-43- QGEE for Detection (Plots Zoomed to show detections)

R_{pq} is used for isolation of failure and due to its slow rise for roll sensor failure, R_{pq} doesn't cross the $R_{pq_floating}$ bounds. Considering the time Δt_1 after detection of the failure for R_{pq} to build up, the sensor failure is isolated. Once the sensor failure is isolated the identification logic monitors the $DQEE$ parameters along roll, pitch and yaw channels. Figure 5-46-Figure 5-50 shows $DQEE$ along the three channels for the six different cases of the sensor biases. The identification is accurate for all the test cases.

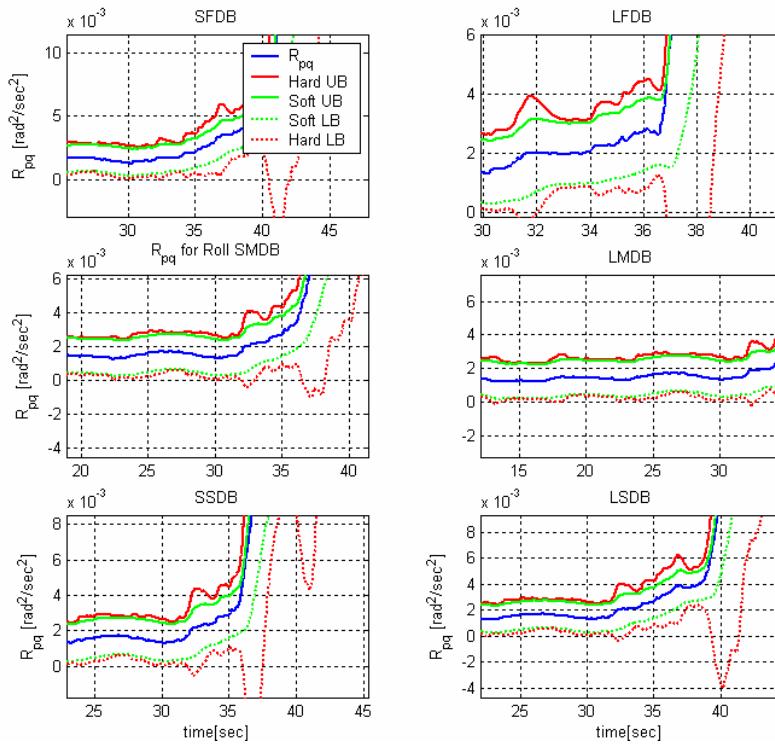


Figure 5-44- No crossing of Bounds with R_{pq} isolates sensor failure

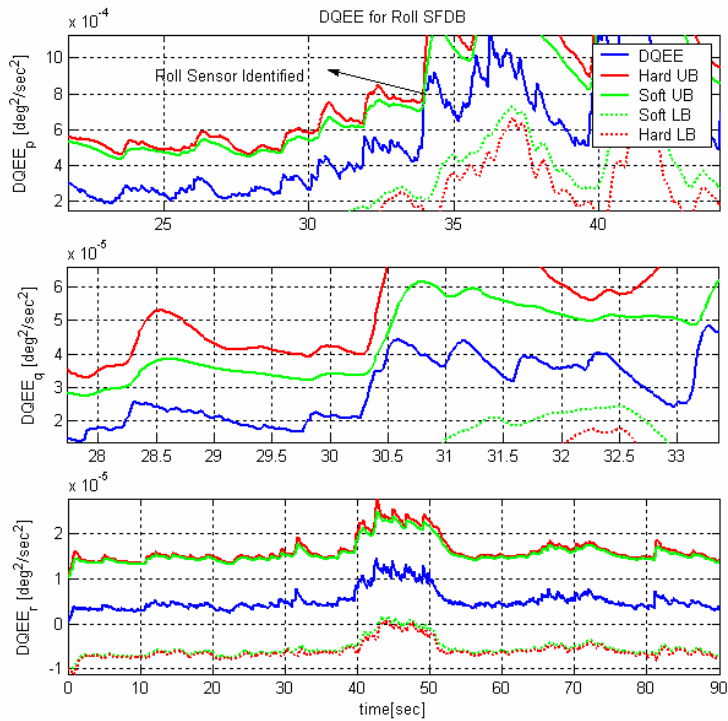


Figure 5-45- Identification of Roll Sensor for SFDB (SF #1)

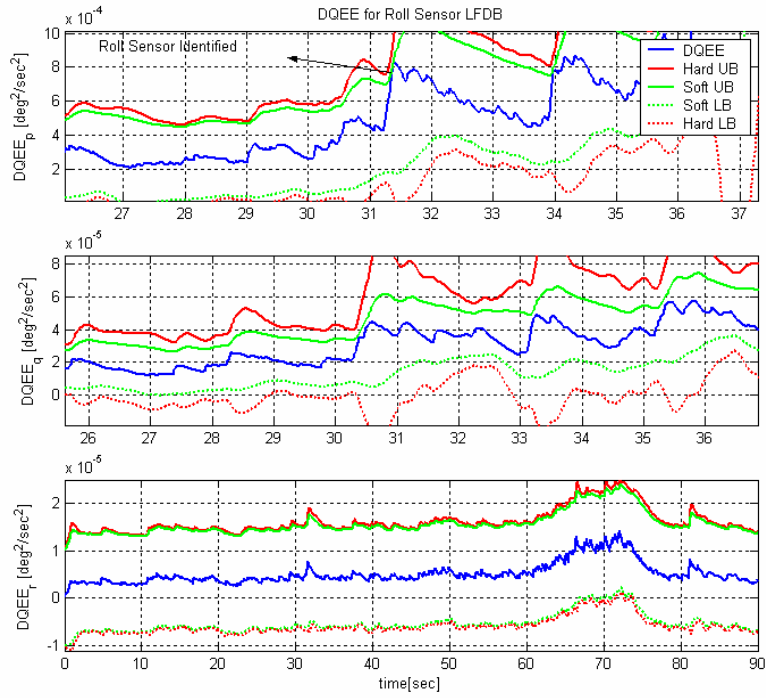


Figure 5-46- Identification of Roll Sensor for LFDB (SF #2)

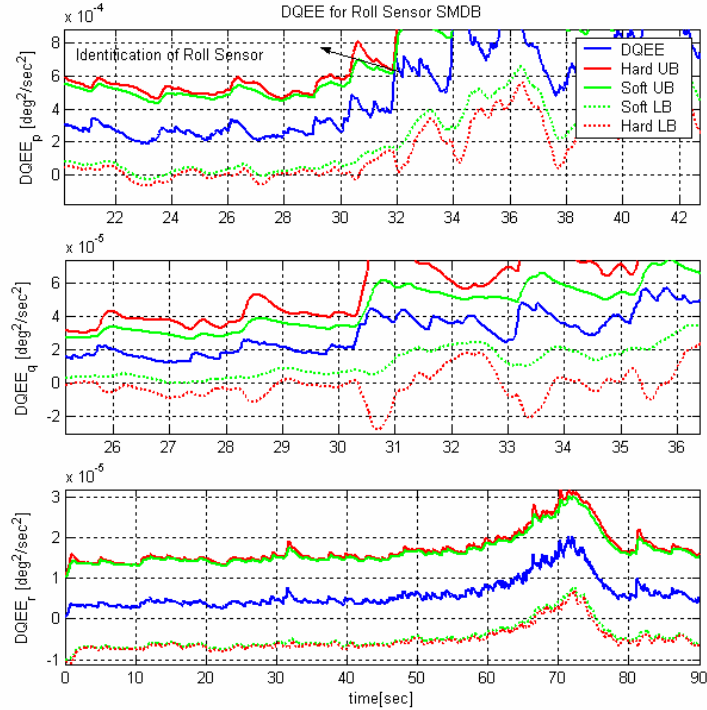


Figure 5-47- Identification of Roll Sensor for SMDB (SF #3)

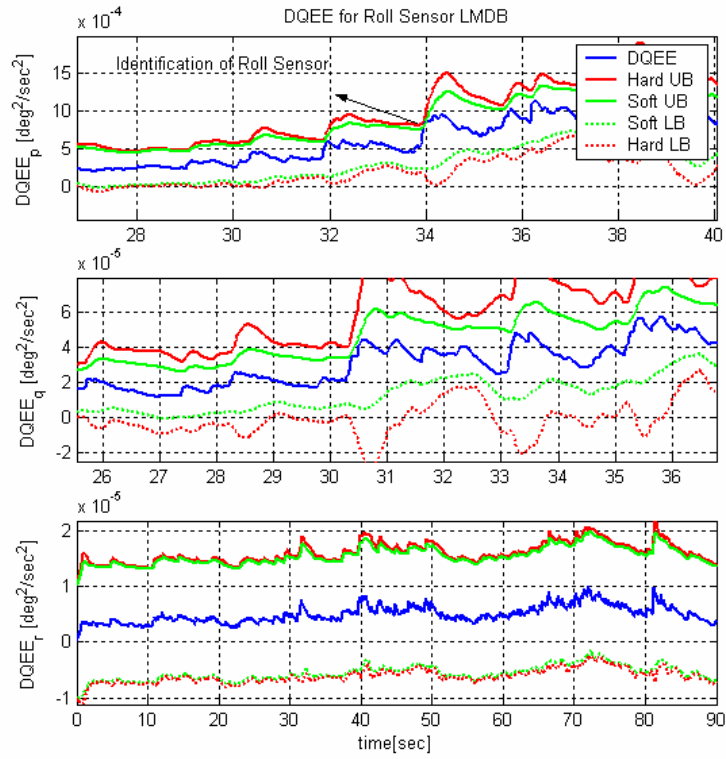


Figure 5-48- Identification of Roll Sensor for LMDB (SF #4)

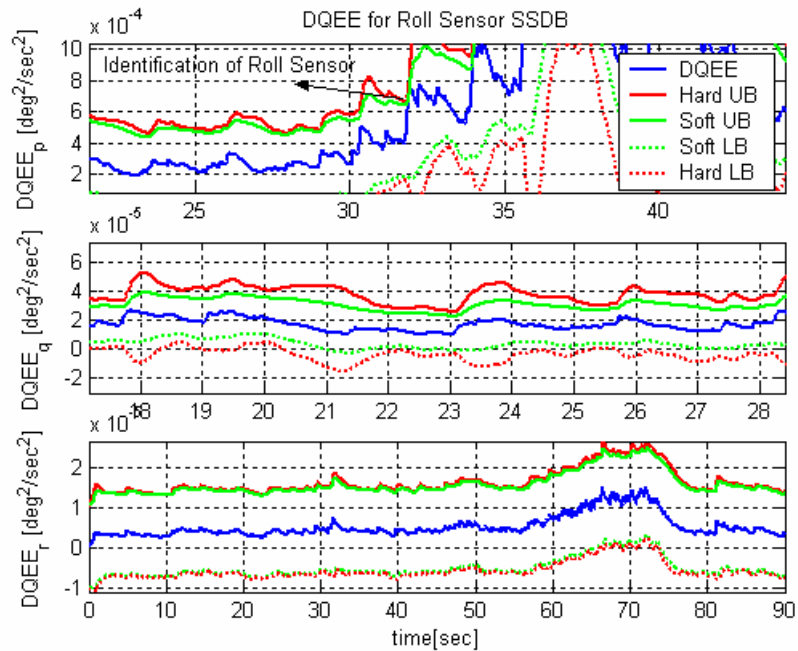


Figure 5-49- Identification of Roll Sensor for SSDB (SF #5)

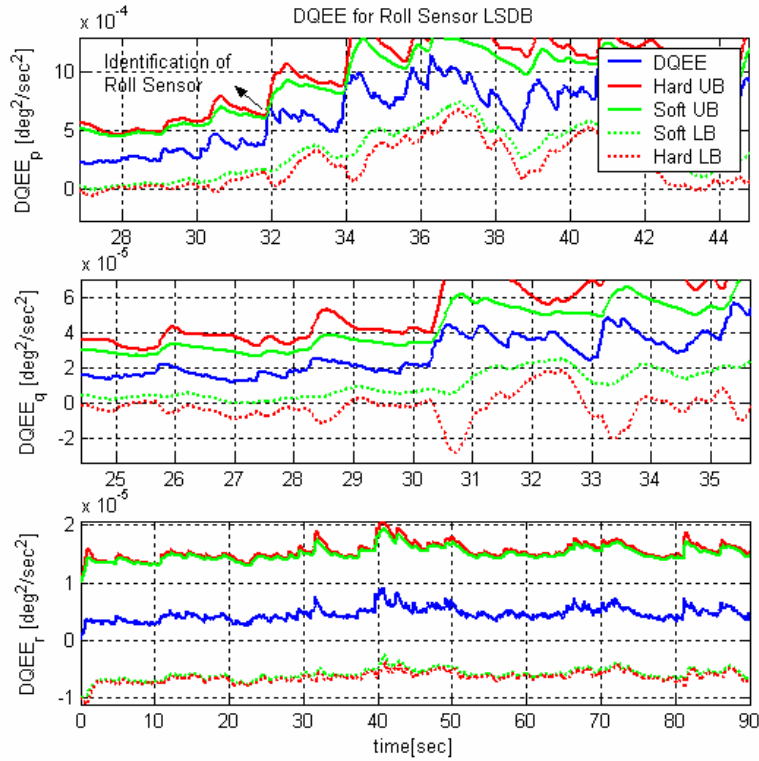


Figure 5-50- Identification of Roll Sensor for LSDB (SF #6)

5.3.4. Analysis of Sensor Failure Cases (Pitch Sensor)

5.3.4.1. Tracking Error Analysis

Pitch Sensor -Run1

	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Mean Roll	9.0036E-05	6.0896E-04	4.2043E-05	9.6828E-06	-4.3943E-04	-2.1842E-04
Mean Pitch	-2.1606E-03	-6.2095E-03	-1.9742E-03	-3.1508E-03	3.5100E-04	-3.6489E-04
Mean Yaw	4.8937E-04	5.5042E-04	4.8006E-04	4.7660E-04	4.4197E-04	4.5988E-04
SD Roll	2.0407E-01	2.0428E-01	2.0402E-01	2.0408E-01	2.0414E-01	2.0403E-01
SD Pitch	1.8509E-01	5.2300E-01	1.3444E-01	1.5942E-01	1.2817E-01	1.3937E-01
SD Yaw	1.5570E-01	1.5575E-01	1.5570E-01	1.5570E-01	1.5570E-01	1.5570E-01

Pitch Sensor -Run2

Mean Roll	-8.7510E-06	5.5668E-04	-1.4299E-04	-2.5965E-04	5.5726E-05	3.4254E-04
Mean Pitch	-1.9799E-03	-4.6385E-03	-7.0681E-04	-7.8591E-04	-1.9935E-03	-5.9891E-03
Mean Yaw	4.7653E-04	5.3986E-04	4.6484E-04	4.5812E-04	4.8126E-04	4.9809E-04
SD Roll	2.0410E-01	2.0478E-01	2.0414E-01	2.0409E-01	2.0400E-01	2.0415E-01
SD Pitch	1.8460E-01	4.3062E-01	1.3464E-01	1.5857E-01	1.2678E-01	1.4158E-01
SD Yaw	1.5571E-01	1.5578E-01	1.5570E-01	1.5570E-01	1.5570E-01	1.5570E-01

Pitch Sensor -Run3

Mean Roll	1.0105E-04	4.4215E-04	3.8581E-05	1.3771E-04	1.0349E-05	-8.3276E-04
Mean Pitch	-2.1559E-03	1.0287E-03	-1.9741E-03	-2.4602E-03	-1.3941E-03	4.2695E-04
Mean Yaw	4.8801E-04	4.2418E-04	4.8039E-04	5.0504E-04	4.7531E-04	4.0965E-04
SD Roll	2.0405E-01	2.0923E-01	2.0403E-01	2.0480E-01	2.0421E-01	2.0414E-01
SD Pitch	1.8499E-01	4.3140E-01	1.3447E-01	1.7015E-01	1.2511E-01	1.4475E-01
SD Yaw	1.5570E-01	1.5713E-01	1.5570E-01	1.5592E-01	1.5569E-01	1.5569E-01

Pitch Sensor -Run4

Mean Roll	-5.9238E-05	1.6375E-03	-9.5945E-04	4.5775E-04	1.1304E-04	2.4406E-04
Mean Pitch	-7.6189E-04	-2.1298E-02	1.4583E-03	-4.6523E-03	-2.3883E-03	-4.6560E-03
Mean Yaw	4.7618E-04	5.8155E-04	3.9907E-04	5.1042E-04	4.8908E-04	4.9325E-04
SD Roll	2.0406E-01	2.0408E-01	2.0421E-01	2.0395E-01	2.0402E-01	2.0415E-01
SD Pitch	1.8536E-01	4.8194E-01	1.3386E-01	1.6545E-01	1.3566E-01	1.3949E-01
SD Yaw	1.5570E-01	1.5582E-01	1.5569E-01	1.5569E-01	1.5570E-01	1.5570E-01

Pitch Sensor -Run5

Mean Roll	6.3375E-05	3.1401E-04	1.1177E-04	-1.0990E-04	1.1554E-04	2.3651E-04
Mean Pitch	-1.8984E-03	-2.8262E-04	-2.5161E-03	-6.0421E-04	5.4420E-05	-2.5429E-03
Mean Yaw	4.8420E-04	5.2992E-04	4.8680E-04	4.7096E-04	4.8129E-04	4.9450E-04
SD Roll	2.0406E-01	2.0361E-01	2.0427E-01	2.0438E-01	2.0412E-01	2.0413E-01
SD Pitch	1.9066E-01	5.0708E-01	1.4116E-01	1.6295E-01	1.3127E-01	1.3886E-01
SD Yaw	1.5571E-01	1.5576E-01	1.5571E-01	1.5570E-01	1.5569E-01	1.5570E-01

Table 5-54 Tracking Error (TE) statistics of Pitch Sensor Failures

5.3.4.2. Stick Activity Analysis

The stick activity for pitch sensor failures is shown in Table 5-55. There is a negligible activity along lateral and directional channel for all cases. The hardest failure (SF#2) has maximum pilot activity in all the test cases. For all other cases the pilot activity is of the same order.

Pitch Sensor -Run1						
Stick Activity	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Longitudinal	0.1007	11.2141	0.0029	0.8938	0.9190	0.7252
Lateral	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0019	0.0000	0.0000
Pitch Sensor -Run2						
Longitudinal	1.1362	4.2131	0.6610	0.5647	0.2493	0.6178
Lateral	0.0000	0.0000	0.0000	0.0000	0.0305	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Pitch Sensor -Run3						
Longitudinal	0.6846	5.4022	1.0190	0.3033	1.2796	0.7927
Lateral	0.0000	0.0065	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Pitch Sensor -Run4						
Longitudinal	1.1865	7.5150	0.9677	0.7743	1.1865	0.8103
Lateral	0.0000	0.0200	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Pitch Sensor -Run5						
Longitudinal	1.1865	7.5150	0.9677	0.7743	0.7898	0.8103
Lateral	0.0000	0.0200	0.0000	0.0000	0.0000	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-55 Stick Activity for Pitch Sensor Failures

Stick Activity	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Longitudinal	0.6278	8.5067	1.2249	0.8744	0.8714	1.0737
Lateral	0.0000	0.0393	0.0000	0.0000	0.0027	0.0000
Directional	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 5-56 Average Stick Activity for Pitch Sensor Failures

5.3.4.3. FDII Analysis of pitch sensor failures

The pitch sensor failure FDII results are tabulated in Table 5-57 - Table 5-61 . In this case as well the results are accurate and there are no cases of wrong detection, isolation and identification.

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Pitch Sensor SFDB	30.34	31.34	31.36	T
2	Pitch Sensor LFDB	30.36	31.34	31.36	T
3	Pitch Sensor SMDB	30.36	31.36	45.28	T
4	Pitch Sensor LMDB	30.36	31.36	31.38	T
5	Pitch Sensor SSDB	30.36	31.36	40.82	T
6	Pitch Sensor LSDB	30.36	31.36	52.68	T

Table 5-57 FDII results for Pitch Sensor Failure (Run1)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Pitch Sensor SFDB	30.34	31.34	31.36	T
2	Pitch Sensor LFDB	30.34	31.34	31.36	T
3	Pitch Sensor SMDB	30.36	31.36	54.7	T
4	Pitch Sensor LMDB	30.36	31.36	31.38	T
5	Pitch Sensor SSDB	30.36	31.36	45.26	T
6	Pitch Sensor LSDB	30.36	31.36	54.2	T

Table 5-58 FDII results for Pitch Sensor Failure (Run2)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Pitch Sensor SFDB	30.34	31.34	31.36	T
2	Pitch Sensor LFDB	30.34	31.34	31.36	T
3	Pitch Sensor SMDB	30.36	31.36	44.84	T
4	Pitch Sensor LMDB	30.34	31.34	31.36	T
5	Pitch Sensor SSDB	30.36	31.36	45.12	T
6	Pitch Sensor LSDB	30.36	31.36	38.64	T

Table 5-59 FDII results for Pitch Sensor Failure (Run3)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Pitch Sensor SFDB	30.34	31.34	31.36	T
2	Pitch Sensor LFDB	30.34	31.34	31.36	T
3	Pitch Sensor SMDB	30.36	31.36	38.54	T
4	Pitch Sensor LMDB	30.36	31.36	31.38	T
5	Pitch Sensor SSDB	30.36	31.36	61.14	T
6	Pitch Sensor LSDB	30.36	31.36	44.96	T

Table 5-60 FDII results for Pitch Sensor Failure (Run4)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Pitch Sensor SFDB	30.34	31.34	31.36	T
2	Pitch Sensor LFDB	30.34	31.34	31.36	T
3	Pitch Sensor SMDB	30.36	31.36	43.78	T
4	Pitch Sensor LMDB	30.36	31.36	31.38	T
5	Pitch Sensor SSDB	30.36	31.36	31.38	T
6	Pitch Sensor LSDB	30.36	31.36	37.82	T

Table 5-61 FDII results for Pitch Sensor Failure (Run5)

Figure 5-51 shows *MQEE* for the pitch sensor failure. It can be seen that it is insensitive to the failure except the LFDB which is the most severe sensor failure among the six cases. *OQEE* has correctly detected the pitch sensor failure and is shown in Figure 5-52. The areas in the plots are zoomed near the failure time to show the time at which detection occurs.

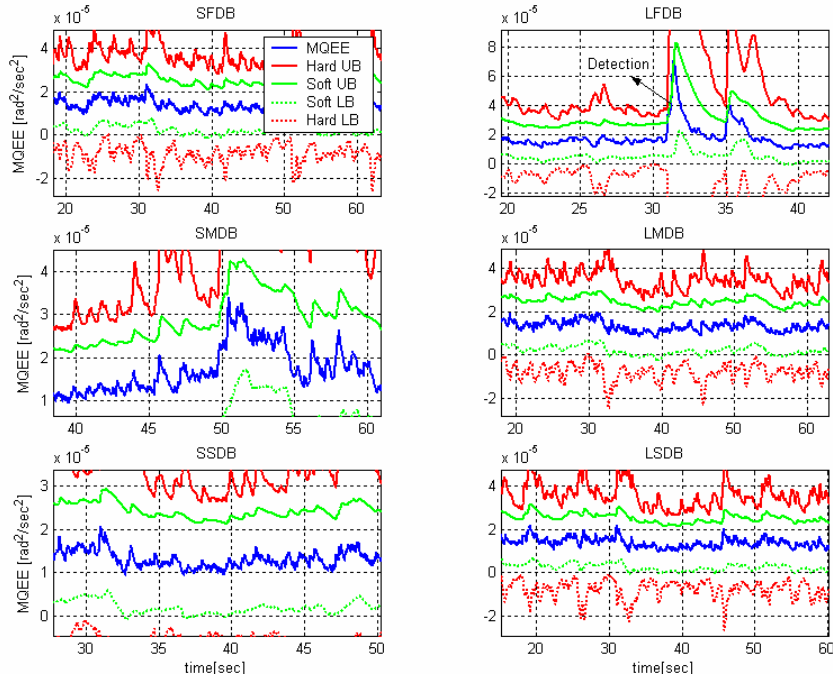


Figure 5-51- MQEE for Pitch Sensor Failure

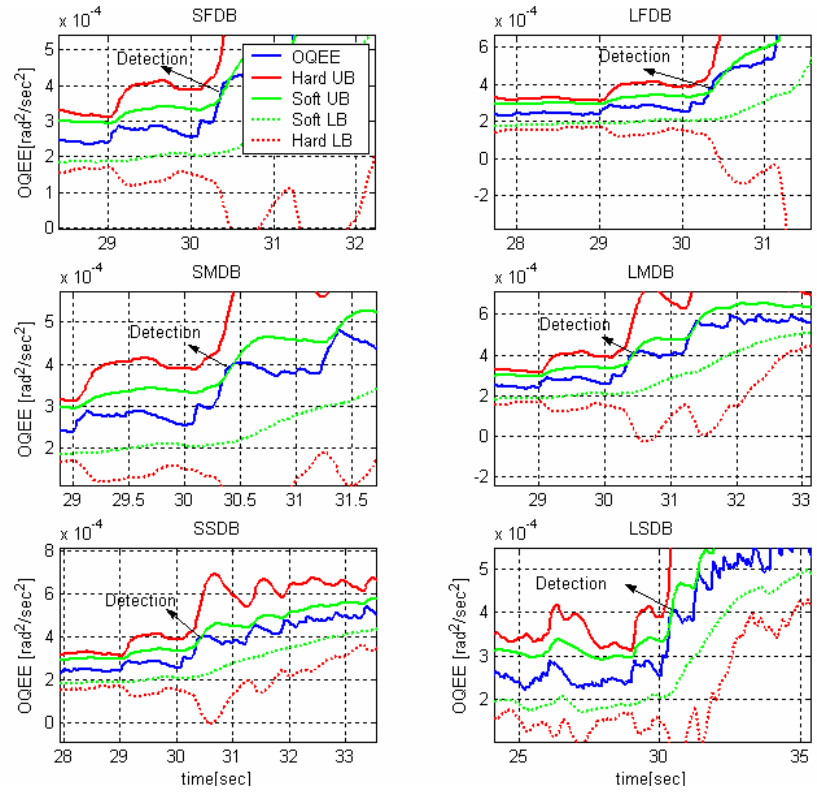


Figure 5-52- OQEE for Pitch Sensor failure (Zoomed to show detections)

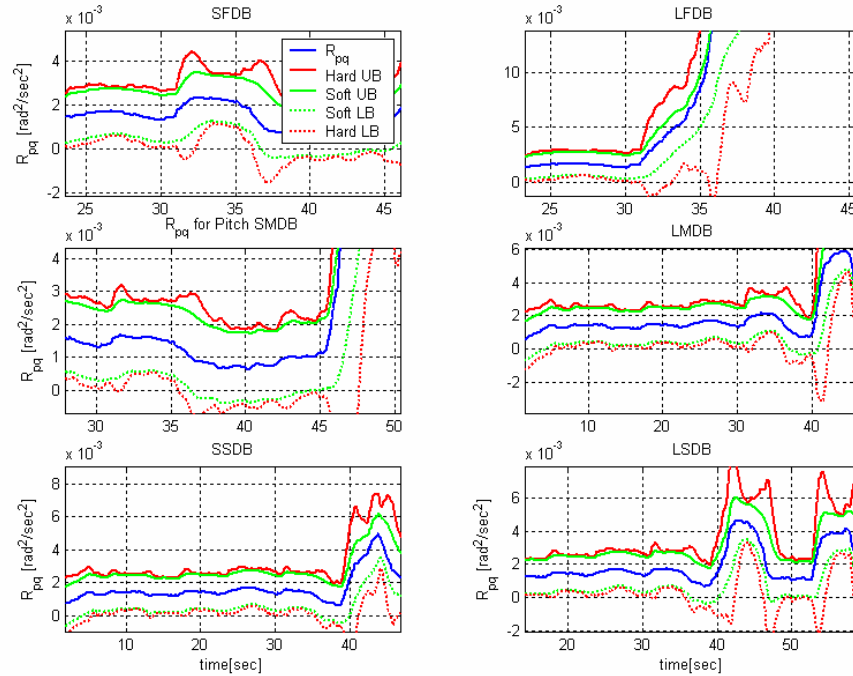


Figure 5-53 R_{pq} monitored to isolate the sensor failure

Figure 5-53 shows the typical variation of R_{pq} for the pitch sensor failure cases and it can be seen that FL bounds are not crossed for nearly 5 sec after the failure is detected. The flag for the sensor failure isolation is thus triggered in the FDII warning system algorithm. The FDII algorithm switches the logic to sensor failure identification by monitoring the $DQEE$

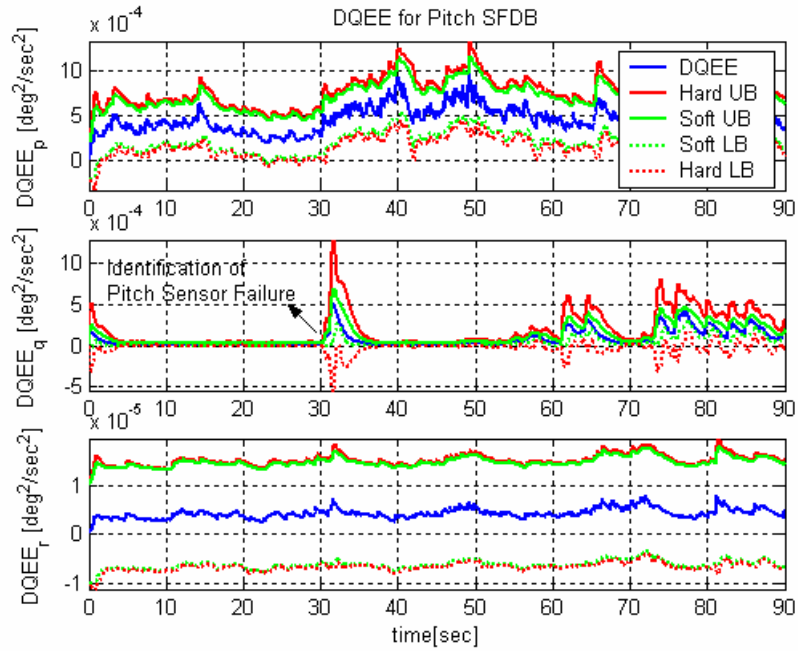


Figure 5-54 Identification of Pitch Sensor failure (SF#1)

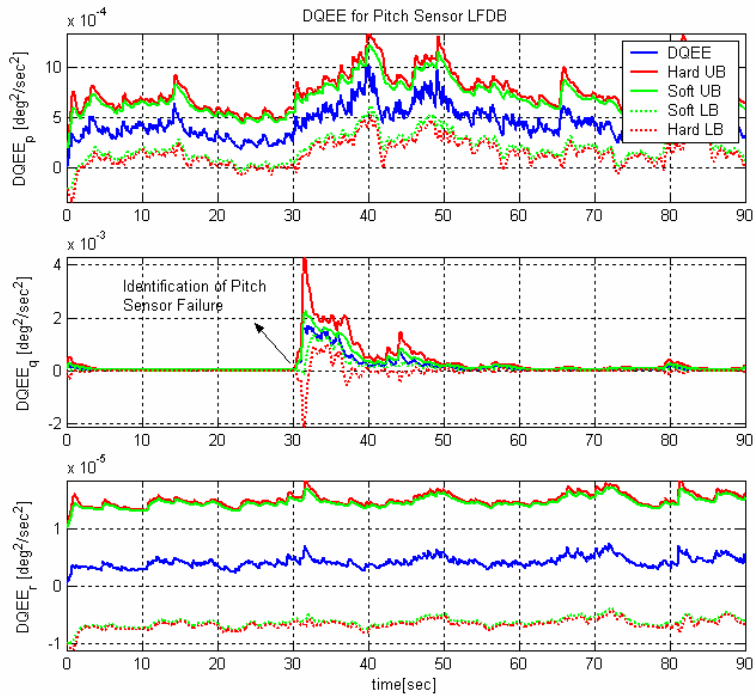


Figure 5-55 Identification of Pitch Sensor Failure (SF#2)

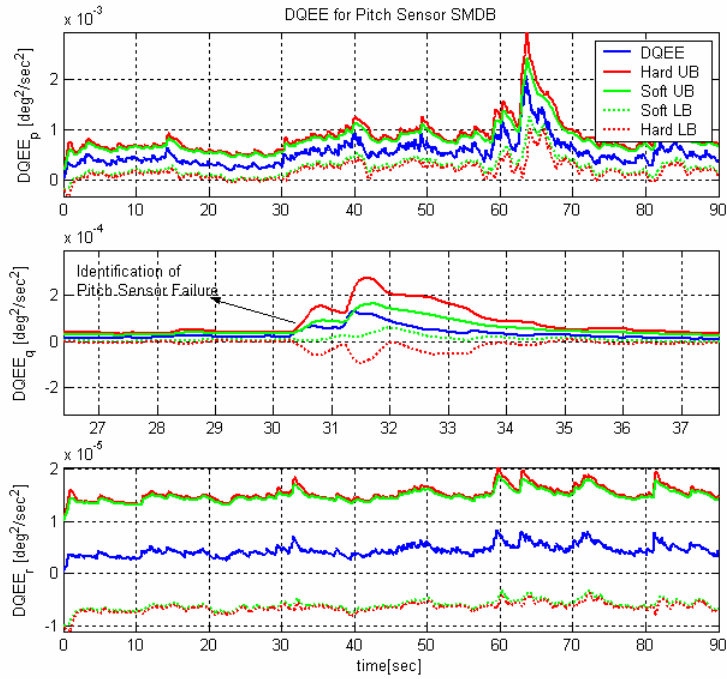


Figure 5-56 Identification of Pitch Sensor Failure (SF#3)

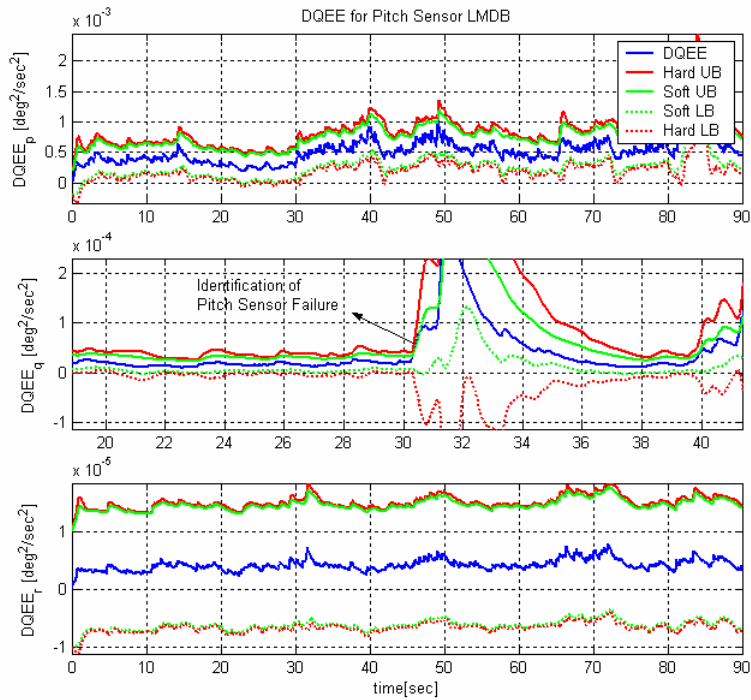


Figure 5-57 Identification of Pitch Sensor Failure (SF#4)

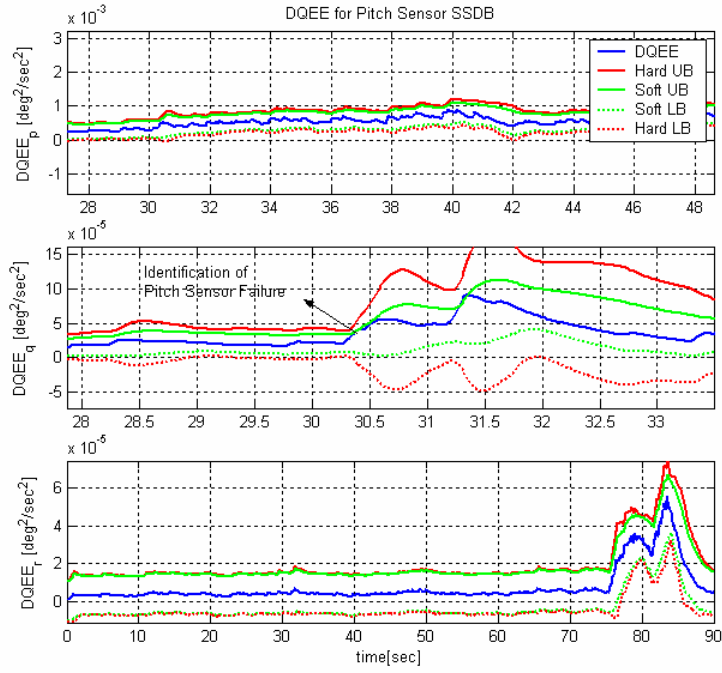


Figure 5-58 Identification of Pitch Sensor Failure (SF#5)

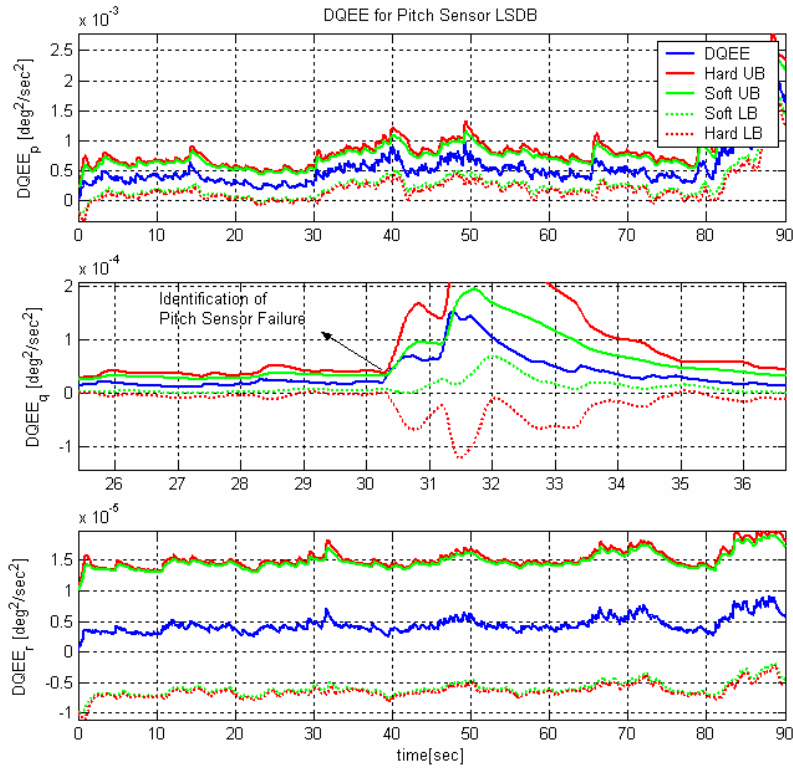


Figure 5-59 Identification of Pitch Sensor Failure (SF #6)

5.3.5. Analysis of Sensor Failure Cases (Yaw Sensor)

5.3.5.1. Tracking Error Analysis

Yaw Sensor -Run1

	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Mean Roll	-1.6421E-03	-2.1341E-04	-1.4759E-03	-6.7301E-03	-1.4760E-03	1.6956E-03
Mean Pitch	-1.6421E-03	-2.1341E-04	-1.4759E-03	-6.7301E-03	-1.4760E-03	1.6956E-03
Mean Yaw	-1.1979E+00	-2.4136E+00	-9.3007E-01	-1.8499E+00	-6.6375E-01	-1.3262E+00
SD Roll	1.4298E-01	1.5636E-01	1.4294E-01	1.4547E-01	1.4294E-01	1.4249E-01
SD Pitch	1.4298E-01	1.5636E-01	1.4294E-01	1.4547E-01	1.4294E-01	1.4249E-01
SD Yaw	9.4791E-01	1.8981E+00	8.6748E-01	1.7131E+00	6.8316E-01	1.3403E+00

Yaw Sensor -Run2

Mean Roll	-1.2871E-02	-1.7740E-03	-4.4153E-04	-1.1104E-03	-9.6764E-04	-2.4882E-03
Mean Pitch	-1.2871E-02	-1.7740E-03	-4.4153E-04	-1.1104E-03	-9.6764E-04	-2.4882E-03
Mean Yaw	-1.1732E+00	-2.4010E+00	-9.2860E-01	-1.8535E+00	-6.6318E-01	-1.3348E+00
SD Roll	1.4767E-01	1.4740E-01	1.4356E-01	1.5010E-01	1.4432E-01	1.4459E-01
SD Pitch	1.4767E-01	1.4740E-01	1.4356E-01	1.5010E-01	1.4432E-01	1.4459E-01
SD Yaw	9.5050E-01	1.8604E+00	8.7416E-01	1.7253E+00	6.8619E-01	1.3306E+00

Yaw Sensor -Run3

Mean Roll	-6.7007E-04	-1.1680E-03	9.5473E-03	4.9286E-04	-1.1467E-03	-1.7731E-03
Mean Pitch	-6.7007E-04	-1.1680E-03	9.5473E-03	4.9286E-04	-1.1467E-03	-1.7731E-03
Mean Yaw	-1.1941E+00	-2.3919E+00	-9.5984E-01	-1.8544E+00	-6.6150E-01	-1.3305E+00
SD Roll	1.4319E-01	1.4311E-01	1.4949E-01	1.4390E-01	1.4328E-01	1.4296E-01
SD Pitch	1.4319E-01	1.4311E-01	1.4949E-01	1.4390E-01	1.4328E-01	1.4296E-01
SD Yaw	9.5086E-01	1.8785E+00	8.6194E-01	1.7177E+00	6.8263E-01	1.3422E+00

Yaw Sensor -Run4

Mean Roll	-2.3289E-03	-1.9481E-03	-1.3874E-03	-1.1291E-03	-1.8545E-03	-3.9876E-04
Mean Pitch	-2.3289E-03	-1.9481E-03	-1.3874E-03	-1.1291E-03	-1.8545E-03	-3.9876E-04
Mean Yaw	-1.1997E+00	-2.3930E+00	-9.2967E-01	-1.8599E+00	-6.6560E-01	-1.3296E+00
SD Roll	1.4316E-01	1.4375E-01	1.4314E-01	1.4326E-01	1.4330E-01	1.4343E-01
SD Pitch	1.4316E-01	1.4375E-01	1.4314E-01	1.4326E-01	1.4330E-01	1.4343E-01
SD Yaw	9.5006E-01	1.8719E+00	8.6356E-01	1.7151E+00	6.8342E-01	1.3355E+00

Yaw Sensor -Run 5

Mean Roll	-9.2856E-04	8.3781E-04	-1.3381E-03	-1.0391E-03	-2.8523E-03	-1.8277E-03
Mean Pitch	-9.2856E-04	8.3781E-04	-1.3381E-03	-1.0391E-03	-2.8523E-03	-1.8277E-03
Mean Yaw	-1.1988E+00	-2.3930E+00	-9.2900E-01	-1.8591E+00	-6.6713E-01	-1.3310E+00
SD Roll	1.4728E-01	1.4388E-01	1.4311E-01	1.4340E-01	1.4318E-01	1.4341E-01
SD Pitch	1.4728E-01	1.4388E-01	1.4311E-01	1.4340E-01	1.4318E-01	1.4341E-01
SD Yaw	9.3666E-01	1.8740E+00	8.6482E-01	1.7109E+00	6.9236E-01	1.3381E+00

Table 5-62 TE statistics for Yaw Sensor Failures

5.3.5.2. Stick Activity Analysis

The pilot activity for the yaw sensor failure cases are shown in Table 5-63 for the 5 runs.

Yaw Sensor -Run1						
Stick Activity	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Longitudinal	0.0000	1.8940	0.0000	1.8262	0.0000	2.3537
Lateral	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Directional	3.0595	24.3675	0.1311	12.8312	0.4138	2.3343
Yaw Sensor -Run2						
Longitudinal	1.3499	3.1650	0.0000	2.1896	1.3499	0.6082
Lateral	0.0000	0.0247	0.0000	0.3769	0.0302	0.0000
Directional	35.0084	14.9832	2.7477	16.3951	12.2264	9.8777
Yaw Sensor -Run3						
Longitudinal	0.2927	0.0000	1.5230	0.3324	0.2927	0.0000
Lateral	0.0747	0.1965	0.0221	0.0250	0.4770	0.0000
Directional	2.4806	2.2408	17.3698	5.3719	1.0700	1.8829
Yaw Sensor -Run4						
Longitudinal	0.0000	1.2387	0.0000	0.0000	0.0000	0.7046
Lateral	0.0708	0.1148	0.0000	0.0453	0.0000	0.1379
Directional	3.7617	6.3574	4.9926	2.3088	4.5386	10.5153
Yaw Sensor -Run5						
Longitudinal	0.6370	0.5484	0.0000	0.5762	0.6370	0.3030
Lateral	0.0000	0.0000	0.1251	0.0836	0.7332	0.0010
Directional	8.9623	11.5055	5.4615	5.3606	5.0543	5.5852

Table 5-63 Stick Activity for Yaw Sensor Failures

The average of the pilot activity for the yaw sensor failure cases is shown in Table 5-64. There is a definitive trend of pilot activity higher for a large bias than small bias. Consider SF#1(SFDB) and SF#2 (LFDB) which differs only in the bias i.e. 4 deg and 8 deg. The stick activity along longitudinal, lateral and directional channel is higher for SF#2 than SF#1 for a fast drifting bias. Similarly, it is higher for SF#4 than SF#3 for a medium drifting bias and higher for SF#6 than SF#5 for a slow drifting bias. This pilot activity is mainly because of the time delay in detection between the switching of sensors with DNN estimates.

Stick Activity	SF #1	SF #2	SF #3	SF #4	SF #5	SF #6
Longitudinal	0.4559	1.3692	0.3046	0.9849	0.4559	0.7939
Lateral	0.0291	0.0672	0.0294	0.1062	0.2481	0.0278
Directional	10.6545	11.8909	6.1405	8.4535	4.6606	6.0391

Table 5-64 Average Stick Activity for Yaw Sensor Failures

5.3.5.3. FDII Analysis of Yaw Sensor Failures

The FDII results for yaw sensor failure cases are shown in Table 5-65 -Table 5-69. The FDII scheme has three cases of wrong isolation and identification. In first run there is a wrong identification for SF #6 (LSDB) being wrongly identified as pitch sensor, a wrong isolation as actuator in Run2 & Run4 for SF# 3(LFDB) and consequently rudder failure identification. This can be attributed to the hard nature of the sensor failure. The dynamic signature of a hard yaw sensor is similar to a rudder failure.

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Yaw Sensor SFDB	31.54	32.54	32.56	T
2	Yaw Sensor LFDB	31.38	32.38	32.4	T
3	Yaw Sensor SMDB	33.78	34.78	34.8	T
4	Yaw Sensor LMDB	31.72	32.72	32.74	T
5	Yaw Sensor SSDB	36.62	37.62	37.64	T
6	Yaw Sensor LSDB	28.06	29.06	29.8	F(Pitch)

Table 5-65 FDII results for Yaw Sensor Failure (Run1)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Yaw Sensor SFDB	31.46	32.46	32.48	T
2	Yaw Sensor LFDB	31.04	31.12	31.14	F(RUDDER)
3	Yaw Sensor SMDB	33.14	34.14	34.16	T
4	Yaw Sensor LMDB	31.74	32.74	32.76	T
5	Yaw Sensor SSDB	31.64	32.64	33.66	T
6	Yaw Sensor LSDB	31.96	32.96	32.98	T

Table 5-66 FDII results for Yaw Sensor Failure (Run2)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Yaw Sensor SFDB	31.48	32.48	32.5	T
2	Yaw Sensor LFDB	30.72	31.72	31.74	T
3	Yaw Sensor SMDB	31.16	32.16	32.18	T
4	Yaw Sensor LMDB	31.14	32.14	32.16	T
5	Yaw Sensor SSDB	34.74	35.74	35.8	T
6	Yaw Sensor LSDB	33.34	34.74	34.76	T

Table 5-67 FDII results for Yaw Sensor Failure (Run3)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Yaw Sensor SFDB	31.56	32.56	32.58	T
2	Yaw Sensor LFDB	30.86	31.46	31.54	F(RUDDER)
3	Yaw Sensor SMDB	33.34	34.34	34.42	T
4	Yaw Sensor LMDB	31.74	32.74	32.76	T
5	Yaw Sensor SSDB	32.6	33.6	33.62	T
6	Yaw Sensor LSDB	30.28	31.28	31.62	T

Table 5-68 FDII results for Yaw Sensor Failure (Run4)

S.No	Failure Cases	Detection	Isolation	Identification	T/F
1	Yaw Sensor SFDB	30.22	31.22	31.24	T
2	Yaw Sensor LFDB	31.5	32.5	32.52	T
3	Yaw Sensor SMDB	30.54	31.54	31.56	T
4	Yaw Sensor LMDB	32.98	33.98	34	T
5	Yaw Sensor SSDB	34.22	35.72	35.74	T
6	Yaw Sensor LSDB	31.18	32.18	32.22	T

Table 5-69 FDII results for Yaw Sensor Failure (Run5)

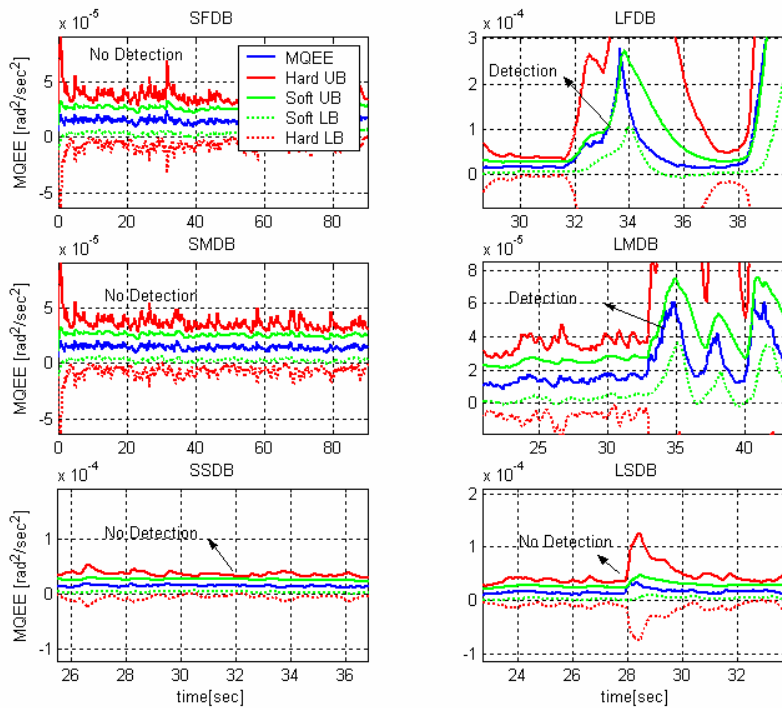


Figure 5-60 MQEE variation for Yaw Sensor Failure

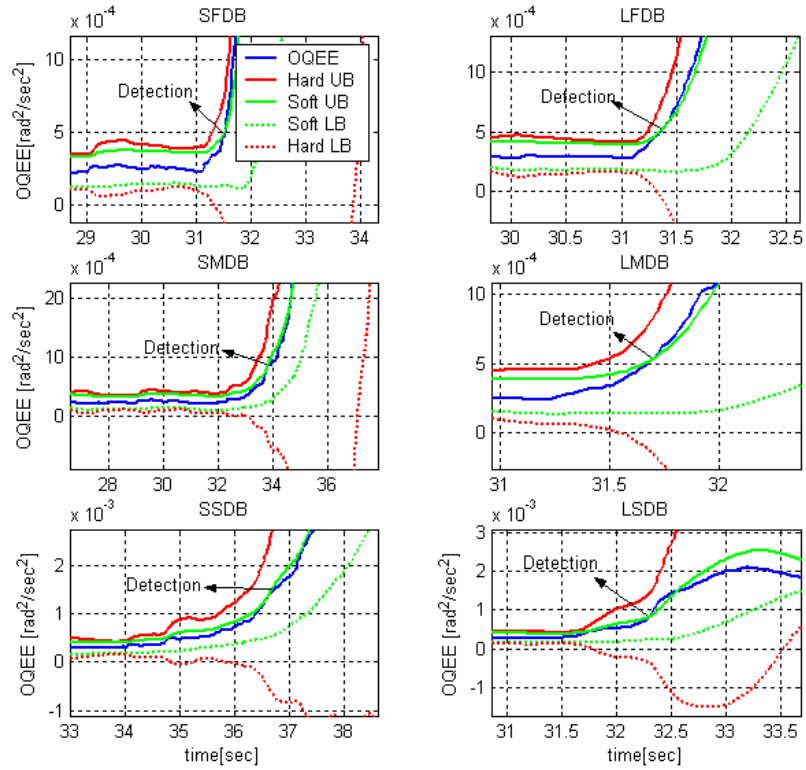


Figure 5-61 QGEE for Detection of Yaw Sensor Failures

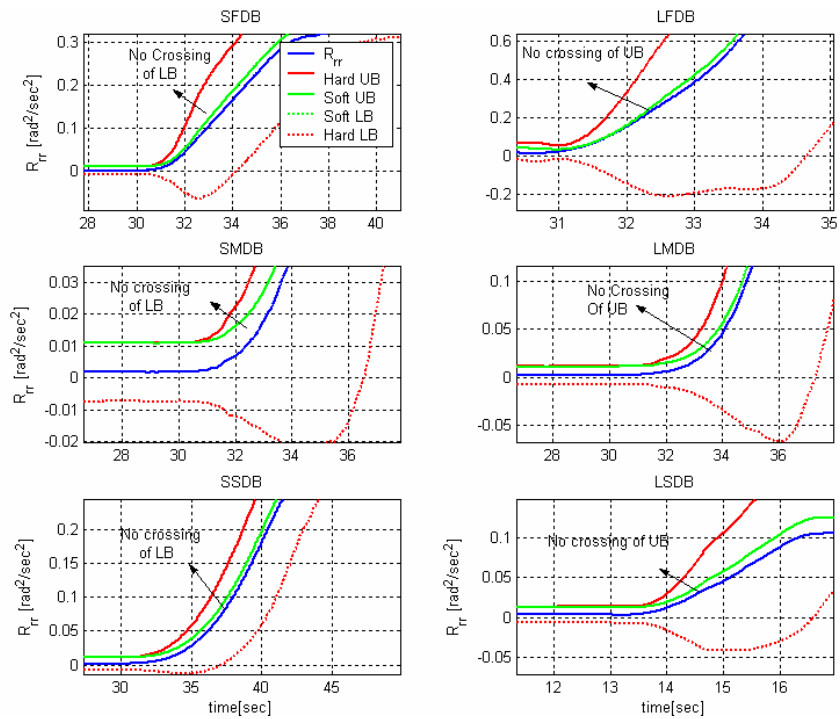


Figure 5-62- Isolation of Yaw Sensor Failure with Rrr

Tests	Total Failures Tested	Correctly Detected	False Alarms	False Detections	False Isolations	False Identifications
Roll sensor failure	30	30	0	0	0	0
Pitch Sensor Failure	30	30	0	0	0	0
Yaw Sensor Failure	30	27	0	0	2	3

Table 5-70-Summary of Piloted Tests of Sensor Failures with Adaptive FDI scheme

The sensor failures are easily isolated and identified in the piloted experiments. Except for the yaw sensor failures all failures are isolated and identified 100% accurately. The sensor failures didn't have any motion cues because of sensor biases. However, the instruments in the cockpit reflected this change such as a dipping horizon indicator in case of pitch or roll sensor failure. The significant change in instrument reading for the tests were for the hardest failure i.e. LFDB (SF#2) which prompted the pilot to correct it. In rest of the cases the pilot activity was because of the DNN values not exact as for a real sensor and the pilot tries to adjust it.

Chapter 6. Conclusion and Recommendations

6.1. Conclusion

The MOTUS flight simulator has been successfully interfaced with the Simulink schemes in this research effort. This is a novel approach to utilize the aural, visual and motion cues of a six-DOF simulator for development and testing of control schemes. Based on the feedback from the various experienced pilots flying the F-15 models using this interface, it can be concluded that the cues are very much realistic and can supersede desktop simulations for many advanced studies. There are a few limitations in using the interface for carrying out the simulation studies. The unavailability of an F-15 graphical model in the X-plane Version 7.61 deemed it necessary to use an F-4 Phantom aircraft graphics for the display and visuals. This assumption didn't affect the flight experiments as the necessary instruments were calibrated to reflect the same values as the Simulink model. Secondly, the cockpit of the MFS was not used for the experiments due to limited access to the simulator MBC.

The NLDI based control schemes together with NN augmentations have demonstrated the earlier trends in the simulator and it corroborated the earlier studies on AFDIA and SFDIA using the fixed threshold approach. The pilot in the loop experiments provided a new dimension to the performance analysis. The results of piloted experiments in the MOTUS flight simulator have demonstrated good results for the FDII scheme for the multiple runs of the actuator and sensor failures except for a few false FDIIs. The adaptive threshold approach to FDII has quick detection of the failure. The scheme was found to be robust to small perturbations arising out of pilot activity. The FDII scheme had the best performance for sensor failures particularly along roll and pitch channels with no cases of false FDII. However, yaw sensor failures induce coupling along all the three channels and therefore is more susceptible to false detection. The actuator failures FDII is acceptable as it detected and identified the failure correctly for 90% of the test cases. Due to the cross-coupling due to pilot activity there were a few cases of false detections and isolations.

From the TE results of stabilator failure, it can be concluded that EMRAN showed a good performance in hard failures. The average pilot workload was lower in case of EMRAN augmentation except one time when it was comparable with No NN. The CP remained the best indicator of the performance with EMRAN having lower values compared to the No NN test cases. There is a reduction with EMRAN along the roll and pitch channels as they are the dominant channel on this type of failure. The performance along yaw didn't show any improvement with EMRAN except for hard failures. In summary EMRAN was very helpful to the pilot in terms of reduced workload in this stabilator failure along dominant channels. The pilot activity was required mainly along roll and pitch channels and the CP give an indication that it required lesser pilot activity on behalf of the pilot to compensate for the failure and maintain the mission profile.

Similar trends are observed in FDII tests of aileron failures and as before EMRAN performance was better along the dominant channel i.e. roll for this failure. Much of the activity for compensating this failure was required along roll. There is a significant improvement with EMRAN along the roll channel in terms of TE and reduced pilot activity. EMRAN had best performance in the dominant channel and shows little or no improvement along the cross channel.

The sensor failure cases didn't have considerable pilot activity because of no motion cues. The pilot activity was due to the visuals (change in instrument readings) which the pilot tried to compensate. The hard failure (SF#2) had the maximum pilot activity as it gave a substantial instrument error and pilot tried compensating it during the interval the sensor readings were replaced by MNNs. The CP was of the same order for sensor failures in rest of the test cases.

Based on the tests it can be concluded that composite parameter is a good indication of the overall system performance with a pilot in loop. It has shown consistent improvement with EMRAN NN augmentations in the dominant channels compared to No NN augmentation. The EMRAN performance degraded in cross channels for example in yaw channel for aileron failures. It can be concluded from the CP values that the NN

augmentation was helpful along the dominant channels to the pilot on the onset of failure. One reason for this observation can be attributed to the pilot compensation after failure. For example in case of stabilator failure the pilot didn't compensate in yaw channel and therefore showed a decrease in EMRAN performance in CP for this channel. In conclusion EMRAN was best for dominant channels particular to a failure and didn't show any appreciable performance in cross channels.

This research covered the overview of the simulation environment set up for flight control research at WVU. An overview of architecture and functionality of the components of the interface of Simulink with MOTUS motion base flight simulator is presented. The pilot in the loop experiments on FDII schemes provided an excellent example of the software-hardware setup at the WVU in carrying out advanced simulation studies for future. It can be concluded that the FDII scheme developed for an integrated sensor/actuator failure with adaptive threshold have shown good performance with accurate detections with a pilot in the loop. This interface developed at the flight simulator extends the simulation capability from desktop to a more realistic environment.

6.2. Recommendations

This research study had opened a new arena for testing flight control schemes that are simulation dependent for development and analysis. The software-hardware interface together with a man in the loop has potential application for carrying out a man-machine interaction study in the simulator. This application can be extended to carry out studies of pilot behavior while carrying out different maneuvers or counteracting different failures. Potential application can be in the field of FDII algorithms with fatigue measurements of pilot after failure and estimating the effect of pilot reaction to series of failures.

In this study the actuator and sensor failures were induced separately and there were no successive failures (actuator after sensor etc) in the same test. Testing the FDII schemes with multiple failures is possible extension to this study and would check the robustness of the schemes and will allow further improvement to the FDII approach.

Also, we have constrained the pilot mission profile to a level flight based on the trim settings at a particular flight condition. The throttle is kept constant throughout the flight and the pilot didn't use any throttle control during the course of experiments. As the first step to use the flight simulator there are many assumptions like constant throttle, level flight and maintaining constant altitude. The NN have been trained for the particular flight scenario and a natural extension would be to accommodate changes in flight conditions and reconfiguration of NN algorithms based on changing flight envelopes. A more complex mission profile should be tested based as there was unavailability of on availability of experienced fighter test pilots. More pilot feedback parameters can be devised in order to quantify the pilot assessment of the problem in hand while performing these types of simulation studies.

References

- ¹ McMahan, J., Flight 1080, *Airline Pilot*
- ² NTSB, “National Transportation Safety Board Accident Report of the American Airline DC10 Crash at Chicago O Hare’ International Airport”, NTAB-AAR-79-17, May 1979.
- ³Himmelblau, D.M., “Fault Detection and Diagnosis in Chemical and Petrochemical Processes” Elsevier Press, Amsterdam
- ⁴Kitamura, M., “Fault Detection in Nuclear Reactors with aid of parametric modeling methods, Chapter 9 in: Patton Frank and Clark, *Fault Diagnosis in dynamic systems: Theory and Applications*”, Prentice Hall.
- ⁵Duyar, A., Merrill, W., “A failure Diagnosis System Based on a Neural Classifier for a Space Shuttle Engine”, *Proceedings of 29th IEEE CDC*, Honolulu, HI, pp.2391-2400
- ⁶ Mehra, R.K., Peschon, J., “An innovation approach to fault detection and diagnosis in dynamic systems”, *Automatica*, pp. 637-640, Pergamon Press, 1971
- ⁷ Mehra, R.K., Seereeram, S., Bayard, D., Haydaegh, F., “Adaptive Kalman Filtering, Failure Detection and Identification for Space Craft Attitude Estimation”, *IEEE Conference on control Applications, Albany, NY, pp 176-181, Sep. 1995.*
- ⁸Napolitano M.R., Younghawn A., Seanor B., “A Fault Tolerant Flight Control System for Sensor and Actuator Failures using Neural Networks”, *Aircraft Design*, 2000, vol.3 no 2
- ⁹ Qi J., Zhao X., Jiang Z., Han J., “An Adaptive Threshold Neural Network Scheme for Rotorcraft UAV Sensor Failure Diagnosis,” *Advances in Neural Networks*. Vol.4493, 2007
- ¹⁰ Wang, X., Syrmos, V.L., “Design of Dynamic Fault-Tolerant Control System using IMM Estimation and RBF Neural Network”, *14th Mediterranean Conference on Control and Automation*, Jun. 2006, pp 1-6
- ¹¹Clark, R.N., “A Simplified Instrument Failure Detection Scheme”, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-14, Jul. 1978, pp 558-563.
- ¹² Wilbers, D.M, Speyer, J.L., “Detection Filters for Aircraft Sensor and Actuator Faults”, *Proceedings of the IEEE International Conference on Control and Applications*, Jerusalem, Apr. 1989

¹³ Jones, G.J., Corbin, M.J., “Analytical redundancy using band-limiting filters”, *IEEE Proceedings*, 135, Pt. D, No. 5, pp 257-265, 1988.

¹⁴ Clark, R.N., Fosth, D.C., Walton, V.M., “Detection of Instrument Malfunction in Control Systems,” *IEEE Transactions in Aerospace and Electronic Systems*, Vol.11, pp. 465-473, 1975

¹⁵ Mehra, R.K., Peschon, I., “An innovation approach to fault detection and diagnosis in dynamic systems,” *Automatica*, Vol. 7, pp. 637-640, 1971

¹⁶ Montgomery, R.C., Caglayan, A.K., “Failure Accommodation in Digital Flight Control Systems by Bayesian Decision Theory”, *Journal of Aircraft*, Vol.13, 1976

¹⁷ Willsky, A., “A survey of design methods for failure detection”, *Automatica*, Nov. 1976, pp. 601-611

¹⁸ Basseville, M., “Detecting changes in Signals and Systems”, *Automatica*, 1988

¹⁹ Clark, R.N., “A dedicated observer approach to Instrument Fault Detection”, *Proceedings of 15th CDC*, Fort Lauderdale, FA, pp. 237-241 Dec. 1979[Observer Based]

²⁰ Patton, R.J., Chen, J., “A reexamination of relationship between parity space and observer based approach to fault diagnosis”, *European Journal of Diagnosis and Safety and Automation*, 1991

²¹ Kratz, F., Nuninger, W., Ploix, S., “Fault Detection for time delays systems: A parity space approach”, *Proceedings of American Control Conference*, Vol. 4, pp.2009-2011, 1998.

²² Gertler, J., “Fault Detection and Isolation using Parity Relations”, *Control Engineering Practice*, Vol.5, Issue 5, pp. 653-661, 1997

²³ Isermann, R., “Process Fault Detection based on modeling and estimation methods”, *Automatica*, 20, pp. 387-404, 1984

²⁴ Massoumnia, “A Geometric Approach to failure detection and identification in linear systems”, PhD Thesis, MIT, Department of Aeronautics and Astronautics, 1986

²⁵ Patton, R.J., Chen, J., “A review of parity Space Approach to Fault Diagnosis”, *Concise Encyclopedia of Simulation and Modeling*, 379-392, Pergamon Press, 1992

- ²⁶Handelman, D.A., Stengel, R.F., "Combining Expert Systems and Analytical Redundancy concepts for Fault Tolerant Flight Control", *Journal of Guidance Control and Dynamics*, pp.39-45, 1989
- ²⁷"Intelligent Flight Control: Advanced Concept Program – Final Report", The Boeing Company, BOEING-STL 99P0040, May 1999.
- ²⁸"Intelligent Flight Control: Advanced Concept Program – Final Report", The Boeing Company, BOEING-STL 99P0040, May 1999
- ²⁹Clark, R.N., "A Simplified Instrument Failure Detection Scheme", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-14, Jul. 1978, pp 558-563.
- ³⁰Clark, R.N., "A Simplified Instrument Failure Detection Scheme", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-14, Jul. 1978, pp 558-563.
- ³¹Moerder, D.D., Halyo, N., Broussard, J.R., Caglayan, A.K., "Application of pre computed control laws in a reconfigurable aircraft flight control system", *Journal of Guidance, Control and Dynamics*, Vol. 12, No.3, pp. 325-333, 1989.
- ³²Ostroff, A.J., "Techniques for Accommodating Control Effector Failures on a Mildly Statistically Unstable Airplane", American Control Conference, 1985
- ³³Shamma, J.S., Athans, M., "Analysis of Gain Scheduled Control of Non-Linear Plants", *IEEE Transactions on Automatic Control*, Vol. 35, Issue 8, pp.898-907. Aug. 1990.
- ³⁴Minsky, M., Papert S., "Perceptrons: An introduction to Computational Geometry," Rev. ed. Cambridge, MA, M.I.T Press, 1988
- ³⁵McCorduck P., "Machines who Think", San Fransico, CA: WH Freeman 1979
- ³⁶Mehra, R., Rago, C., Seereeram, S., "Autonomous Failure Detection and Identification and Fault Tolerant Estimation with Aerospace Applications", *Proceedings of IEEE Aerospace Conference*, Vol.2, 21-28 March, 1998, pp 133-138.
- ³⁷Eide, P., Maybeck, P., "An MMAE failure detection system for F-16", *IEEE Transactions on Aerospace and Electronic Systems*", Vol.32, No.3, Jul. 1996

³⁸Calise A. J., Lee S., Sharma M., "Direct Adaptive Reconfigurable Control of a Tailless Fighter Aircraft", *Proc. of the 1998 AIAA Guidance, Navigation and Control Conference*, Boston MA, August 1998, AIAA 98-4108.

³⁹Rysdyk R.T., Calise A. J., "Fault Tolerant Flight Control via Adaptive Neural Network Augmentation", AIAA 98-4483, August 1998.

⁴⁰Calise A. J., Sharma M., "Adaptive Autopilot Design for Guided Munitions", *AIAA Journal of Guidance, Control and Dynamics*, vol. 23 no 5, 2000

⁴¹Kaneshige J., Bull J., Totah J.J., "Generic Neural Flight Control and Autopilot System", *AIAA Paper 00-4281*, Denver, Co, August 2000

⁴²Kaneshige J., Gundy-Burlet K., "Integrated Neural Flight and Propulsion Control System", AIAA Paper 01-4386

⁴³Napolitano M.R., Song Y., Seanor B., "On-Line Parameter Estimation for Restructurable Flight Control Systems", *Aircraft Design*, 4, 2001, Elsevier Science Ltd

⁴⁴Perhinschi M. G., Lando M., Massotti L., Campa G., Napolitano M. R., Fravolini M. L., "On-Line Parameter Estimation Issues for the NASA IFCS F-15 Fault Tolerant Systems", *Proc. of the 2002 American Control Conference*, Anchorage AK, May 2002, AIAA2002-1079

⁴⁵Perhinschi M. G., Lando M., Massotti L., Campa G., Napolitano M. R., Fravolini M. L., "On-Line Parameter Estimation Issues for the NASA IFCS F-15 Fault Tolerant Systems", *Proc. of the 2002 American Control Conference*, Anchorage AK, May 2002, AIAA2002-1079

⁴⁶Perhinschi M.G., Campa G., Napolitano M.R., Lando M., Massotti L., Fravolini M.L., "A Simulation Tool for On-line Real Time Parameter Identification", *Proceedings of the 2002 AIAA Modeling and Simulation Technologies Conference*, Monterey, Ca, August 2002.

⁴⁷Brumbaugh, R.W. "An Aircraft Model for the AIAA Controls Design Challenge", NASA CR 186019, Dec. 1991

⁴⁸Perhinschi M.G., Campa G., Napolitano M.R., Lando M., Massotti L., Fravolini M.L., "Modeling and Simulation of Failures for Primary Control Surfaces", *Proceedings of the 2002 AIAA Modeling and Simulation Technologies Conference*, Monterey, Ca, August 2002

- ⁴⁹ Curry, T., Collins, E.G., Selekwa, M., "Robust fault Detection Using Robust L1 estimation and Fuzzy Logic", Proceedings of American Control Conference, Jun. 25-27, 2001
- ⁵⁰ Wilsky A.S., "Failure Detection in Dynamic Systems", Agard LS-109, Neuilly sur Seine, France, October, 1980, 2.1-2.14
- ⁵¹ Kerr T.H., "False Alarm and Correct Detection Probabilities over a Time Interval for Restricted Classes of Failure Detection Algorithms", *IEEE Transactions of Information Theory*, 1982, IT-28, No. 4, pp. 619-631
- ⁵² Ha C.M., Wei Y.P., Bessolo J.A., "Reconfigurable Aircraft Flight Control System Via Neural Networks", *Proceedings of the 1992 Aerospace Design Conference*, AIAA Paper 92-1075, Irvine, Ca, Feb. 1992
- ⁵³ Huang C., Tylock J., Engel S., Whitson J., Eilbert J., "Failure-Accommodating Neural Network Flight Control", *Proceedings of the AIAA Guidance, Navigation and Control Conference*, AIAA Paper 92-4394, Hilton Head, SC, Aug. 1992
- ⁵⁴ Napolitano M.R., Chen C.I., Naylor S., "Aircraft Failure Detection and Identification Using Neural Networks", *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 6, pp. 999-1009, Nov.-Dec. 1993
- ⁵⁵ Napolitano M.R., Casdorff V., Neppach C., Naylor S, "On-line Learning Neural Architectures and Cross-Correlation Analysis for Actuator Failure Detection and Identification", *International Journal of Control*, Vol. 63, No. 3, pp. 433-455, 1996
- ⁵⁶ Napolitano M. R., Younghawn A., Seanor B., "A Fault Tolerant Flight Control System for Sensor and Actuator Failures Using Neural Networks," *Aircraft Design*, 2000, vol. 3 No. 2
- ⁵⁷ Perhinschi M., Napolitano M. R., Seanor B., Burken J., Larson R., "An Adaptive Threshold Approach for Actuator failure Detection and Identification," *IEEE Transactions on Control Systems Technology*, Vol.14, No.3, May 2006.
- ⁵⁸ Perhinschi M., Napolitano M.R., Campa G., Fravolini M.L., "Integration of Fault Tolerant System for Sensor and Actuator Failures within the WVU NASA F-15 Simulator", *AIAA Guidance Navigation and Control Conference*, Austin TX, 2003

- ⁵⁹ Perhinschi M. G., Campa G., Napolitano M.R., Lando M., Massotti L., Fravolini M. L., "A Simulation Tool for On-Line Real Time Parameter Identification", AIAA2002-4685 *Proceedings of the AIAA Modeling and Simulation Conference*, Monterey, CA 2002
- ⁶⁰ Polycarpou M., "On-Line Approximators for Nonlinear System Identification: A unified Approach", *Control and Dynamic Systems Series*, vol. 7, Ac. Press, Jan. 1998.
- ⁶¹ Chiang, C., Youssef, H.M., "Neural Network and Fuzzy Logic Approach to Aircraft Reconfigurable Control Design", *Proceedings of the American Control Conference*, Vol.5, pp.3505-3509, Jun. 1995
- ⁶² Al-Jarrah, O.M., Al-Rousan, M., "Failure detection and accommodation in dynamic systems using Adaptive Neuro-Fuzzy Systems", *IEE Proceeding in Control Theory and Applications*, Vol.148, pp.283-290, Jul.2001
- ⁶³ Campa G., Fravolini M.L., Napolitano M.R., "A Library of Adaptive Neural Networks for Control Purposes", *IEEE International Symposium on Computer Aided Control System Design*, September 18-20, 2002 Glasgow, Scotland, UK.
- ⁶⁴ Fravolini M.L., Campa G., Napolitano M.R., La Cava M., "Comparison of different growing Radial Basis Function Algorithms for Control System Applications", *Proceeding of the 2002 American Control Conference*, Anchorage, AK, May 2002
- ⁶⁵ Butler, Geoff F., Malcom J. Corbin, S. Mephram, James F. Stewart, and Richard R. Larson, "NASA /RAE Collaboration on Nonlinear Control using the F-8C Digital Fly-by-Wire Aircraft, NASA TM-84296, 1983.
- ⁶⁶ Perhinschi M. G., Napolitano M.R., Campa G., Burke H. E., Larson R. R., Burken J., Fravolini M. L., "Design and Testing of a Safety Monitor Scheme on the NASA Gen_2 IFCS F-15 Flight Simulator", in *Proc. of the 2004 AIAA Intelligent Systems Technical Conference* Chicago IL, Sept. 2004
- ⁶⁷ Meyer A., Van Kampen H., "X-Plane On-Line Instruction Manual", Laminar Research Inc., Columbia SC, 8th edition, 2002 (www.X-Plane.com)