

NAVAL POSTGRADUATE SCHOOL MONTEREY, CALIFORNIA



THESIS

OPTIMAL FAULT DETECTION AND RESOLUTION DURING MANEUVERING FOR AUTONOMOUS UNDERWATER VEHICLES

by

Andrew S. Gibbons

March 2000

Thesis Advisor:

Anthony J. Healey

Approved for public release; distribution is unlimited.

DTIC QUALITY INSPECTED 3

20000502 065

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 2000.		3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE: Optimal fault detection and resolution during maneuvering for Autonomous Underwater Vehicles			5. FUNDING NUMBERS	
6. AUTHOR(S) ANDREW S. GIBBONS				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Research, 800 N. Quincy St., Arlington, VA 22217-5660			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) In order to increase robustness, reliability, and mission success rate, autonomous vehicles must detect debilitating system control faults. Prior model-based observer design for 21UUV was analyzed using actual vehicle sensor data. It was shown, based on experimental response, that residual generation during maneuvering was too excessive to detect manually implemented faults. Optimization of vehicle hydrodynamic coefficients in the model significantly decreased maneuvering residuals, but did not allow for adequate fault detection. Kalman filtering techniques were used to improve residual reduction during maneuvering and increase residual generation during fault conditions. Optimization of the Kalman filter's system noise matrix, measurement noise matrix, and input gain scalar multiplier produced fault resolution which allowed for accurate detection of faults of relatively minor magnitude within minimal time constraints.				
14. SUBJECT TERMS: Autonomous Underwater Vehicles, Robotics, Robust Fault Detection, Extended Kalman Filtering, Optimization, Reliable Fault Sensitivity			15. NUMBER OF PAGES ##	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18 298-102

Approved for public release; distribution is unlimited.

**OPTIMAL FAULT DETECTION AND RESOLUTION DURING
MANEUVERING FOR AUTONOMOUS UNDERWATER VEHICLES**

Andrew S. Gibbons
Lieutenant, United States Navy
B.S., United States Naval Academy, 1993

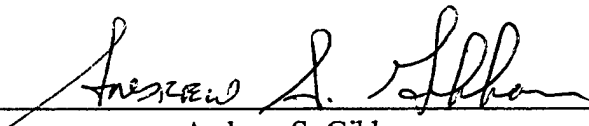
Submitted in partial fulfillment of the
Requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
March 2000**

Author:

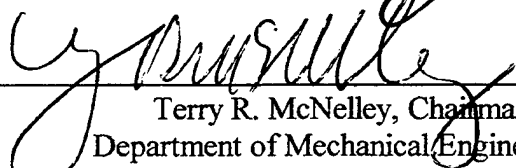


Andrew S. Gibbons

Approved by:



Anthony J. Healey, Thesis Advisor



Terry R. McNelley, Chairman
Department of Mechanical Engineering

ABSTRACT

In order to increase robustness, reliability, and mission success rate, autonomous vehicles must detect debilitating system control faults. Prior model-based observer design for 21UUV was analyzed using actual vehicle sensor data. It was shown, based on experimental response, that residual generation during maneuvering was too excessive to detect manually implemented faults. Optimization of vehicle hydrodynamic coefficients in the model significantly decreased maneuvering residuals, but did not allow for adequate fault detection. Kalman filtering techniques were used to improve residual reduction during maneuvering and increase residual generation during fault conditions. Optimization of the Kalman filter's system noise matrix, measurement noise matrix, and input gain scalar multiplier produced fault resolution which allowed for accurate detection of faults of relatively minor magnitude within minimal time constraints.

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	GENERAL BACKGROUND AND LITERATURE	1
B.	SCOPE OF THIS WORK.....	6
II.	DESCRIPTION OF FAULTS AND ASSOCIATED DETECTION TECHNIQUES.....	9
A.	TYPES OF FAULTS	9
1.	Environmentally Induced Faults	9
2.	System Induced Faults	10
B.	MAGNITUDE AND NATURE OF FAULTS	10
C.	TECHNIQUES FOR FAULT DETECTION AND DIAGNOSTICS	11
1.	'Limits and Trends' Analysis.....	11
2.	Model-Free Detection.....	12
3.	Model-Based Detection	12
D.	STEERING SUBSYSTEM MODEL-BASED OBSERVER DESIGN FOR 21UUV.....	13
1.	Theory.....	14
2.	Application	16
3.	The Effects of Modeling Uncertainty.....	17
III.	MODEL-BASED OBSERVER PERFORMANCE USING ACTUAL 21UUV DATA	19
A.	FAULT DETECTION ARCHITECTURE FOR 21UUV	19
B.	DESCRIPTION OF 21UUV MODEL.....	21
1.	Development of 21UUV Steering Model.....	21
2.	21UUV X-fin Configuration	25
C.	21UUV DATA ANALYSIS.....	27
1.	Parameters and Characteristics of the Run.....	27
2.	Track Analysis of the Run.....	28
3.	Data Sets for Residual Analysis (Maneuvering Specific).....	33
D.	RESIDUAL GENERATION OF NOMINAL OBSERVER DESIGN	41

E.	IMPLEMENTATION OF FAULT INTO MODEL-BASED OBSERVER DESIGN	45
1.	Description of Manual Fault Implementation.....	45
2.	Analysis of Fault Detection	46
F.	CONCLUSIONS.....	53
IV.	OPTIMIZATION OF VEHICLE MODEL HYDRODYNAMIC COEFFICIENTS	55
A.	ANALYSIS OF DYNAMIC AND CONTROL MATRIX ERROR	55
1.	Error Analysis of Steering Subsystem Model.....	55
2.	Choice of Hydrodynamic Coefficients for Optimization.....	57
B.	OPTIMIZATION OF HYDRODYNAMIC COEFFICIENTS	60
1.	<i>Matlab's</i> Sequential Quadratic Programming Method	60
a.	Quadratic Programming Subproblem.....	61
b.	Line Search and Merit Function Calculation	63
c.	Updating the Hessian Matrix	64
2.	Optimization Arguments for Use in 'Constr'	66
a.	Function String.....	66
b.	x0.....	66
c.	Options	67
d.	VLB and VUB	67
e.	SS and ES	68
3.	Scalar Reduction by Use of a Weighting Matrix	68
4.	Use of a Performance Index to Increase Fault Detection.....	70
5.	Root Mean Squared Error Analysis of Generated Residuals	71
C.	RESULTS OF HYDRODYNAMIC COEFFICIENT OPTIMIZATION.....	72
1.	Tabular Results of Hydrodynamic Coefficient Optimization.....	72
a.	Optimized Hydrodynamic Values for Each Data Set Interval.....	73
b.	Performance Characteristics of Optimized Coefficients	74
2.	Graphical Results of Hydrodynamic Coefficient Optimization	75
D.	MANUAL FAULT INTRODUCTION AND DETECTION.....	83
E.	CONCLUSIONS.....	90

V.	UTILIZATION OF THE EXTENDED KALMAN FILTER FOR FAULT DETECTION	93
A.	BASIC INTRODUCTION TO KALMAN FILTERING.....	93
B.	ANALYSIS OF BASIC KALMAN FILTER FAULT DETECTION	96
C.	ANALYSIS OF UNCERTAIN KALMAN PARAMETERS.....	102
D.	OPTIMIZATION OF Q, R, AND β	104
1.	Parameters for Optimization of the Kalman Filter	104
2.	Tabular and Graphical Results for Q, R, and β Optimization	106
a.	Optimized Components for Each Data Set Interval.....	106
b.	Performance Characteristics of Optimized Alpha	107
3.	Graphical Results for Q, R, and β Optimization.....	109
E.	MANUAL FAULT INTRODUCTION AND DETECTION.....	112
F.	CONCLUSIONS.....	120
VI.	ANALYSIS OF FAULT RESOLUTION OF FINAL DESIGN.....	123
A.	DEVELOPMENT OF A FAULT DETECTION ALGORITHM.....	123
B.	ANALYSIS OF FAULT DETECTION FOR EACH DATA INTERVAL	124
C.	FAULT MAGNITUDE SENSITIVITY FOR DETECTION	128
D.	METHODOLOGY FOR FAULT DETECTION DESIGN TAILORING.....	134
E.	CONCLUSIONS.....	134
VII.	CONCLUSIONS AND RECOMMENDATIONS.....	137
A.	CONCLUSIONS.....	137
B.	RECOMMENDATIONS	137
APPENDIX A.	<i>MATLAB</i> CODE ASSOCIATED WITH ORIGINAL MODEL-BASED OBSERVER DESIGN.....	139
APPENDIX B.	<i>MATLAB</i> CODE ASSOCIATED WITH OPTIMIZATION OF MODEL- BASED OBSERVER DESIGN.....	141

APPENDIX C. <i>MATLAB</i> CODE ASSOCIATED WITH OPTIMIZED KALMAN FILTER DESIGN	143
LIST OF REFERENCES	145
INITIAL DISTRIBUTION LIST	147

LIST OF FIGURES

Figure 1.1	Residual Generation Defined	4
Figure 3.1	Fault Detection Architecture w/ Fuzzy Inference System for Fault State Resolution	20
Figure 3.2	Stern-view Aspect of X-Fin Configuration on 21UUV	25
Figure 3.3	Steering Command Response to Port Turn	26
Figure 3.4	Rap Count Analysis of Heading Data	29
Figure 3.5	No Rap Count Plot Compared w/ Rap Count Plot	30
Figure 3.6	Dead-Reckoning Solution of 21UUV Track	32
Figure 3.7	Data Set Interval One	35
Figure 3.8	Data Set Interval Two	35
Figure 3.9	Data Set Interval Three	36
Figure 3.10	Data Set Interval Four	36
Figure 3.11	Data Set Interval Five	37
Figure 3.12	Sideslip Velocities for Each Data Set Interval	38
Figure 3.13	Yaw Rates for Each Data Set Interval	39
Figure 3.14	Heading Angles for Each Data Set Interval	40
Figure 3.15	Sideslip Velocity Residuals for Observer Design	42
Figure 3.16	Yaw Rate Residuals for Observer Design	43
Figure 3.17	Heading Residuals for Observer Design	44
Figure 3.18	Sideslip Fault Detection Using Observer Design (Data: 4800-5800)	48
Figure 3.19	Yaw Rate Fault Detection Using Observer Design (Data: 7500-8750) ..	49
Figure 3.20	Heading Fault Detection Using Observer Design (Data: 9250-10250) ..	50
Figure 3.21	Sideslip Fault Detection Using Observer Design (Data: 10750-11500) ..	51
Figure 3.22	Heading Fault Detection Using Observer Design (Data: 18500-20500) ..	52
Figure 4.1	Algorithm for <i>Matlab</i> 's Sequential Quadratic Programming (SQP)	65
Figure 4.2	Residual Reduction by Hydrodynamic Coefficient Optimization (Data Interval: 4800-5800)	76
Figure 4.3	Residual Reduction by Hydrodynamic Coefficient Optimization (Data Interval: 7500-8750)	77
Figure 4.4	Residual Reduction by Hydrodynamic Coefficient Optimization (Data Interval: 9250-10250)	78
Figure 4.5	Residual Reduction by Hydrodynamic Coefficient Optimization (Data Interval: 10750-11500)	79
Figure 4.6	Residual Reduction by Hydrodynamic Coefficient Optimization (Data Interval: 18500-20500)	80
Figure 4.7	Coefficient Set THREE Performance Over Extended Interval Data: 4000-10000, Heading Residual Response	81
Figure 4.8	Coefficient Set ONE Performance Over Extended Interval Data: 4000-10000, Heading Residual Response	82
Figure 4.9	Optimized Observer Design Response to Fin Fault "Sideslip" (Data Interval: 4800-5800)	85
Figure 4.10	Optimized Observer Design Response to Fin Fault "Yaw Rate" (Data Interval: 7500-8750)	86

Figure 4.11	Optimized Observer Design Response to Fin Fault "Heading" (Data Interval: 9250-10250)	87
Figure 4.12	Optimized Observer Design Response to Fin Fault "Yaw Rate" (Data Interval: 10750-11500)	88
Figure 4.13	Optimized Observer Design Response to Fin Fault "Heading" (Data Interval: 18500-20500)	89
Figure 5.1	Steering Subsystem Kalman Filter Architecture.....	95
Figure 5.2	Basic Kalman Filter Sideslip Residual Generation (Data: 4800-5800)....	97
Figure 5.3	Basic Kalman Filter Yaw Rate Residual Generation (Data: 4800-5800) ..	98
Figure 5.4	Basic Kalman Filter Heading Residual Generation (Data: 4800-5800)...	98
Figure 5.5	Basic Kalman Filter Relative Error Values (Data: 4800-5800)	99
Figure 5.6	Basic Kalman Filter Sideslip Residual Fault Detection (Data: 4800-5800).....	100
Figure 5.7	Basic Kalman Filter Yaw Rate Residual Fault Detection (Data: 4800-5800).....	101
Figure 5.8	Basic Kalman Filter Heading Residual Fault Detection (Data: 4800-5800).....	101
Figure 5.9	Basic Kalman Filter Relative Error Fault Detection (Data: 4800-5800).....	102
Figure 5.10	Optimized Kalman Filter Residual Reduction (Data: 4800-5800)	110
Figure 5.11	Optimized Kalman Filter Residual Reduction (Data: 7500-8750)	110
Figure 5.12	Optimized Kalman Filter Residual Reduction (Data: 9250-10250)	111
Figure 5.13	Optimized Kalman Filter Residual Reduction (Data: 10750-11500)	111
Figure 5.14	Optimized Kalman Filter Residual Reduction (Data: 18500-20500)	112
Figure 5.15	Fin Fault Detection by Optimized Kalman Design (Data: 4800-5800) ...	114
Figure 5.16	Fin Fault Detection by Optimized Kalman Design (Data: 7500-8750) ...	115
Figure 5.17	Fin Fault Detection by Optimized Kalman Design (Data: 9250-10250) .	116
Figure 5.18	Fin Fault Detection by Optimized Kalman Design (Data: 10750-11500).....	117
Figure 5.19	Fin Fault Detection by Optimized Kalman Design (Data: 18500-20500).....	118
Figure 6.1	0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 4800-5800).....	125
Figure 6.2	0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 7500-8750).....	126
Figure 6.3	0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 9250-10250).....	126
Figure 6.4	0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 10750-11500).....	127
Figure 6.5	0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 18500-20500).....	127
Figure 6.6	Fault Sensitivity to 0.2-radian Fault (Data: 10750-11500).....	130
Figure 6.7	Fault Sensitivity to 0.1-radian Fault (Data: 10750-11500).....	131
Figure 6.8	Fault Sensitivity to 0.06-radian Fault (Data: 10750-11500).....	132
Figure 6.9	Fault Magnitude vs. Time to Detect.....	133
Figure 6.10	Individual Vehicle Adaptation Procedure.....	135

LIST OF TABLES

Table 3.1	Data Set Intervals for Evaluation of Residual and Fault Detection.....	34
Table 4.1	Optimized Scalar Multipliers for Each Hydrodynamic Coefficient	73
Table 4.2	Performance Characteristics of Optimized Coefficients	74
Table 5.1	Optimized Scalar Multipliers for Alpha	107
Table 5.2	Performance Characteristics of Optimized Alpha	108

ACKNOWLEDGMENTS

I would like to acknowledge the support of the Office of Naval Research and the collaboration of Mr. Chris Hillenbrand and Mr. Mike Keegan.

A challenge as great as this thesis is surmountable only with the assistance of teachers, friends, and family. I am sincerely thankful for the unwavering guidance, support, and generosity displayed and provided to me by my thesis advisor, Professor Anthony Healey. His extensive knowledge and vast experience in the area of this research was fundamental to the entire process of creating this work, from beginning to end. I would also like to thank all the professors, officers, and administrative staff of the Naval/Mechanical Engineering Curriculum that have personally supported my efforts to succeed at the Naval Postgraduate School. Their dedication to excellence is the foundation to the success and achievement of this institution.

Finally, I must thank with all my heart, the love and support granted to me by my wonderful wife, Kimberly, and our two sons, Eli and Ethan. Their undeniable and unfathomable support and encouragement has been the motivating force behind my desire to meet and surpass the challenge of this work.

I. INTRODUCTION

A. GENERAL BACKGROUND AND LITERATURE

When a potentially dangerous or extreme mission arises where the use of human resources presents the element of excessive risk, the utilization of automated systems to satisfactorily complete the mission becomes very desirable, if not mandatory, to alleviate possible human harm. In a scenario that calls for the use of Autonomous Underwater Vehicles (AUV's), such as minefield mapping, it is imperative that the on-scene commander has the utmost confidence in the reliability of the operational assets that are assigned under his/her control to execute the assigned mission. As with all autonomous systems or machines tasked with carrying out complex mission assignments in extreme environments, AUV's may experience unforeseen problems that might threaten the mission reliability and completeness of operational goals. Thus, it is imperative to maximize the possibility of mission completeness by utilizing AUV control systems that are capable of detecting a variety of failures within their subsystems and autonomously correcting for such failures.

The ability for an AUV to compensate for its own failures may arise from the use of fault detectors combined with a Fuzzy Logic Inference System. This system would analyze the detected fault and decide whether the fault's impact may be lessened by the compensation from other on-board means, or whether the fault is severe enough to essentially transfer decision making to higher levels of authority, i.e. the on-scene commander. Prior to the execution of any fault compensating actions, it is necessary that the actual fault be *detectable* through all system and measurement noise processed by the

control systems. The ability to accurately detect the fault is paramount to the ability of the AUV to adequately compensate for the failure and subsequently, to increase the likelihood of mission completion.

The technological achievements in the design, modeling, and production of AUV's have been outstanding over recent years. There has been an abundance of current advances in technology and research that has led the way for the accomplishment of this work in the area of fault detection. Although much progress has occurred, it is still necessary to improve upon the precision of underwater navigation, the development of more sensitive sensors, the capability and dependability of underwater communications, and the reliability of long-term mission completion. Due to the need for more work to be completed in the field of AUV technology, there is a large range of work currently ongoing. Some of the more recent works in AUV technology are described here to give an example of the intense interest and importance of advancing AUV capabilities.

It has been shown that accurate underwater navigation within operational limits is possible. Healey and Lienard (1993) proved that for the combined speed, steering, and diving response of a slow moving AUV, multivariable sliding mode autopilots, based on feedback and the assumption of decoupled modeling was very satisfactory. Healey (1994) has achieved further developments in hover control behavior using the ST1000 and ST725 high frequency sonars to provide data about the environment. Marco and Healey (1996) demonstrated a method to navigate an AUV in a local area using an acoustic sensor for position information derived from feature detection. Marco (1996) produced a work which described the advantages of AUV's over ROV's or manned submarines, in which he designed and verified a working hybrid control system

combining mission management with robust motion controllers. And finally, Bellingham (1997), Smith (1995), and An (1998) have described the uses of an AUV for oceanographic survey and have given results on positioning accuracy for survey missions.

In the field of fault detection and resolution algorithms, the works cited here are recent studies into different fault detect methods and techniques. A good summary of some basic fault detection methods with some examples of detecting faults in an electrically driven centrifugal pump and detecting leaks for pipelines was done by Isermann (1984). Healey (1992) proposed the use of Extended Kalman Filters and Artificial Neural Networks to provide the detection and isolation of impending subsystem failures. Bell, et al. (1992) developed, evaluated and successfully tested a tool that automates the reasoning portion of a Failure Modes and Effects Analysis. Healey (1993) discussed the use of both batch least squares and Kalman Filters for system parameter identification as a means to detect performance change. Hurni (1997) used *Simulink* to model and simulate a tool for Failure Modes and Effects Analysis of the steering subsystem of an AUV. And finally, Melvin (1998) proposed the use of model-based observers for the detection of fault induced dynamic signals in the diving, steering, and roll control systems of the Naval Undersea Warfare Center's experimental "21Unmanned Underwater Vehicle (UUV)". A model was designed in *Simulink* and was used to simulate numerous vehicle behaviors and detect for faults in the control systems. Other works for applications of process control and aircraft flight control are discussed by Patton (1997) and Mangoubi (1998).

This work will concentrate on the steering subsystem fault detection of the 21UUV. The basis for fault detection lies in the generation of residuals being the difference in a sensor-measured value and a value estimated by the system model. For instance, if a control state were completely measurable by some sensor signal, $y(t)$, then its comparison to the model's estimated state, $\hat{y}(t)$, would produce a residual difference, $v(t)$, if the two were not of equal value. Written algebraically, a residual is simply represented as,

$$v(t) = y(t) - \hat{y}(t).$$

A simple graphical representation of this concept is shown in Figure 1.1.

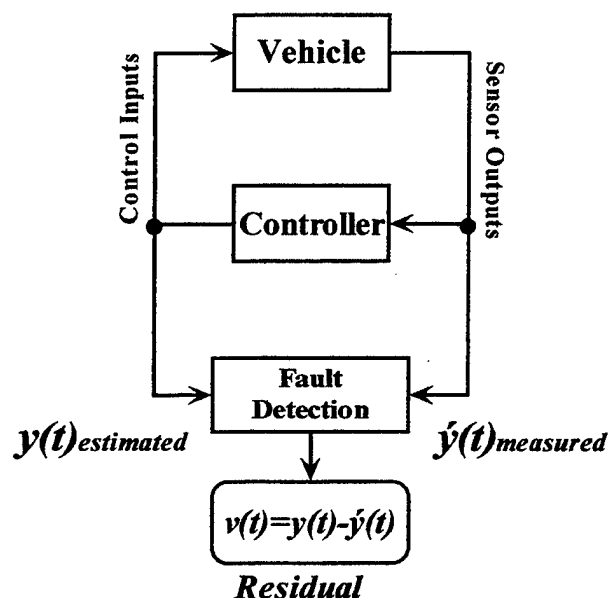


Figure 1.1 Residual Generation Defined

This basic definition of a residual is the founding concept of fault detection using model-based observer techniques. The understanding of this basic concept is vital to the purpose of improving upon the most recent fault detection and resolution methods.

Unfortunately, the most recent research into fault detection and resolution by use of model-based observer residual generation produces somewhat unsatisfactory results because of its inability to properly suppress the inherent residuals generated due to maneuvering and system noise. As mentioned previously, Melvin (1998) utilized model-based observed residual generation to model and simulate fault detection and resolution in the 21UUV. Without proper real-time data from the 21UUV, actual residual analysis was not possible. Upon implementation of real-time 21UUV data into the model-based design, it is found that manually introduced faults cannot be resolved from the residuals generated by basic maneuvering and system noise.

Due to the failure of this model-based observer design to adequately detect faults within the steering subsystem of the 21UUV, it is necessary to investigate the probable causes of this failure and to attempt to improve and/or eliminate them. Due to the complexity and exactness of the model of the 21UUV, it is possible that inaccuracies in the hydrodynamic coefficients that form the basis of the 21UUV model will introduce errors into the subsystem processing of residuals for maneuvering. Also of concern is the uncertainty in the system noise matrix, $\mathbf{Q}$, and the measurement noise matrix, $\mathbf{R}$, in the model-based observer. If chosen correctly, the $\mathbf{Q}$ and $\mathbf{R}$ matrices may significantly compensate for the maneuvering and system noise responsible for adding to the generation of excessive residuals in the steering subsystem. This thesis will then investigate the uncertainties in the hydrodynamic coefficients of the system model and

will couple the use of Kalman Filtering with model-based observer residual generation to accurately detect manually inputted faults in actual 21UUV data.

B. SCOPE OF THIS WORK

Due to the enormous amount of previous research conducted in the area of fault detection and resolution, it is noted that the problem in autonomous fault detection is very complex and intriguing. Due to the assistance of the Naval Undersea Warfare Center, the sensor measurements from an actual mission run of the 21UUV are available for this work. This thesis will have the distinct advantage of developing techniques and methods for fault detection and resolution that can be directly evaluated against actual performance parameters. The purpose of this thesis is four-fold:

1. To evaluate the performance of the previously developed model-based observer for residual generation of the 21UUV's steering subsystem. With the use of actual 21UUV data, manual faults will be implemented into the data run and it will be ascertained whether or not this model could successfully distinguish between a fault and a normal maneuver.
2. To optimize the uncertain hydrodynamic coefficients that define the dynamics and input matrices of the 21UUV's steering subsystem model. Utilizing the optimized hydrodynamic coefficients as evaluated over a given data interval, residual reduction will be quantified and further fault detection will be investigated.
3. To implement Kalman filtering into the steering subsystem residual generation process of the 21UUV. Relative error reduction will be quantified and fault detection will be investigated by use of this method. Optimization of the Q and R

matrices of the Kalman filter will be accomplished and the resulting relative error reduction will allow for accurate fault detection.

4. To implement a fault detection and resolution algorithm into the steering subsystem and evaluate the sensitivity and time lapse to detection of an actual fault.

Chapter II will explain the types of faults experienced in autonomous vehicle systems and will discuss the different methods for fault detection and diagnosis in a subsystem. Also included in this chapter will be the derivation of a comprehensive steering subsystem model-based observer for the 21UUV and its associated steering observer residual detector.

Chapter III will investigate the performance of the previously designed model-based observer for residual reduction of actual 21UUV sensor measurement data. Manually implemented faults will be evaluated in the generated residuals and a determination will be made whether fault detection is possible using this model design. Also included in Chapter III is a description of 21UUV steering dynamics and proposed fault detection architecture by Healey (1998).

Chapter IV will consist of the investigation of the uncertainties of the hydrodynamic coefficients forming the basis of the steering subsystem model of the 21UUV. An optimization of the control and input matrices of the steering subsystem will produce values of hydrodynamic coefficients that reduce the residual generation of the model-based observer design. Again, analysis will be performed on the ability of the improved model to detect faults in a given set of sensor measurement data.

Chapter V will introduce the use of Kalman filtering into the residual reduction of the steering subsystem. A performance index will be proposed which allows for the optimization of the **Q** and **R** matrices of the Kalman filter. By using the performance index, relative error magnitudes due to maneuvering will be reduced while relative error magnitudes due to faults will increase. Fault detection will be shown to be possible by the use of this optimized Kalman filter design.

Chapter VI will expound upon the fault detecting characteristics of the optimized Kalman filter design by using a fault detect algorithm to determine system sensitivity to imposed faults and subsequent time-to-detect for faults.

Chapter VII will contain conclusions of this work derived from Chapter's III, IV, and V and will provide recommendations for further study in this area of fault detection.

II. DESCRIPTION OF FAULTS AND ASSOCIATED DETECTION TECHNIQUES

Due to the difficult environment in which an autonomous vehicle may operate, it is necessary to have reliable and robust subsystems that are capable to accurately detect faults whenever present. The ability to detect faults will increase mission reliability by giving the autonomous system the opportunity to mitigate these faults on line and continue with its assigned mission. If a fault were left undetected, the degradation of mission performance would occur at a rate corresponding to the severity of the fault. Left alone over time, this degradation may lead to complete mission failure or even system loss, given a severe fault. The purpose of this chapter is to define various fault types and to describe methods of fault detection and diagnosis. Since the steering subsystem of the 21UUV will be studied in this work, a comprehensive model of the steering design will be included.

A. TYPES OF FAULTS

In the analysis of fault detection, two types of faults are identified as the majority of faults most common to subsystem failures. These two fault types are listed below.

1. Environmentally Induced Faults

Environmentally induced faults are faults that derive from varying signals caused by the effect of environmental conditions on the performance of the system. Such dynamic signals may arise from seaway wave action on the vehicle and the inability of

the sensor suite to accurately detect vehicle motion. These dynamic signal faults are commonly defined as disturbances and are not technically malfunctions in the subsystem.

2. System Induced Faults

System induced faults are faults that are incurred from hardware and software failures in the vehicle's subsystems. Hardware failures may include the loss of a fin or the disabling of a sensor. Software faults occur from the failure of the modeled system and its operational programming to execute according to design. Computer hardware configuration malfunction may also cause operational failure.

B. MAGNITUDE AND NATURE OF FAULTS

The magnitude and introduction aspect of a fault may be characterized by one of two aspects. One characterizing aspect of a fault is its incipient or developing nature over a long period of time. This aspect of a fault may arise as a result of a slow degradation in the performance of one of the vehicles measurement sensors. Dependent upon the degradation rate of the subsystem over time, the ability of the autonomous system to detect an incipient fault is difficult, at best. It is proposed that some graceful degradation of system performance may be allowable as long as partial subsystem control is maintained for the entirety of the mission.

The second characterizing aspect of a fault is the abruptness at which a signal varies in a short period of time. A large 'jump' in signal magnitude may be indicative of a sudden hardware failure or sudden loss in system control. Such faults are relatively easy to detect as long as the magnitude in the increased signal stays relatively large over a

set time period. Large spikes in signals may not be directly related to faults if the signal increase was due to an anomalistic reading of the sensor suite. These large spikes have to be filtered out of the overall signal response analysis in order to minimize the occurrence of "false detects".

C. TECHNIQUES FOR FAULT DETECTION AND DIAGNOSTICS

Fault detection and diagnostics may be classified into three categories: 1) Limits and Trends Analysis, 2) Model -Free Detection, and 3) Model-Based Detection. The following is a description of the three methods of fault detection and diagnostics.

1. 'Limits and Trends' Analysis

As described by Healey (1998), a survey of fault detection and diagnostic methods indicates that alarms can be easily monitored if signals remain static and slow changing throughout a defined time period. This is accomplished by using 'limits and trends' analysis. The actuation of an alarm or 'detect' occurs when a single signal exceeds a preset threshold. Once an alarm is actuated, information pertaining to the associated fault may be passed on to fuzzy logic inference systems for potential reconfiguration of the subsystems. An example is excessive motor temperature.

Unfortunately, the transient nature of dynamic signals makes limits and trends analysis invalid. Dynamic signals tend to exceed a threshold, but to later come back into range of preset bounds. This causes the use of thresholding alone to be insufficient for proper fault detection clarity. Dynamic signals that would produce such transients would include a broken fin, a sheered propeller shaft, or a ballasting failure.

2. Model-Free Detection

'Model-free' detection may be employed for certain dynamic signal analysis.

Model-free detection takes samples of the given dynamic signal and extracts constant features of the signal and compares them to preset threshold levels. Model-free methods are useful to detect frequency components in servo error signals and could be used to identify levels of seaway induced disturbances considered as faults, Newland (1993) & Healey (1998). Spectrum analysis and condition based monitoring are examples.

3. Model-Based Detection

Model-based detection utilizes the analysis of residuals produced from model-based observer design to detect the presence of a fault. Faults may arise from a fouled actuator, or a failed sensor. As previously described in Chapter I, a residual is the difference between a sensor measured value and a value estimated by the system model. By the generation of residuals, fault detection can be accomplished by analyzing the resultant residual value associated with particular state values of motion for the vehicle. Excessive residual generation may be deemed as the result of a fault in the subsystem.

Model-based methods have better ability to detect dynamic signals developed from autopilot errors. Autopilot errors tend to be large when steering to new course, but lessen when the vehicle achieves desired course. The residuals generated from model-based observers are not sensitive to servo errors caused by command changes and they respond primarily to non-ideal loads, disturbances from waves, and sensor signal errors, Melvin (1998).

D. STEERING SUBSYSTEM MODEL-BASED OBSERVER DESIGN FOR 21UUV

The steering subsystem and associated model-based observer of the 21UUV are designed using the methods outlined in Healey (1995) and Healey (1998). Due to the analysis of work previously shown by Melvin (1998), the following steering observer residual detector theory and application are taken from his work in order to preserve continuity and substantiate this work's claim on performance inadequacies in fault detection for that given design.

Although this work concentrates mostly on the steering subsystem of the 21UUV, it is necessary to state the assumption that the 21UUV is controlled by four main subsystems, which are uncoupled, and use six degrees of freedom. This defines four autopilot controllers – the steering, diving, roll, and speed control systems. Consequently, there are four observer based residual generators, with one generator for each controller. Each observer based residual generator would generate and process residuals for each corresponding subsystem.

Each subsystem is modeled as a non-interacting Linear Time Invariant system:

$$S:(A,B,C,D) \in \mathbb{R}^{n \times n}.$$

1. Theory

The steering subsystem dynamics for the 21UUV are modeled by the following equations of motion:

$$\begin{aligned}\mathbf{x}'(t) &= [\hat{v}_r(t), \hat{r}(t), \hat{\psi}(t)]; \\ u(t) &= \delta_s(t) \\ \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{E}\mathbf{f}_s(t) + \mathbf{F}\mathbf{d}(t); \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{f}_s(t);\end{aligned}$$

The state variables v_r , r , and ψ are the vehicles' sway velocity (side slip), yaw rate, and heading angle, respectively. $\mathbf{B}$ and $\mathbf{E}$ are the input vectors for the control planes and $\mathbf{F}$ is the input vector for disturbances from waves and currents. The variable $\delta_s(t)$ is the steering command input and $\mathbf{f}_s(t)$ represents added forces caused by sensor errors. We assume that the inertial system of the 21UUV is of high quality and all state variables are measured with little noise. The output matrix, $\mathbf{C}$, is then taken as identity.

A model based observer can be formed from the given idealized subsystem dynamics:

$$\begin{aligned}\hat{\mathbf{x}}'(t) &= [\hat{v}_r(t), \hat{r}(t), \hat{\psi}(t)]; \\ u(t) &= \delta_s(t) \\ \hat{\dot{\mathbf{x}}}(t) &= (\mathbf{A} - \mathbf{K}\mathbf{C})\hat{\mathbf{x}}(t) + \mathbf{B}u(t) + \mathbf{K}\mathbf{y}(t); \\ \mathbf{v}(t) &= \mathbf{y}(t) - \mathbf{C}\hat{\mathbf{x}}(t).\end{aligned}$$

The residuals are represented by the vector $\mathbf{v}(t)$ and are the differences between the sensor measured values and model-based predictions for side slip, yaw rate, and

heading states. A state observation error, ε_x , can then be defined as the difference between the fully measurable state equation and the model-based prediction state equation.

$$\begin{aligned}\dot{\varepsilon}_x(t) &= (\dot{\mathbf{x}} - \hat{\dot{\mathbf{x}}}) = \{\mathbf{A}\dot{\mathbf{x}}(t) - \mathbf{A}\hat{\dot{\mathbf{x}}}(t)\} + (\mathbf{B} - \hat{\mathbf{B}})u(t) + \mathbf{E}\mathbf{f}_s(t) + \mathbf{F}\mathbf{d}(t) - \mathbf{K}\{\mathbf{y}(t) - \mathbf{C}\hat{\mathbf{x}}(t)\} \\ y(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{f}_s; \\ \dot{\varepsilon}_x(t) &= (\mathbf{A} - \mathbf{K}\mathbf{C})\varepsilon_x(t) + \mathbf{E}\mathbf{f}_s(t) + \mathbf{F}\mathbf{d}(t) + \mathbf{K}\mathbf{f}_s(t) \\ \text{where} v(t) &= \mathbf{C}\varepsilon_x(t) + \mathbf{f}_s(t).\end{aligned}$$

The residual generation system may be viewed as a system subject to $u(t)$ and $y(t)$ as inputs with $v(t)$ as output such that it has a system transfer function:

$$v(s) = \mathbf{C}[\mathbf{sI} - (\mathbf{A} - \mathbf{K}\mathbf{C})]^{-1} \{\mathbf{E}\mathbf{f}_s(s) + \mathbf{F}\mathbf{d}(s) + \mathbf{K}\mathbf{f}_s(s)\} + \mathbf{f}_s(s).$$

Note, if $\mathbf{C}\mathbf{A}_c^{-1}\mathbf{E} = \mathbf{0}$, then $\mathbf{f}_s$ does not appear in $v(s)$ and $\mathbf{f}_s$ are undetectable in $v(s)$.

Also, if $\mathbf{E}=\mathbf{F}$; $\mathbf{f}_s$ and $\mathbf{d}$ are indistinguishable; see Patten and Chen (1998).

2. Application

For a slower speed of 6 feet per second, the model of the 21UUV (to be described in detail in Chapter III) has dynamics (**A**) and input matrices (**B**) of:

$$\mathbf{A} = \begin{bmatrix} -0.1140 & -2.3282 & 0 \\ -0.0649 & -0.3015 & 0 \\ 0 & 1 & 0 \end{bmatrix};$$
$$\mathbf{B} = \begin{bmatrix} 0.3308 \\ -0.1224 \\ 0 \end{bmatrix}.$$

For the example used in Melvin (1998), the wave amplitude was set at 2ft, the autopilot for depth control was of sliding mode design, and the placed poles included a single pole at the origin. The resulting gains for the sliding mode controller were:

$$\lambda = [-0.4 \quad -0.41 \quad 0];$$
$$k = [0.5762 \quad -1.6663 \quad 0];$$
$$s' = [0.0164 \quad 0.8804 \quad 0.4740].$$

The *Matlab* command 'place' was used instead of Linear Quadratic Estimation in order to ensure real numbers were generated for the observer poles vice complex poles and eigenvectors. The observer poles were placed close to the control poles, [-0.2, -0.21, -0.22]. The observer poles were found to be:

$$\mathbf{K}_o = \begin{bmatrix} 0.0860 & -2.3282 & 0 \\ -0.0649 & -0.0915 & 0 \\ 0 & 1 & 0.2200 \end{bmatrix}.$$

The observer was modeled by the state-space equation:

$$\begin{aligned}
 \mathbf{x}'(t) &= \mathbf{A}_o \mathbf{x}(t) + \mathbf{B}_o \mathbf{u}_o(t) \\
 \mathbf{v}(t) &= \mathbf{C}_o \mathbf{x}(t) + \mathbf{D}_o \mathbf{u}_o(t) \\
 &\text{with.....} \\
 \mathbf{u}_o &= [\delta_s, y]' \\
 \mathbf{A}_o &= \mathbf{A} - \mathbf{K}_o' \mathbf{C} \\
 \mathbf{B}_o &= [\mathbf{B} \quad \mathbf{K}_o'] \\
 \mathbf{C}_o &= -\mathbf{C} \\
 \mathbf{D}_o &= [\text{zeros}(3,1) \quad \text{eye}(3,3)]
 \end{aligned}$$

The Matlab file developed and used by Melvin (1998) to generate the steering observer and steering observer residual detector, "steer_obs_des.m", is included in Appendix A.

3. The Effect Of Model Uncertainty

In the above analysis, it is assumed that the system model is perfect. This means that the true $\mathbf{A}$, $\mathbf{B}$ pair for the vehicle is indeed the $\mathbf{A}$, $\mathbf{B}$ pair used to generate the residual. This is not likely to be the case and if we define

$[\mathbf{A}, \mathbf{B}] \Rightarrow$ true system pair, and

$[\hat{\mathbf{A}}, \hat{\mathbf{B}}] \Rightarrow$ model-used pair, then

the residual is now defined by

$$v(s) = C[sI - \hat{A}_c]^{-1} \{E f_a(s) + \delta A \hat{x}(s) + \delta B \delta_s(s) + f_d(s) + K f_s(s) + f_s(s)\}$$

where

$$\delta A = (A - \hat{A}),$$

$$\delta B = (B - \hat{B}).$$

The problem lies in finding an $\hat{A}_c$ for the residual generator so that

$E f_a(s) \neq \delta A \hat{x}(s) + \delta B \delta_s(s)$. Defining,

$$v_1 = C[sI - \hat{A}_c]^{-1} E f_a(s)$$

$$v_2 = C[sI - \hat{A}_c]^{-1} [\delta A \hat{x}(s) + \delta B \delta_s(s)],$$

we see that v_2 is mostly driven by maneuvering where $\delta_s, \hat{x}(s)$ are non-zero while v_1 is the residual generated by an actual fault. Distinguishing v_1 and v_2 is not easy unless

$$\|\delta \hat{A} \hat{x}(s)\| \ll \|E f_a(s)\|.$$

It is the objective of this work to minimize $\|E f_a(s)\| - \|\delta \hat{A} \hat{x}(s)\|$ for all $\hat{x}(s)$ during maneuvers.

III. MODEL- BASED OBSERVER PERFORMANCE USING ACTUAL 21UUV DATA

A. FAULT DETECTION ARCHITECTURE FOR 21UUV

Healey (1998) has proposed a fault detection architecture for 21UUV. This architecture will be briefly described here in order to show existing techniques that will utilize this work's advancements in fault detection and resolution.

The proposed fault detection architecture is based upon using subsystem detection circuits to look for fault signals of specific magnitude and duration. If both magnitude and duration levels exceed threshold levels, the fault detector declares a fault. This architecture will respond appropriately to mitigate the fault by linking the associated fault signal with pre-set response actions guided by fuzzy logic methodology.

Robustness of fault detection is increased significantly when residuals produced from multiple sources are compared together for an overall assessment of system health. The sources generating residuals for this architecture are the fin stroke detectors, servo error detectors, observer residual detectors, and wave activity detectors. Measurements are produced from the vehicle's sensor suite and fed back into the controller and associated fault detectors. The controllers produce control inputs to the vehicle and also send control inputs to the fault detectors. The fault detectors take the inputs from the controllers and compare them with sensor outputs to produce residuals. The fault detectors analyze the resultant residuals and determine if a fault is present. Fault signal attributes are then transferred to the fuzzy inference system. The fuzzy inference system makes 'judgments' based on fuzzy logic rules as to the severity of the fault and promulgates command adjustments to the controllers in order to compensate for the fault.

If the fault is judged to be 'too severe', the fuzzy inference system transfers health assessment responsibility to a higher level of authority. An illustration of the proposed fault detection architecture for 21UUV is shown in Figure 3.1.

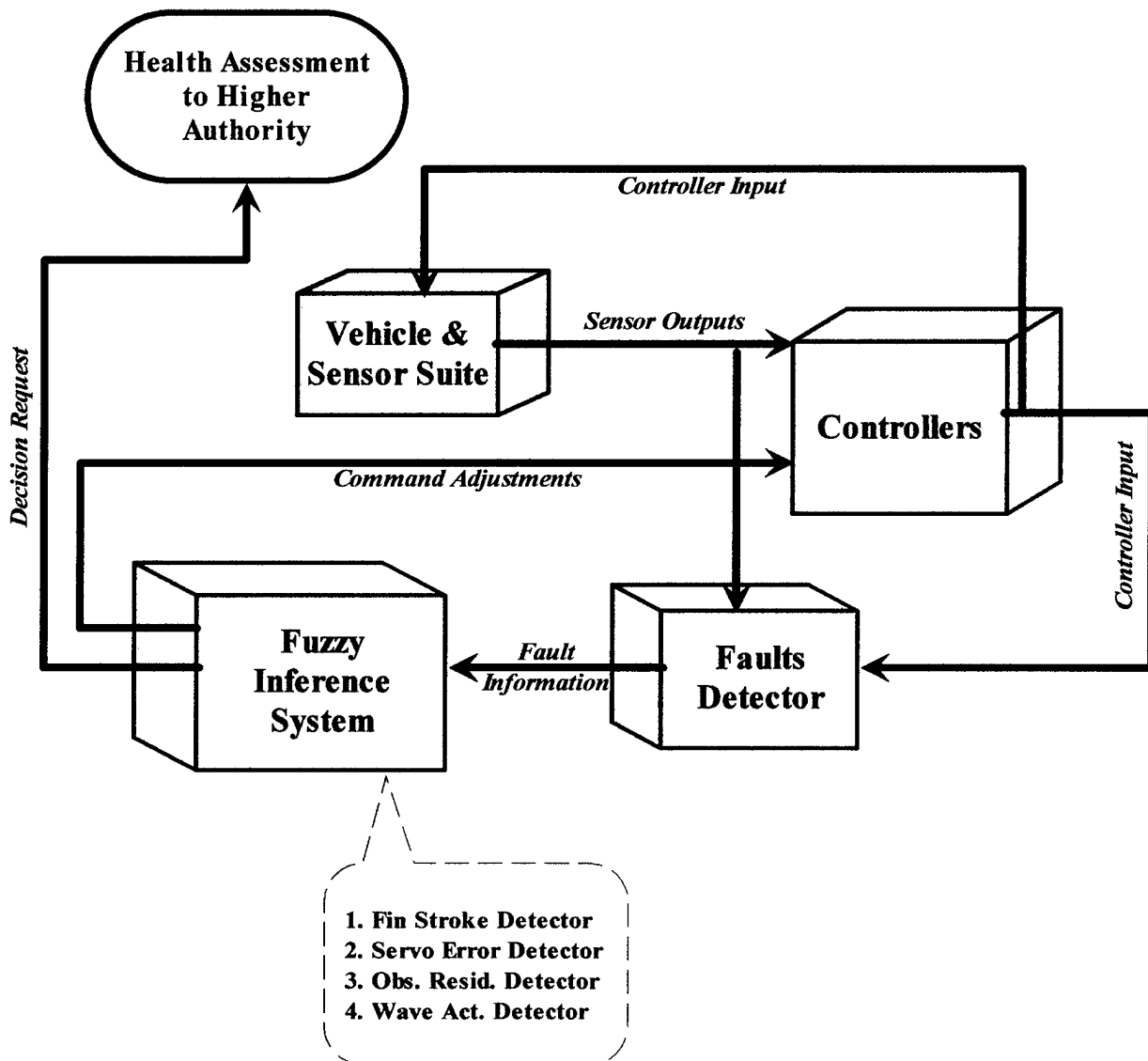


Figure 3.1 Fault Detection Architecture w/ Fuzzy Inference System for Fault State Resolution (Healey, 1998)

B. DESCRIPTION OF 21UUV MODEL

1. Development of 21UUV Steering Model

As with all vehicles that 'fly' through a given fluid medium, there exist specific equations of motion defining the maneuvering and motion control of autonomous vehicles. For this work and the modeling of steering control for 21UUV, the following are considered:

1. 21UUV behaves as a rigid body
2. The earth's rotation is negligible compared to that of the vehicle when defining inertial acceleration components of the vehicle's center of mass
3. The primary forces that act on the 21UUV have inertial and gravitational origins
4. For marine vehicles, other sources of force are hydrostatic, propulsive, thruster, and hydrodynamic forces from lift, and added mass

For the simplified case of rigid body motion for the steering model, we ignore the vertical plane of motion. In so doing, we significantly simplify the equations of motion (EOM) by setting the following to zero: (see Healey 1995, ME 4823 notes for nomenclature, {<http://web.nps.navy.mil/~me/healey/papers/ME4823.pdf>})

$$\begin{aligned}w_r &= 0, \\p &= 0, \\q &= 0, \\Z &= 0, \\\phi &= 0, \\\theta &= 0.\end{aligned}$$

The resulting motions of interest for the steering model become $[u_r, v_r, r]$. For nominal steady state conditions with steady forward motion, we can assume $u_r = U_o$, the forward speed of the vehicle.

The EOM then become (Healey 1995):

$$\begin{aligned} u_r &= U_o \\ m\dot{v}_r &= -mU_o r + \Delta Y_f(t) \\ I_z \dot{r} &= \Delta N_f(t) \\ \dot{\psi} &= r \\ \dot{X} &= U_o \cos \psi - v_r \sin \psi + U_{cx} \\ \dot{Y} &= U_o \sin \psi + v_r \cos \psi + U_{cy} \end{aligned}$$

Through the assumption of ‘small’ motions, the fluid forces under the conditions of ‘flight’ are linearized using a Taylor series expansion to produce body force ‘hydrodynamic coefficients’. These hydrodynamic coefficients depend on the shape characteristics of the vehicle and can determine the vehicle’s natural stability of motion. These coefficients are often assumed to be constant, but this assumption has limited applicability.

The primary dynamics equations for steering and directional stability of the vehicle include the sway, v , and yaw, r , motions. If we neglect the effects of surge motion changes and roll motion coupling and use the linearized constant coefficient force model, we get the dynamic response and the path of the vehicle:

Dynamic Response of the Vehicle

$$m\dot{v}_r = -mU_o r + \dot{Y}_{\dot{v}_r} \dot{v}_r + Y_{v_r} v_r + Y_{\dot{r}} \dot{r} + Y_r r + Y_{\delta} \delta_r(t)$$

$$I_z \dot{r} = N_{\dot{v}_r} \dot{v}_r + N_{v_r} v_r + N_{\dot{r}} \dot{r} + N_r r + N_{\delta} \delta_r(t)$$

Path of the Vehicle

$$\dot{\psi} = r$$

$$\dot{X} = U_o \cos \psi - v_r \sin \psi + U_{cx}$$

$$\dot{Y} = U_o \sin \psi + v_r \cos \psi + U_{cy}$$

The state vector for the steering subsystem model can be written as:

$$\mathbf{x} = [v, r, \psi].$$

The matrixes form of the sway and yaw equation coupled with heading can be written in expanded definition as:

$$\begin{bmatrix} m - Y_{\dot{v}_r} & -Y_r & 0 \\ -N_{\dot{v}_r} & I_z - N_r & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v}_r \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} Y_{v_r} & Y_r - mU_o & 0 \\ N_{v_r} & N_r & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v_r \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} Y_{\delta} \\ N_{\delta} \\ 0 \end{bmatrix} \delta_r(t)$$

$\begin{matrix} (3 \times 3) & & (3 \times 1) & & (3 \times 3) & & (3 \times 1) & & (3 \times 1) \\ \mathbf{M} & & \dot{\mathbf{x}} & & \mathbf{AA} & & \mathbf{x} & & \mathbf{BB} \end{matrix}$

$$\mathbf{y} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ r \\ \psi \end{bmatrix}$$

$\begin{matrix} (3 \times 3) & & (3 \times 1) \\ \mathbf{C} & & \mathbf{x} \end{matrix}$

For the 21UUV, the normalized hydrodynamic coefficients listed in the above state matrixes are given as:

$$\begin{aligned}
 m &= 88.9518 \text{slugs}; \\
 I_{zz} &= 2632.47 \text{slugs} \cdot \text{ft}^2 \\
 Y_{\dot{v}_r} &= 1.041e-02; \\
 Y_{\dot{r}} &= 1.753e-04; \\
 N_{\dot{v}} &= 1.753e-04; \\
 N_{\dot{r}} &= -7.504e-04; \\
 Y_{v_r} &= -7.406e-03; \\
 Y_r &= 2.655e-03; \\
 N_{v_r} &= -6.746e-03; \\
 N_r &= -1.477e-03; \\
 Y_{\delta} &= 4.216e-03; \\
 N_{\delta} &= -2.176e-03.
 \end{aligned}$$

The values m and I_{zz} are mass properties, while the remaining values are normalized taken from Healey (1995). The steering subsystem is modeled as an independent system following the equation form:

$$\begin{aligned}
 \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{u}) \\
 \mathbf{y} &= g(\mathbf{x})
 \end{aligned}$$

Represented in state-space form with linearized characteristics, the steering subsystem appears as:

$$\dot{\mathbf{x}} = [\underset{A}{M^{-1}AA}]\mathbf{x} + [\underset{B}{M^{-1}BB}]u$$

with...

M=mass matrix;
AA=state matrix;
BB=control matrix;

and...

$$A = [\underset{(3 \times 3)}{M^{-1}AA}]$$

$$B = [\underset{(3 \times 1)}{M^{-1}BB}]$$

2. 21UUV X-fin Configuration

It is necessary at this point to give a quick description of the X-fin Configuration of the 21UUV. The 21UUV utilizes a four fin configuration in the shape of an 'X' to perform all manners of maneuvering. Figure 3.2 is a simple schematic of the stern view of the fins and their numbering sequence.

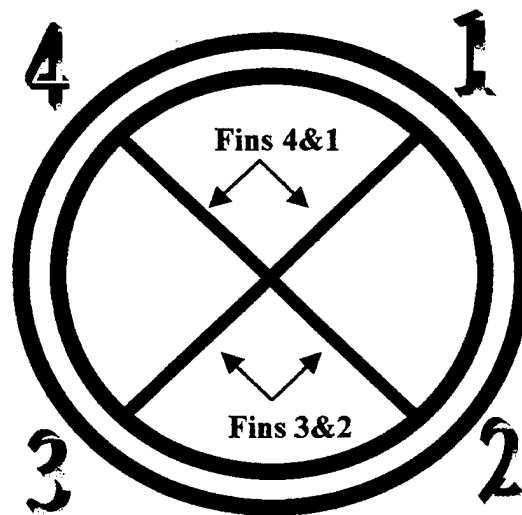
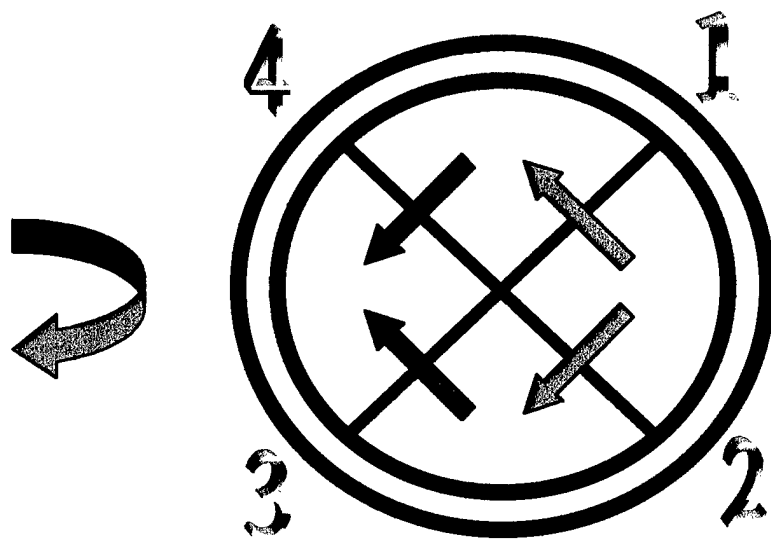


Figure 3.2 Stern-view Aspect of X-Fin Configuration on 21UUV

The fins constitute the control planes for all the diving, steering, and rolling maneuvers of the vehicle. The fins are coupled together by gain matrixes in the control block of the vehicle. The fins are coupled together by gain matrixes in the control block of the vehicle's steering subsystem. Appropriate signals are generated which initiate rotation of each fin in a manner to produce the desired maneuver. An example of fin deflection for a steering command is shown in Figure 3.3. The positive steering command, δ_r , calls for a turn to port (left-hand turn). Arrows represent the directional forces on the fins. The arrows are drawn across the fin in the direction of acting force. The vertical components of force acting on the fins are canceled out due to the cancellation of static forces in opposition in the vertical plane. The remaining horizontal force components add together to produce a resulting turn to port.



$\delta_r = \text{Port Turn}$

Figure 3.3 Steering Command Response to Port Turn

For a turn to starboard, the fins would react in the opposite direction in order to produce horizontal fin forces acting in the starboard direction. The individual fin commands are obtained through the multiplication of the generalized fin commands with the following gain matrix,

$$\begin{array}{c} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \\ \text{Individual} \\ \text{Fin} \\ \text{Commands} \end{array} = \begin{array}{c} \begin{bmatrix} 1 & -1 & 1 \\ 1 & 1 & 1 \\ -1 & 1 & 1 \\ -1 & -1 & 1 \end{bmatrix} \\ K\text{-Gain} \\ \text{Matrix} \end{array} \begin{array}{c} \begin{bmatrix} \delta_s \\ \delta_r \\ \delta_a \end{bmatrix} \\ \text{Generalized} \\ \text{Fin} \\ \text{Commands} \end{array} .$$

Note also, that in order to obtain the generalized fin commands the individual fin commands are multiplied by the transpose of the gain matrix as follows,

$$\begin{array}{c} \begin{bmatrix} \delta_s \\ \delta_r \\ \delta_a \end{bmatrix} \\ \text{Generalized} \\ \text{Fin} \\ \text{Commands} \end{array} = \begin{array}{c} \begin{bmatrix} 0.25 & 0.25 & -0.25 & -0.25 \\ -0.25 & 0.25 & 0.25 & -0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{bmatrix} \\ \text{Transpose}(K) \end{array} \begin{array}{c} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \\ \text{Individual} \\ \text{Fin} \\ \text{Commands} \end{array} .$$

C. 21UUV DATA ANALYSIS

1. Parameters and Characteristics of the Run

The 21UUV data used in this work was graciously provided for by the Naval Undersea Warfare Center. The 21UUV data run is comprised of a myriad of sensor measurements taken over the course of a 36.5-minute exercise. The 21UUV was capable of recording measurements every 0.1seconds throughout the run. The vehicle generated

over 21900 data points for the overall run. The track of the vehicle reveals multiple course changes that allowed for successive waypoint interception throughout the area of operation. The course legs tend to overlap themselves as the vehicle goes from one waypoint to another. Of interest for this work was the vehicle's ability to accurately record the vehicle's heading, ψ , the vehicle's yaw rate, r , and the vehicle's sideslip velocity, v . These three vehicle motions form the basis for the state vector comprising the vehicle's steering subsystem. With these three measurements and the velocity of the vehicle, residuals may be generated using the previously developed model-based observer method.

2. Track Analysis of the Run

In order to properly analyze the data obtained from the 21UUV's run, the data must first be 'processed' to provide useful information. It is very important to first convert all measurements into radian form. This is an obvious step, but if not completed, the analysis of the data would be in error. Also of extreme importance in the analysis of the data is the 'Rap Count' measurement performed by the vehicle. As the vehicle maneuvers through its expansive run, its heading measurement often reaches a minimum angle of -180° or a maximum angle of $+180^\circ$. If the heading of the vehicle approaches either maximum or minimum values of $+180^\circ$ or -180° and continues through these values of heading, the measurement will leap 360° and continue in the same direction but from the opposite heading value (either $+180^\circ$ or -180°). This sudden leap in heading measurement tends to cause the generated residuals for heading to increase significantly. This is due to the steering observer's 'estimation' that the next measurement will be

continuous and proceed in the direction of the previous measurement and not leap 360° in such an abrupt manner. This abrupt measurement characteristic of the measurement suite on 21UUV is shown in Figure 3.4. Here, an unprocessed measurement data set for heading is displayed over its respective time interval of 1300-1500 seconds. As can be seen from the figure, a sudden and abrupt leap in heading measurement often occurs when the vehicle maneuvers beyond $+180^\circ$. This sudden leap is very detrimental to any residual analysis of the data and must be corrected if useful residual generation is to be performed.

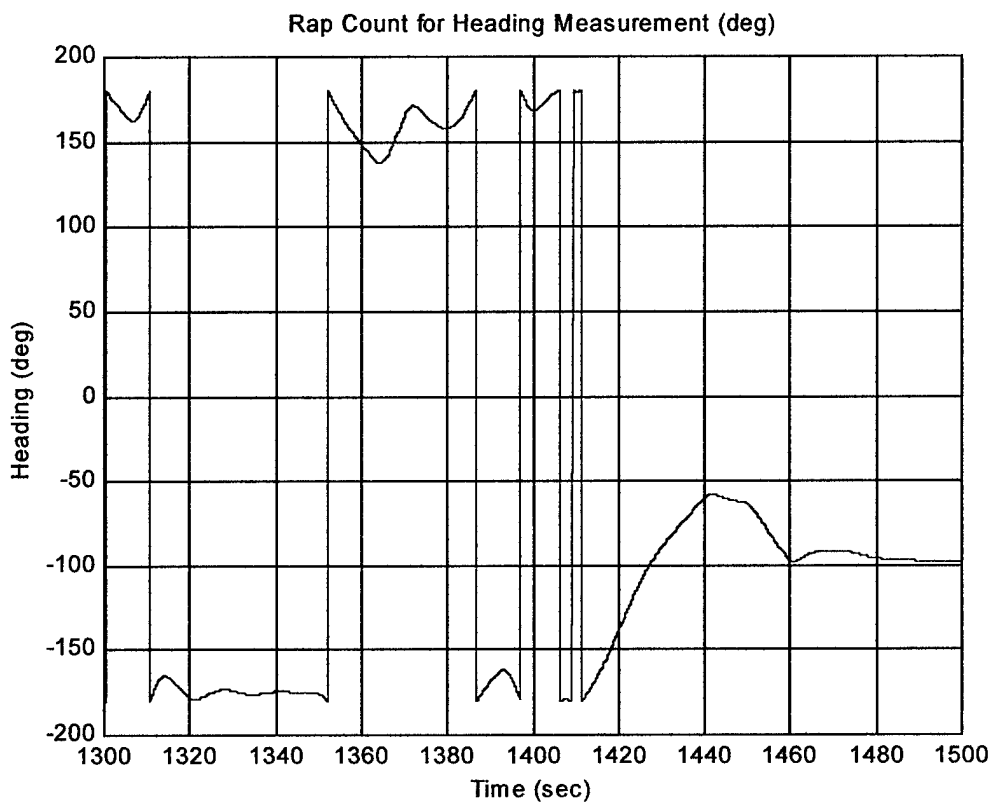


Figure 3.4 Rap Count Analysis of Heading Data

Note for Figure 3.4: Source Code Name – "rap_count.m"

A simple *Matlab* code is implemented as a preprocessor for all subsequent model-based observer and residual generation analysis in order to alleviate the problem of rap counting. This code, “rap_count.m”, is found in Appendix A. Figure 3.5 shows the same data interval for heading plotted with rap counting removed and superimposed over the original rap counted data measurements. As can be seen from this figure, the resultant heading measurement plot is continuous and would not cause any excessive residual generation due to abrupt changes from not rap counting.

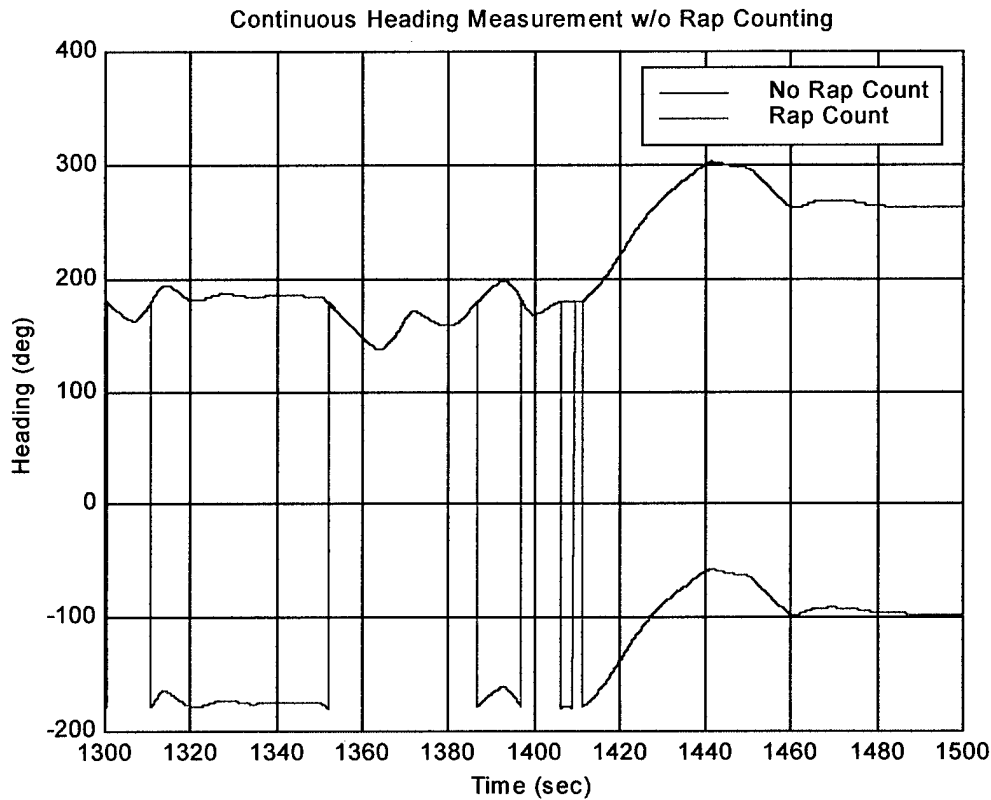


Figure 3.5 No Rap Count Plot Compared w/ Rap Count Plot

Note for Figure 3.5: Source Code Name – “rap_count.m”

The actual longitudinal and latitudinal track characteristics may be reconstructed for proper analysis of the vehicle's run by use of dead-reckoning techniques. Using a *Matlab* code called "dead.m", which is contained in Appendix A, the track is plotted as shown in Figure 3.6. It is obvious from this plot that the vehicle drove to numerous waypoints in the execution of the run. The overshoot for each turn of the vehicle is also apparent from the plot. Since residual generation tends to break down in areas of high maneuvering, Figure 3.6 will be used to isolate five locations for further residual analysis using model-based observer techniques.

The data is initially broken down into 'legs' containing approximately 1500 data points each. The legs are delineated with specific markers in order to make tracing the vehicle's run easier. Along with the arrows indicating the direction of the vehicle, the plot's legend specifies which marker relates to which leg of the 21900 data point run.

The track analysis begins at data point 3000 because data prior to this point was taken while the vehicle was without forward motion. The plotted track of data set 1-3000 shows errant behavior on the part of the vehicle and it actually displays backward motion of the vehicle at given times. The purposeful withdrawal of this data interval has no impact whatsoever on the analysis of fault detection for the remainder of the run.

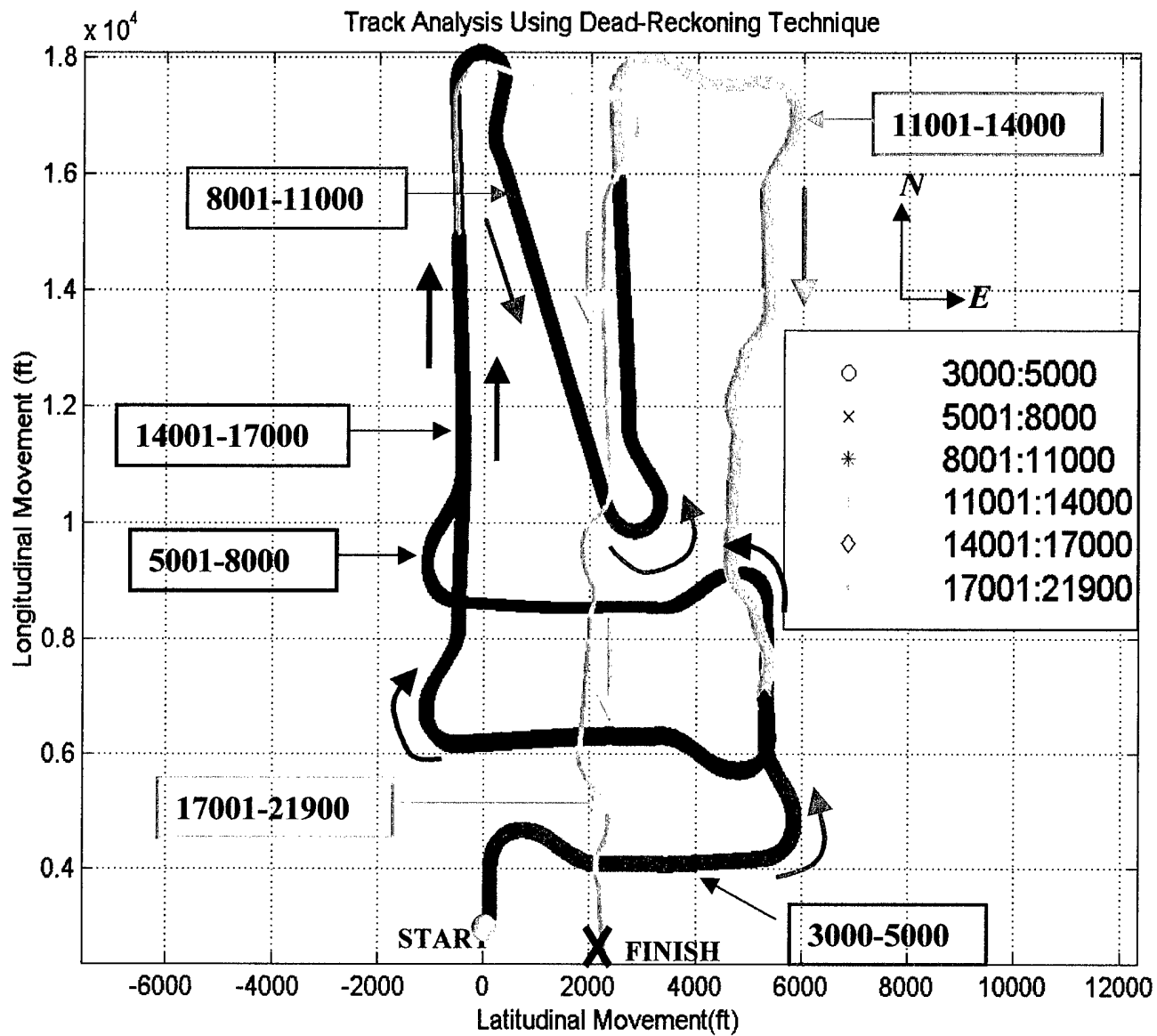


Figure 3.6 Dead-Reckoning Solution of 21UUV Track

Note for Figure 3.6: Source Code Name – “dead.m”

3. Data Sets for Residual Analysis (Maneuvering Specific)

As noted earlier, a common problem with using residual generation for fault detection is the increase in residuals during maneuvers. The purpose of a model-based observer is to estimate the next set of state variables of motion during the vehicles run. Given this task, it is very difficult for the observer with model errors to accurately estimate states that are rapidly changing due to large changes in heading. Due to the inability of the observer to exactly estimate the next value for the state variables, the resultant residuals tend to increase throughout and shortly after the performance of a maneuver. This increase in residuals from v_2 (defined in the last chapter) makes subsequent fault detection difficult. With large increases in the residual generation of the steering subsystem during maneuvering, false-fault detects are very common using current fault detection schemes. This being the case, it is imperative that this work concentrates on the specific problem of detecting faults throughout maneuvering specific intervals in the vehicle's run. If accurate and reliable fault detection can be accomplished during vehicle maneuvering, satisfactory fault detection for the steering subsystem can be proven to be attainable.

For this work, five intervals of data will be analyzed in order to properly design fault detection techniques. Of the five intervals, four will include large changes in heading and one will encompass an interval where constant heading was maintained for a specified distance. The intervals of evaluation will be taken from the original track and the values of the state variables for the steering subsystem (sideslip velocity, yaw rate, and heading) will be graphed to display behavior characteristics over the course of the data interval.

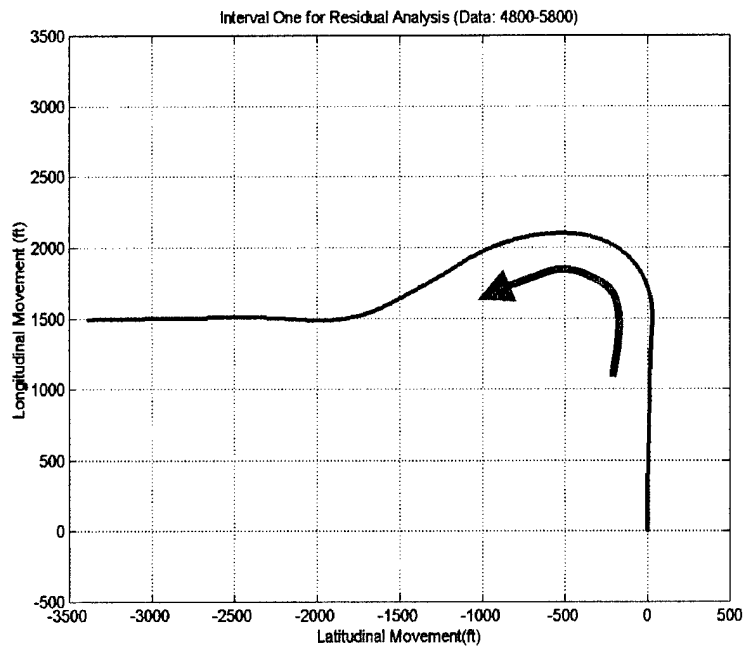


Figure 3.7 Data Set Interval One

Note on Figure 3.7: Source Code Name – “dead_int1.m”

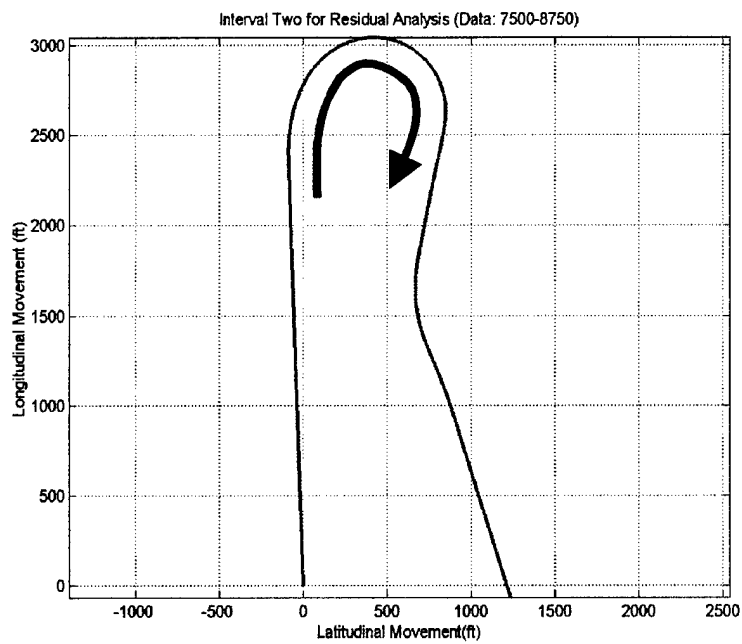


Figure 3.8 Data Set Interval Two

Note on Figure 3.8: Source Code Name – “dead_int2.m”

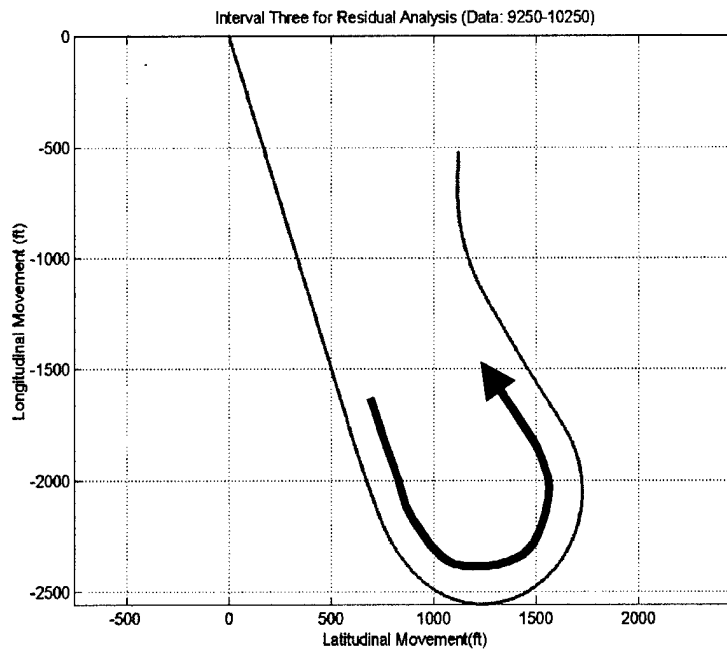


Figure 3.9 Data Set Interval Three

Note on Figure 3.9: Source Code Name – “dead_int3.m”

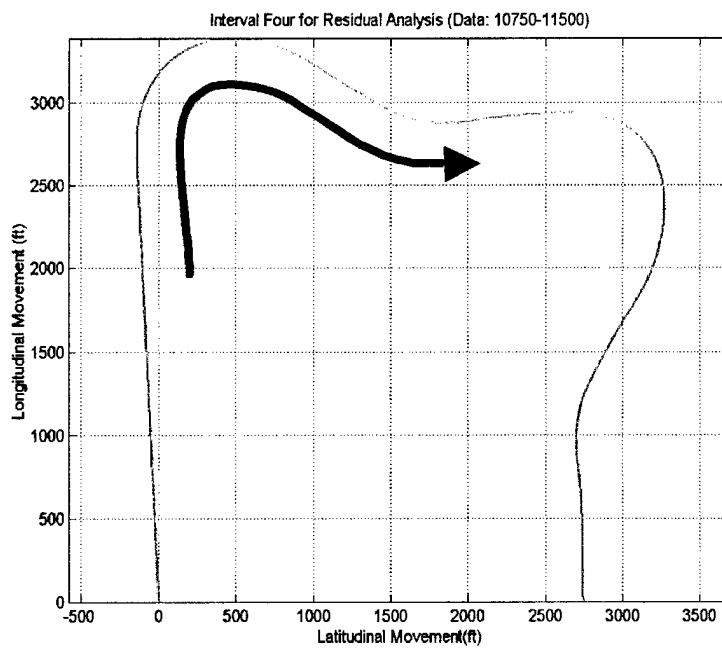


Figure 3.10 Data Set Interval Four

Note on Figure 3.10: Source Code Name – “dead_int4.m”

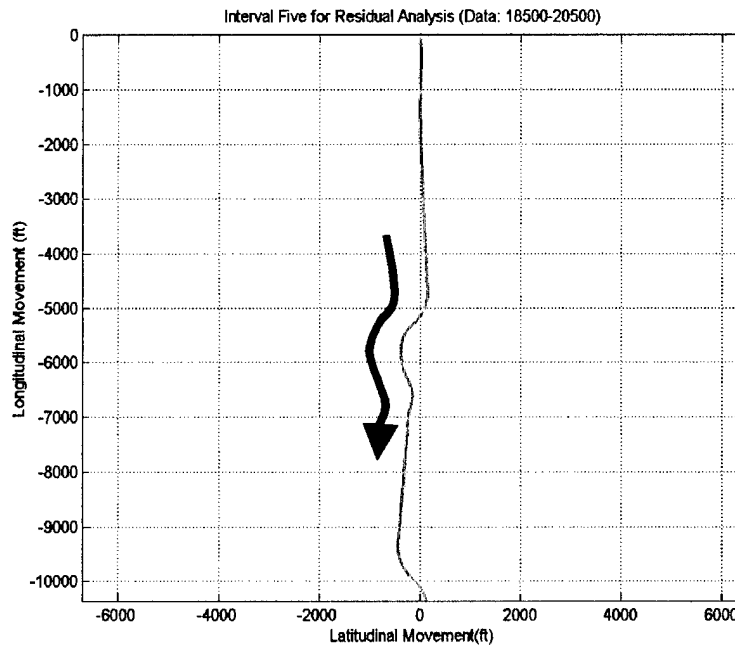


Figure 3.11 Data Set Interval Five

Note on Figure 3.11: Source Code Name – “dead_int5.m”

Each data set interval has its own characteristic state variable response. Figures 3.12, 3.13, and 3.14 display the sideslip velocity, yaw rate, and heading of each data set interval, respectively. As can be seen from these plots, there is significant variation and fluctuation in the state variables that is resultant of the high degree of maneuvering being conducted by the vehicle. These figures are included here in order to show the tremendous task that must be accomplished in finding faults within the given data set intervals.

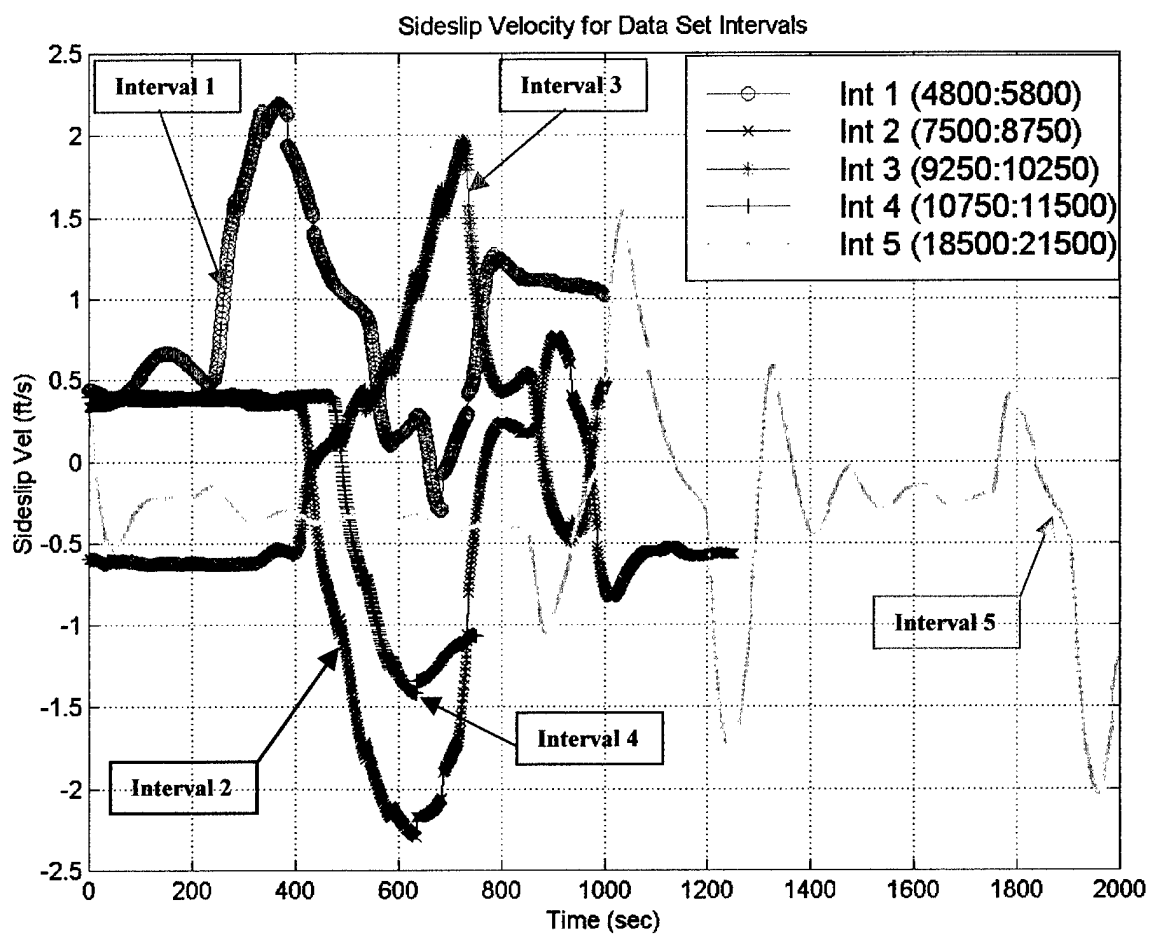


Figure 3.12 Sideslip Velocities for Each Data Set Interval

Note for Figure 3.12: Source Code Name – “state_resp.m”

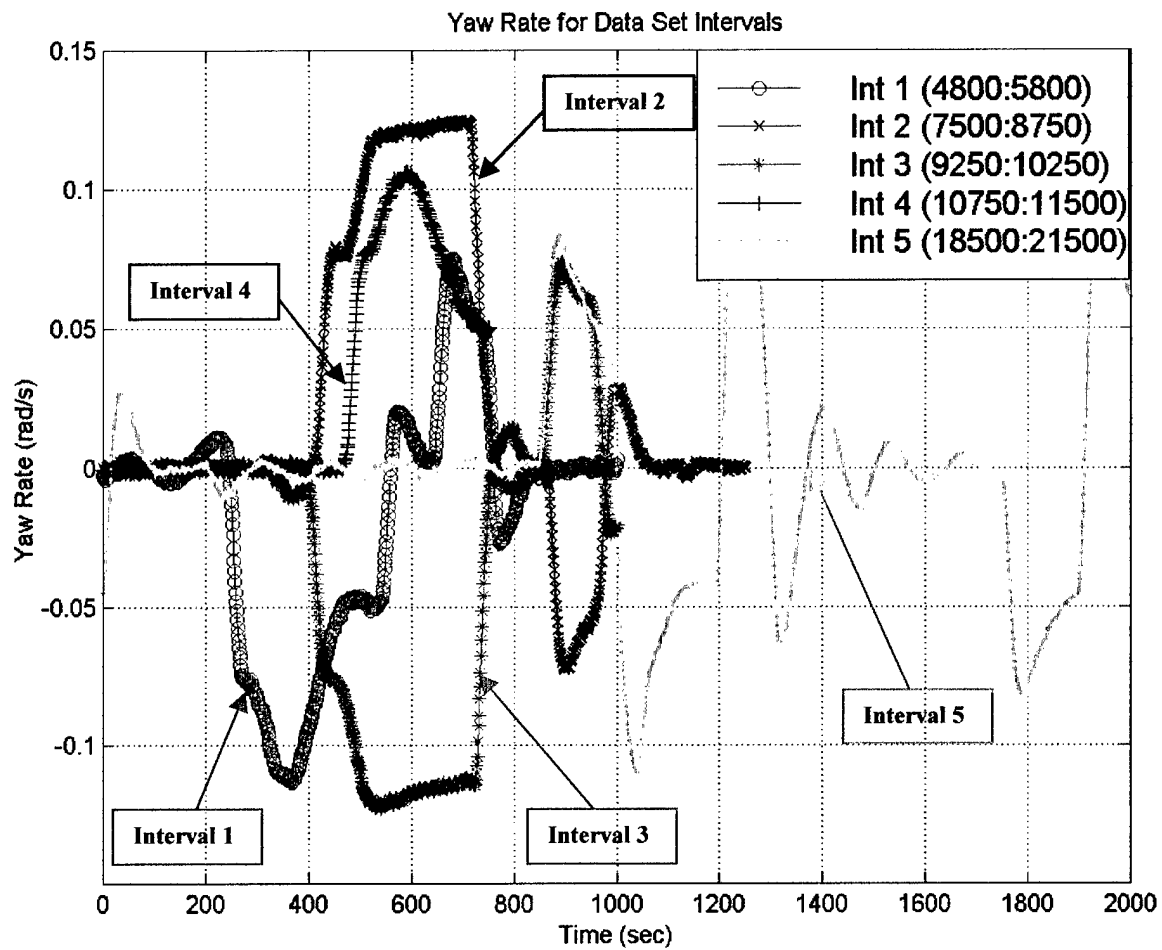


Figure 3.13 Yaw Rates for Each Data Set Interval

Note for Figure 3.13: Source Code Name – “state_resp.m”

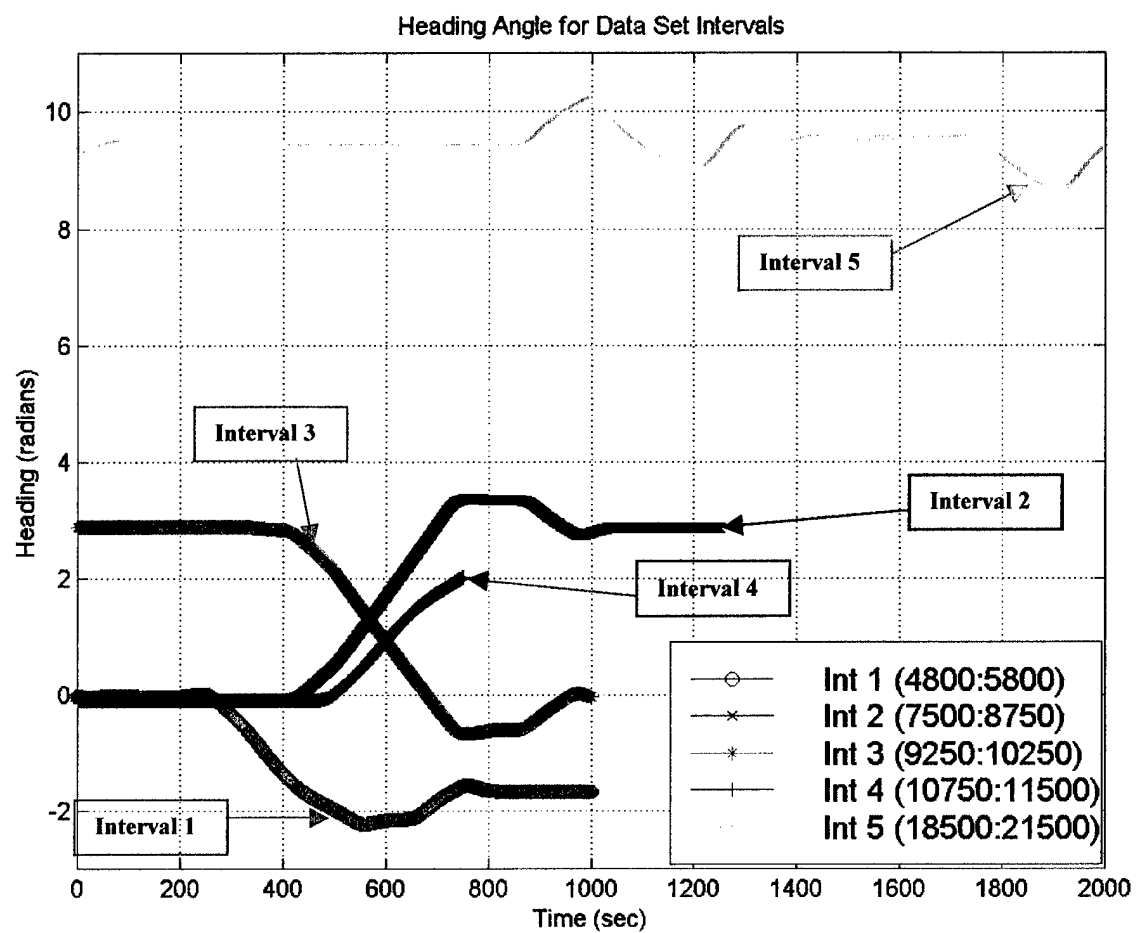


Figure 3.14 Heading Angles for Each Data Set Interval

Note for Figure 3.14: Source Code Name – “state_resp.m”

D. RESIDUAL GENERATION OF NOMINAL STEERING OBSERVER DESIGN

One of the scopes of this work is to analyze the performance of the model-based observer used to generate residuals as designed by Melvin (1998). It is necessary to plot the residuals produced by this design actual 21UUV sensor data. The residuals generated using this design will be evaluated over the previously specific intervals. Again, it is important to note the magnitude of residuals produced during the maneuvering specific intervals of this data set. Initial viewing of the residuals generated over these intervals will lead to a better understanding of how the act of maneuvering the vehicle increases the residual output of model-based observer.

Figures 3.15, 3.16, and 3.17 are the residuals produced by the nominal model-based observer for sideslip, yaw rate, and heading, respectively. Each data set is represented on the residual plots and is labeled with respect to its interval of evaluation. The important concept to take from these plots of model-based observer residuals is that the residuals produced by maneuvering of the 21UUV are very large and without periodicity. The propensity of vehicle maneuvering to increase the generation of residuals is very pronounced in each data set shown. Thus, in order to detect a fault in the steering subsystem, proper resolution of the fault through the inherent residuals of the system must occur. The implementation and resolution characteristics of previously designed model-based observer techniques will be the subject of the next section in this work.

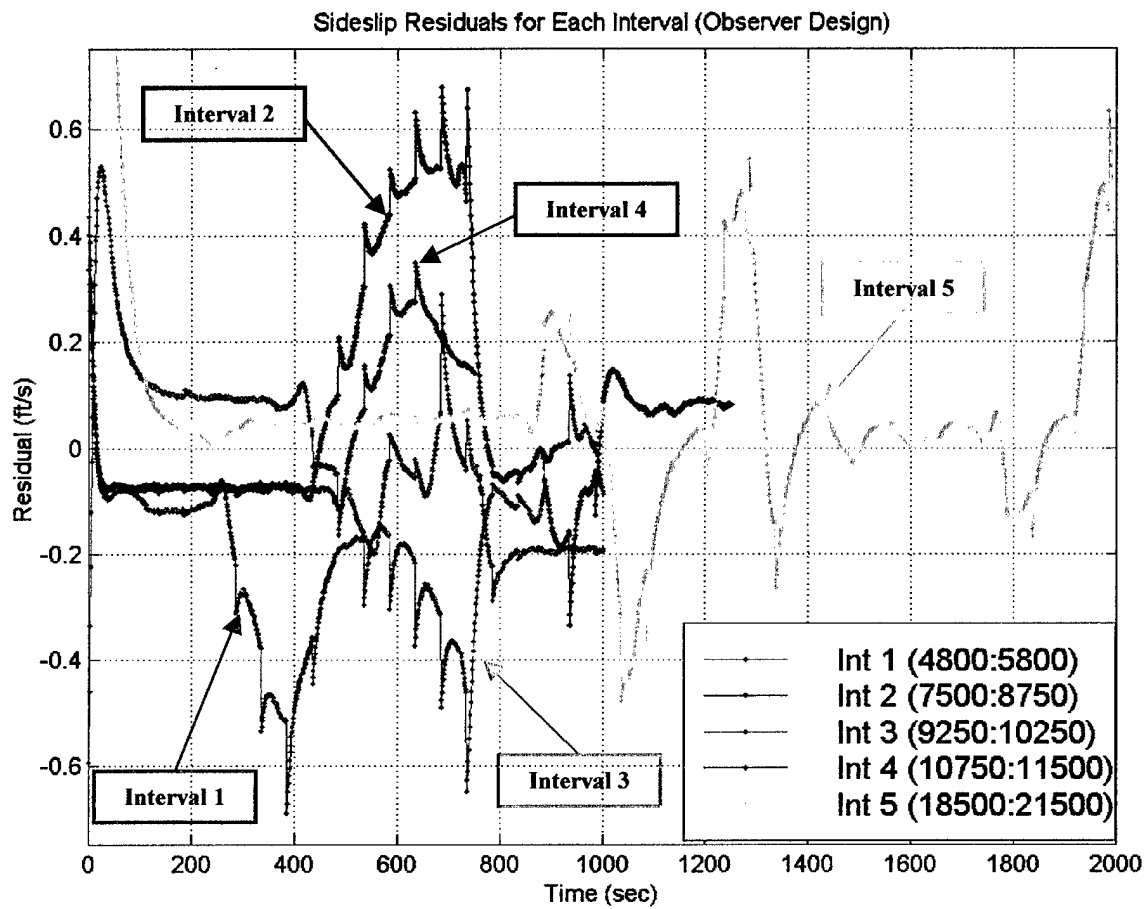


Figure 3.15 Sideslip Velocity Residuals for Observer Design

Note for Figure 3.15: Source Code Name – “O_dl.m”

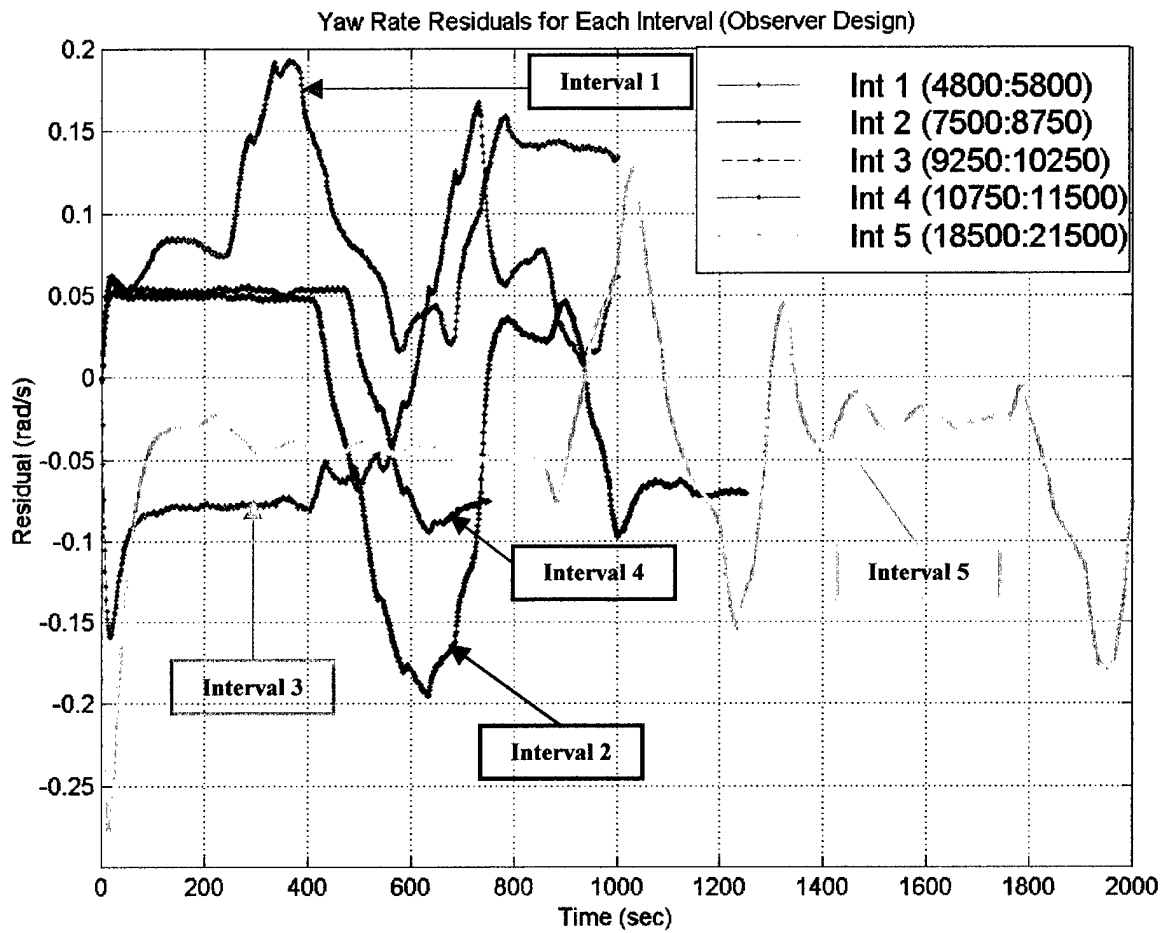


Figure 3.16 Yaw Rate Residuals for Observer Design

Note for Figure 3.16: Source Code Name – “O_dl.m”

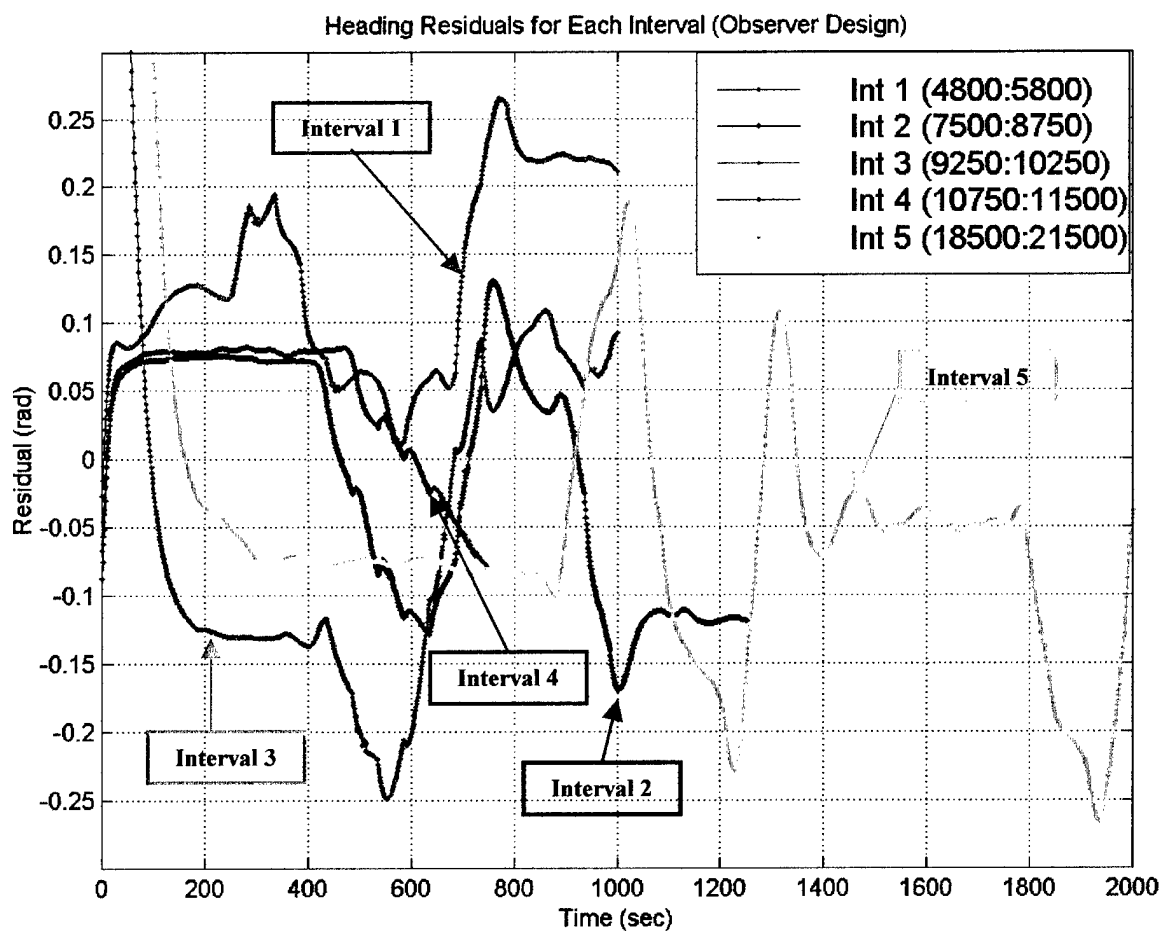


Figure 3.17 Heading Residuals for Observer Design

Note for Figure 3.17: Source Code Name – “O_dl.m”

E. IMPLEMENTATION OF FAULT INTO MODEL-BASED OBSERVER DESIGN

1. Description of Manual Fault Implementation

The introduction of a fault into the model-based observer design was accomplished by adding a 0.4-radian deflection into the rudder command of the steering subsystem. The 0.4 radian deflection command reflects a situation where a fin is stuck in a full stroke position. This additive 0.4-radian input command should produce residuals that clearly indicate a fault. By inhibiting the vehicle to reach proper heading commands, the stuck fin would generate sensor measurements that were not estimated by the controllers. The difference between the estimated state values and the measured state values should produce adequate residual response that can be seen throughout the residuals generated by the observer as shown in Figures 3.15, 3.16, and 3.17.

Revisiting the observer equation for estimation of states, the additive input of the malfunctioning fin appears as the new variable f_s . Written in state-space form, the new observer equation and subsequent residual equation would become:

$$\begin{aligned}\hat{\mathbf{x}} &= (\mathbf{A} - \mathbf{K}\mathbf{C})\hat{\mathbf{x}} + \mathbf{B}(u + f_s) + \mathbf{K}\mathbf{y}; \\ \hat{\mathbf{x}} &= \mathbf{A}_o\hat{\mathbf{x}} + [\mathbf{B} : \mathbf{K}] \begin{bmatrix} (u + f_s) \\ \mathbf{y} \end{bmatrix}; \\ &\text{with....} \\ \mathbf{v} &= (\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}) = -\mathbf{C}_o\hat{\mathbf{x}} + \mathbf{D}_o \begin{bmatrix} (u + f_s) \\ \mathbf{y} \end{bmatrix}.\end{aligned}$$

five plots chosen to be included in this work are indicative of the overall residual generation by the fin fault throughout the entire data set (data: 3000-21900). Each data set is represented in the following plots. There were four different scenarios of fault implementation represented in the model-based observer residual plots. The data set and its respective scenario for fault implementation are as follows:

Data Set: 5200:5800 – Fault occurs during the maneuver and remains constant beyond maneuver completion. (Figure 3.18)

Data Set: 7500:8750 – Fault occurs before the maneuver and remains constant beyond maneuver completion. (Figure 3.19)

Data Set: 9250:10250 – Fault occurs during the maneuver and is corrected prior to maneuver completion. (Figure 3.20)

Data Set: 10750:11500 – Fault occurs before the maneuver and is corrected prior to maneuver completion. (Figure 3.21)

Data Set: 18500:20500 – Fault occurs during the maneuver and is corrected prior to maneuver completion. (Figure 3.22)

The *Matlab* code that generates the following plots is 'O_d1_faults.m' and is contained in Appendix A.

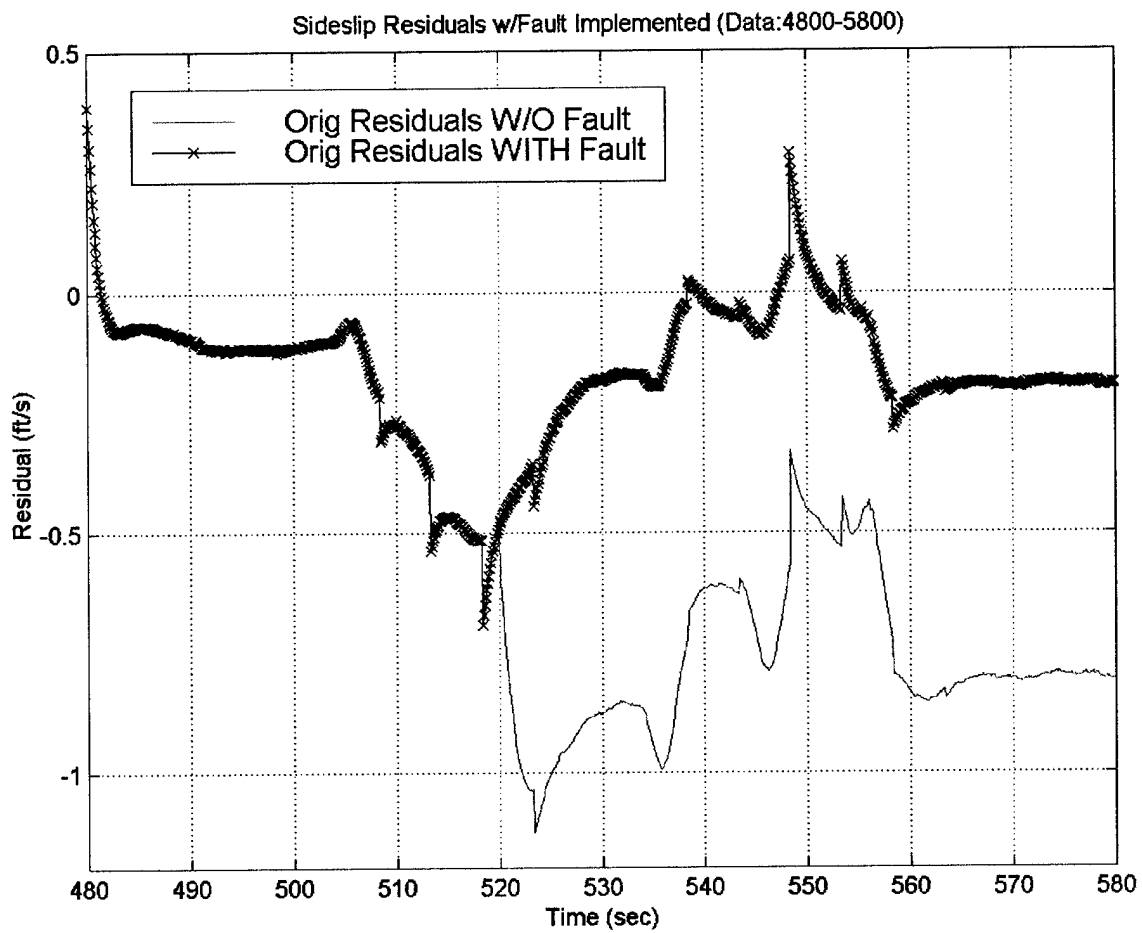


Figure 3.18 Sideslip Fault Detection Using Observer Design (Data: 4800-5800)

Notes for Figure 3.18: Source Code Name – “O_d1_faults.m”
Coefficients Used for Residual Generation: Original

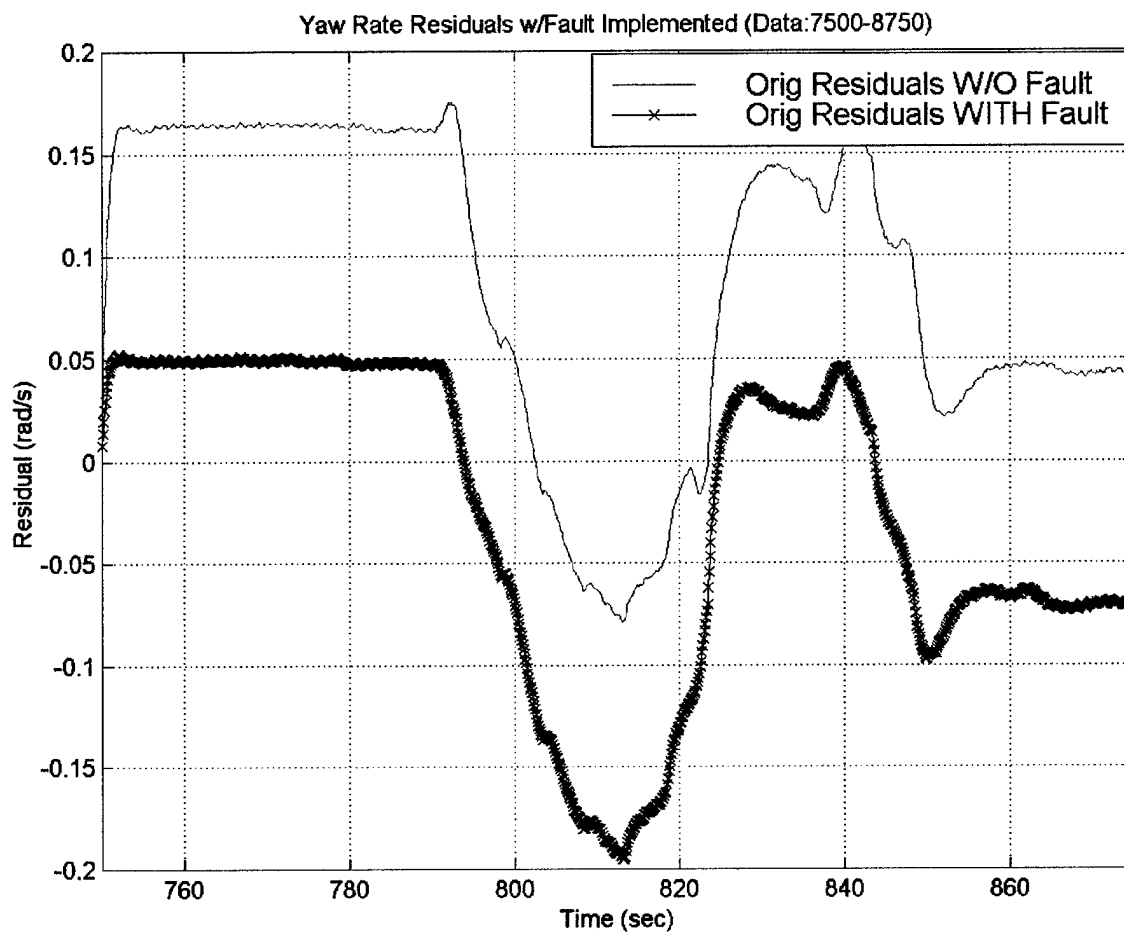


Figure 3.19 Yaw Rate Fault Detection Using Observer Design (Data: 7500-8750)

Notes for Figure 3.19: Source Code Name – “O_d1_faults.m”
Coefficients Used for Residual Generation: Original

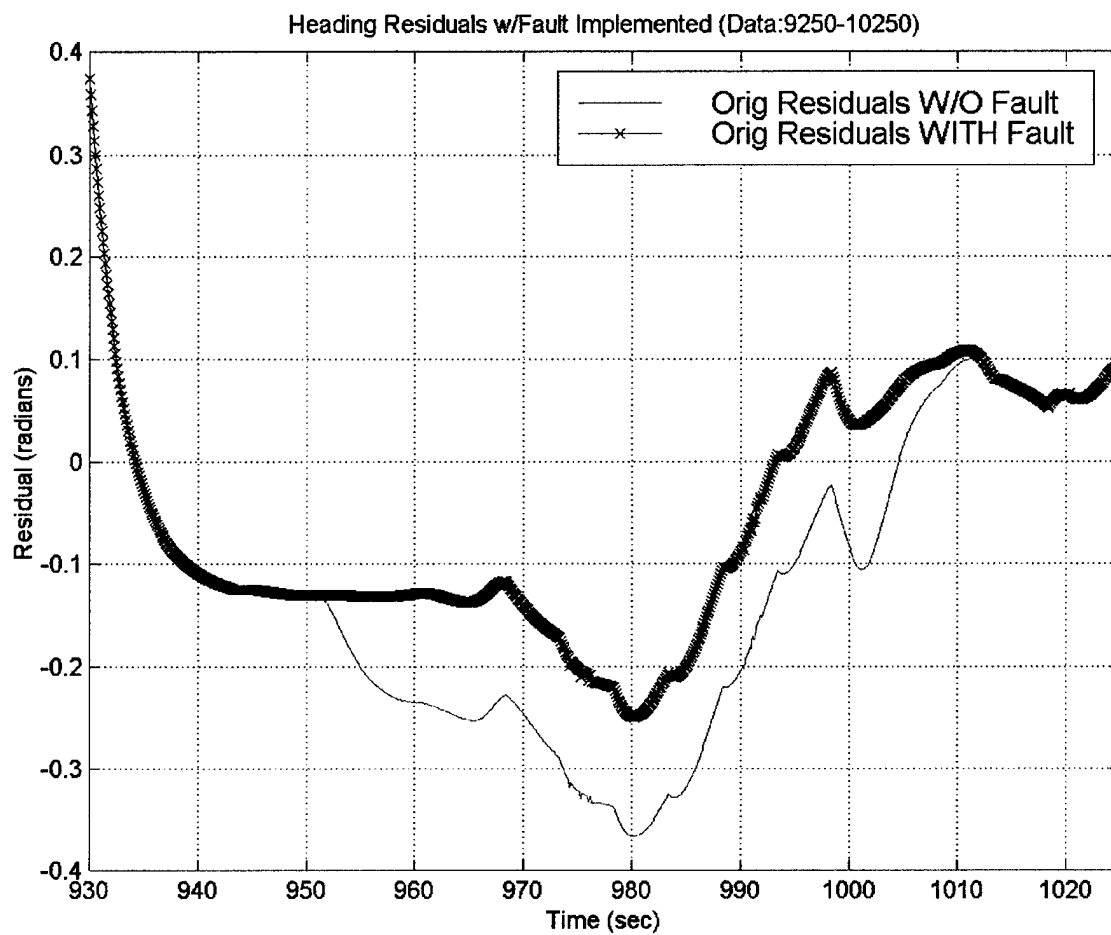


Figure 3.20 Heading Fault Detection Using Observer Design (Data: 9250-10250)

Notes for Figure 3.20: Source Code Name – “O_d1_faults.m”
Coefficients Used for Residual Generation: Original

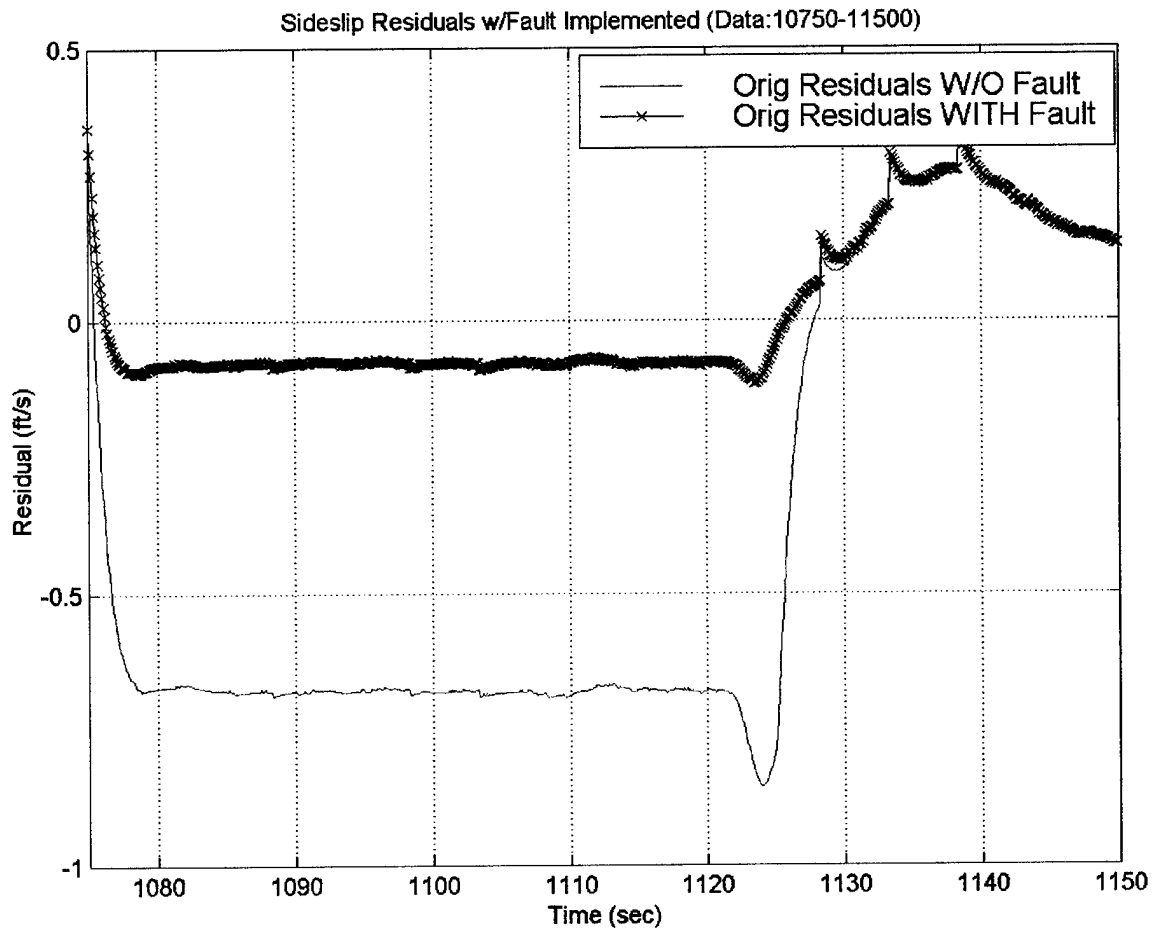


Figure 3.21 Sideslip Fault Detection Using Observer Design (Data: 10750-11500)

Notes for Figure 3.21: Source Code Name – “O_d1_faults.m”
Coefficients Used for Residual Generation: Original

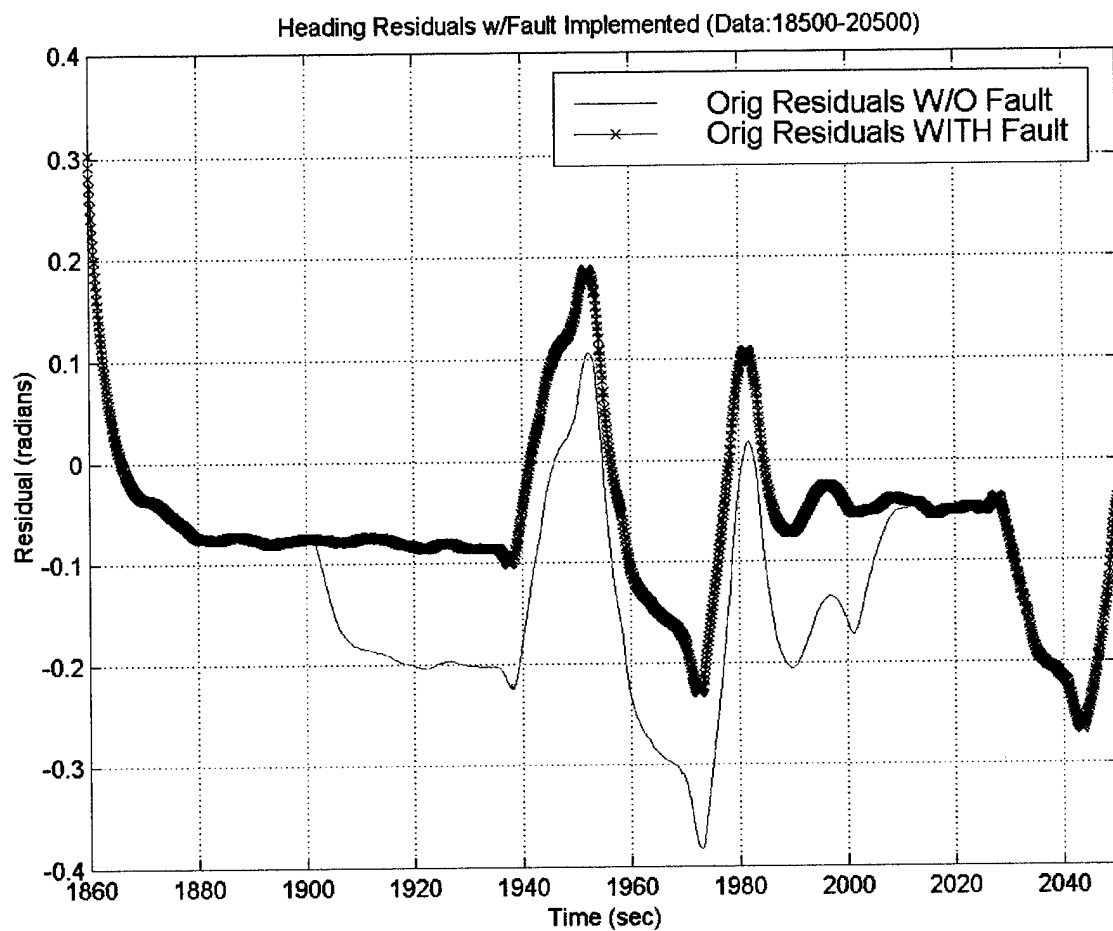


Figure 3.22 Heading Fault Detection Using Observer Design (Data: 18500-20500)

Notes for Figure 3.22: Source Code Name – “O_d1_faults.m”
Coefficients Used for Residual Generation: Original

F. CONCLUSIONS

A model-based observer design attempts to accurately estimate the state variable values of the steering subsystem through predetermined gain matrix calculations. This gain matrix is calculated by placing observer poles at desired locations that provide good observer speed and accuracy. Uncertainty in the model coefficients will cause excessive residual generation in the subsystem because the observer will be unable to adequately predict varying values of the state variables throughout a maneuver-intensive vehicle run. The residuals generated during maneuvers are of greater magnitude and volatility than those generated during steady-state flight. These residuals 'mask' the residuals generated by a full-stroke fin failure. Without the ability to accurately and reliably detect a full-stroke fin fault, the current model-based observer design is inadequate for robust and reliable fault detection.

Inaccuracies may be present in the hydrodynamic coefficients that form the dynamic and control matrixes of the steering subsystem. If such inaccuracies exist, improper modeling of the estimated dynamic and control matrixes will lead to excessive residual generation due to the increase in measurement differentials. Investigation into this possible cause of excessive residual generation is the focus of the next chapter.

THIS PAGE LEFT INTENTIONALLY BLANK

IV. OPTIMIZATION OF VEHICLE MODEL HYDRODYNAMIC COEFFICIENTS

A. ANALYSIS OF DYNAMIC AND CONTROL MATRIX ERROR

The failure of the original nominal model-based observer to properly generate residuals of small magnitude during maneuvering intervals inhibits the detection of subsequent steering subsystem faults. Without suppressing maneuvering generated residuals and without amplifying fault residuals that exceed threshold levels, it was difficult to discern fault residuals from fault-free residuals corresponding to normal vehicle operation. Obviously, a method must be developed to reduce the residuals generated during vehicle maneuvers. It is proposed that a portion of the increase in residual reduction during maneuvers originates from the inexact values of the hydrodynamic coefficients that constitute the dynamic and control matrixes of the steering subsystem model. Without exact values to formulate the dynamic and control matrixes of the steering subsystem, the observer model will fail to accurately estimate the values of state variables as the vehicle maneuvers in other-than-steady-state conditions. The increased difference between the measured and estimated values of the state variables will generate larger residual response and reduce fault detection possibilities.

1. Error Analysis of Steering Subsystem Model

In order to understand the origin of errors resulting from inaccurate hydrodynamic coefficients in the steering subsystem model, it is necessary to revisit the method describing the state observation error, ϵ_x , as shown in Chapter II. The state observation error was previously defined as the difference between the fully measurable state

equation and the model-based predicted state equation. While this fundamental difference still holds true, the first derivation of the state observation error did not include the possibility of modeling inaccuracies within the dynamic and control matrixes, **A** and **B**, respectively. The original state observation error was shown to be:

$$\begin{aligned}\dot{\varepsilon}_x(t) &= (\dot{\mathbf{x}} - \dot{\hat{\mathbf{x}}}) = \{\mathbf{A}\dot{\mathbf{x}}(t) - \hat{\mathbf{A}}\dot{\hat{\mathbf{x}}}(t)\} + (\mathbf{B} - \hat{\mathbf{B}})\mathbf{u}(t) + \mathbf{E}\mathbf{f}_a(t) + \mathbf{F}\mathbf{d}(t) - \mathbf{K}\{\mathbf{v}(t)\} \\ \mathbf{v}(t) &= \mathbf{y}(t) - \mathbf{C}\hat{\mathbf{x}}(t); \\ &\text{gives....} \\ \dot{\varepsilon}_x(t) &= (\mathbf{A} - \mathbf{K}\mathbf{C})\varepsilon_x(t) + \mathbf{E}\mathbf{f}_a(t) + \mathbf{F}\mathbf{d}(t) + \mathbf{K}\mathbf{f}_s(t) \\ &\text{with....}\mathbf{v}(t) = \mathbf{C}\varepsilon_x(t) + \mathbf{f}_s(t);\end{aligned}$$

If differences between the measured and modeled **A** and **B** matrixes were considered, the resulting state observation matrix may be formulated as follows:

$$\begin{aligned}\dot{\varepsilon}_x(t) &= (\dot{\mathbf{x}} - \dot{\hat{\mathbf{x}}}) = \{\mathbf{A}\dot{\mathbf{x}}(t) - \hat{\mathbf{A}}\dot{\hat{\mathbf{x}}}(t)\} + (\mathbf{B} - \hat{\mathbf{B}})\mathbf{u}(t) + \mathbf{E}\mathbf{f}_a(t) + \mathbf{F}\mathbf{d}(t) - \mathbf{K}\{\mathbf{y}(t)\} \\ &\text{where....}\mathbf{y}(t) = \mathbf{v}(t) + \mathbf{C}\hat{\mathbf{x}}(t); \\ &\text{gives....} \\ \dot{\varepsilon}_x(t) &= (\hat{\mathbf{A}} - \mathbf{K}\mathbf{C})\varepsilon_x(t) + (\delta\mathbf{A})\hat{\mathbf{x}}(t) + (\delta\mathbf{B})\mathbf{u} + \mathbf{E}\mathbf{f}_a(t) + \mathbf{F}\mathbf{d}(t) + \mathbf{K}\mathbf{f}_s(t) \\ &\text{with....}\mathbf{v}(t) = \mathbf{C}\varepsilon_x(t) + \mathbf{f}_s(t);\end{aligned}$$

$\delta\mathbf{A}$ and $\delta\mathbf{B}$ are defined as the deviations in the dynamic and control matrixes of the fully measurable state equation and the model-based predicted state equation. The dynamic, or maneuvering error, is defined as the difference in the measured dynamic matrix and the estimated dynamic matrix, given as:

$$\delta\mathbf{A} = (\mathbf{A} - \hat{\mathbf{A}}).$$

The control, or rudder error, is defined as the difference in the measured control matrix and the estimated control matrix, given as:

$$\delta \mathbf{B} = (\mathbf{B} - \hat{\mathbf{B}}).$$

It is proposed that optimization of the hydrodynamic coefficients that define the $\mathbf{A}$ and $\mathbf{B}$ matrixes will minimize the effects $\delta \mathbf{A}$ and $\delta \mathbf{B}$ have on the overall state observation error and will subsequently lower residual generation during maneuvers. By optimizing certain hydrodynamic coefficients over a given data interval, it is proposed that resultant coefficient values will more accurately represent the hydrodynamic characteristics of the vehicle while it maneuvers. A more accurate representation of the hydrodynamic characteristics of the 21UUV will greatly reduce residual generation.

2. Choice of Hydrodynamic Coefficients for Optimization

The choice of which hydrodynamic coefficients to optimize comes from the study of the closed-loop, state-space representation of the steering subsystem. This representation was shown to be:

$$\mathbf{M} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u.$$

The matrixes above can be defined using hydrodynamic coefficients as:

$$\mathbf{M} = \begin{bmatrix} m - Y_{\dot{v}_r} & -Y_r & 0 \\ -N_{\dot{v}} & I_{zz} - N_{\dot{r}} & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{A} = \begin{bmatrix} Y_{v_r} & Y_r - mU_o & 0 \\ N_{v_r} & N_r & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} Y_{\delta} \\ N_{\delta} \\ 0 \end{bmatrix}.$$

Simplifying the state-space equation by multiplying through by the inverse of $\mathbf{M}$ gives:

$$\dot{\mathbf{x}} = \underset{\mathbf{A}}{[\mathbf{M}^{-1}\mathbf{A}]} \mathbf{x} + \underset{\mathbf{B}}{[\mathbf{M}^{-1}\mathbf{B}]} \mathbf{u}.$$

Matrix $\mathbf{A}$ and matrix $\mathbf{B}$ are now re-defined as:

$$\mathbf{A} = \underset{(3 \times 3)}{[\mathbf{M}^{-1}\mathbf{A}]},$$

$$\mathbf{B} = \underset{(3 \times 1)}{[\mathbf{M}^{-1}\mathbf{B}]},$$

with....

$$\delta \mathbf{A} = \underset{(3 \times 3)}{\delta [\mathbf{M}^{-1}\mathbf{A}]},$$

$$\delta \mathbf{B} = \underset{(3 \times 1)}{\delta [\mathbf{M}^{-1}\mathbf{B}]}.$$

It is recognized that the coefficients that constitute the mass matrix $\mathbf{M}$, are assumed to be relatively sound and do not add to the production of any significant error in the steering model. Of the coefficients that make up the control matrix, both mass, m , and forward velocity, V , do not need to be optimized, since both values are taken to be accurate. The remaining coefficients, which are suitable candidates for optimization, are Y_v , Y_r , N_v , N_r , Y_δ , and N_δ . Y_v and Y_r are coefficients of hydrodynamic sway force induced by sideslip and yaw, respectively. N_v and N_r are coefficients of hydrodynamic yaw moment induced by sideslip and yaw, respectively. Y_δ is a coefficient of linearized sway force produced by the rudder. N_δ is a coefficient of yaw moment produced by the rudder.

The values assigned to these hydrodynamic coefficients are non-dimensionalized when used in the steering model. These coefficients are optimized about the following given initial values:

$$\begin{aligned} Y_v &= -7.406e-03, \\ Y_r &= 2.655e-03, \\ N_v &= -6.746e-03, \\ N_r &= -1.477e-03, \\ Y_\delta &= \alpha * (4.216e-03), \\ N_\delta &= \alpha * (-2.176e-03). \end{aligned}$$

Y_δ and N_δ are both scaled by a value ' α '. This coupling by a scalar value for these two coefficients reduces the overall number of optimized coefficients from six to five.

B. OPTIMIZATION OF HYDRODYNAMIC COEFFICIENTS

In order to proceed further with the optimization of the chosen hydrodynamic coefficients, the tool for which optimization will be accomplished shall be addressed here.

1. *Matlab's Sequential Quadratic Programming Method*

Due to the number of chosen hydrodynamic coefficients from the dynamic and control matrixes, *Matlab's* 'constr' function was utilized for optimization purposes. This function is used to find the constrained minimum of a scalar function of several variables starting with an initial estimate. *Matlab's Optimization Toolbox* (Branch and Grace, 1996) contains all the information necessary to explain the methodology behind the function's algorithm. The basics of the algorithm behind 'constr' are taken from the *Optimization Toolbox* and described here for background on the process by which the optimized coefficients can be found.

'Constr' uses a Sequential Quadratic Programming (SQP) method for optimization. The SQP implementation includes three main steps to find the values for constrained nonlinear optimization. The three main steps are:

- Solving a quadratic programming subproblem
- Line search and merit function calculation
- Updating the Hessian matrix to provide an improved quadratic approximation

a. Quadratic Programming Subproblem

The first step in the SQP is to determine a desirable search direction. At each major iteration of the SQP method, a quadratic subproblem is solved such that a quadratic approximation to the augmented objective function is given by (Branch and Grace, 1996):

$$\begin{aligned} \min (d \in \mathbb{R}^n) \\ q(d) &= \frac{1}{2} d^T H d + c^T d \\ A_i d &= b_i \longrightarrow i = 1, \dots, m_e \\ A_i d &\leq b_i \longrightarrow i = m_e + 1, \dots, m \end{aligned}$$

The design variables are the components of d , and the Hessian matrix is given as H .

Here, A_i refers to the i^{th} row of the m -by- n matrix A .

The procedure for obtaining the solution for the next search direction consists of two phases. The first phase consists of the calculation of the next feasible point along the given search direction. The second phase then involves the generation of an iterative sequence of feasible points that converge to the solution. Estimates of the active constraints that are on the constraint boundaries at the solution point are contained in the active set, $\bar{A}_k$. The subscript k is the value of the number of performed iterations. At each iteration, $\bar{A}_k$ is updated and used for the basis of the next new search direction, $\hat{d}_k$. The variable, $\hat{d}_k$, is used here as a quadratic subproblem search direction variable. It does not represent the search direction, d_k , which is related to the search direction of the major iterations of the SQP method. The new quadratic subproblem search direction, $\hat{d}_k$,

is calculated and minimizes the objective function. The search direction remains on the active constraint boundaries. After calculation of the new search direction, a step is taken of the form:

$$x_{k+1} = x_k + \eta \hat{d}_k.$$

There are only two choices at each iteration for the step length, η , due to the quadratic nature of the objective function. If $\eta=1$, an exact step is taken to the minimum of the objective function that is restricted by the null space $\bar{A}_k$. When an exact step as this is taken, then this is the solution to the quadratic subproblem. If a step of unity cannot be taken, the step along $\hat{d}_k$ is less than unity and is to the nearest constraint. A new constraint is included in the active set for the next iteration.

Lagrange multipliers, λ_k , are calculated when n independent constraints are included in the active set, without locating a minimum. The Lagrange multipliers are calculated so that they satisfy the nonsingular set of linear equations

$$\bar{A}_k^T \lambda_k = c.$$

If all elements of λ_k are positive, x_k is the optimal solution to the quadratic subproblem. If any component of the Lagrange multipliers is negative, and does not correspond to an equality constraint, then the corresponding element is deleted from the active set and a new iterate is sought (Branch and Grace, 1996).

b. Line Search and Merit Function Calculation

After determining the new search direction d_k , the design is updated using a one-dimensional search problem that is used to form the new iterate:

$$x_{k+1} = x_k + \eta_k d_k.$$

The step length parameter, η_k , for the search direction is calculated to sufficiently decrease the value of the merit function. The merit function used in this implementation is given as (Branch and Grace, 1996):

$$\Psi(x) = f(x) + \sum_{i=1}^{m_c} r_i \cdot g_i(x) + \sum_{i=m_c+1}^m r_i \cdot \max\{0, g_i(x)\}.$$

The recommended setting for the penalty parameter is (Powell, 1983):

$$r_i = (r_{k+1})_i = \max_i \left\{ \lambda_i, \frac{1}{2}((r_k)_i + \lambda_i) \right\} \\ i = 1, \dots, m$$

The initial penalty parameter in this implementation is set to:

$$r_i = \frac{\|\nabla f(x)\|}{\|\nabla g_i(x)\|},$$

where $\|\cdot\|$ is the Euclidean norm.

This ensures larger contributions to the penalty parameter from constraints with smaller gradients, which would be the case for active constraints at the solution point (Branch and Grace, 1996).

c. *Updating the Hessian Matrix*

The search direction has been determined and the one-dimensional search to update the design has been performed, at this point. Now, it is necessary to update the Hessian matrix of the Lagrangian function, H , in order to provide an improved quadratic approximation to the augmented objective design. Powell (1977) recommends the Broydon-Fletcher-Shanno-Goldfarb (BFGS) method where λ_i ($i=1, \dots, m$) is an estimate of the Lagrange multipliers. The Hessian update (BFGS) is given as:

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T H_k}{s_k^T H_k s_k}$$

where

$$s_k = x_{k+1} - x_k$$

$$q_k = \nabla f(x_{k+1}) + \sum_{i=1}^n \lambda_i \cdot \nabla g_i(x_{k+1}) - \left(\nabla f(x_k) + \sum_{i=1}^n \lambda_i \cdot \nabla g_i(x_k) \right)$$

It is recommended by Powell (1977) to keep the Hessian positive definite even though it may be positive indefinite at the solution point. Branch and Grace (1996) propose that a positive definite Hessian is maintained providing $q_k^T s_k$ is positive at each update and that H is initialized with a positive definite matrix. They proceed to state that when $q_k^T s_k$ is not positive, q_k is modified on an element-by-element basis so that $q_k^T s_k > 0$.

Vanderplaats (1999) proposes a flowchart outlining the algorithm for this method as shown in Figure 4.1. The flowchart has been modified to accurately reflect the specific methodology and terminology described in *Matlab's Optimization Toolbox*.

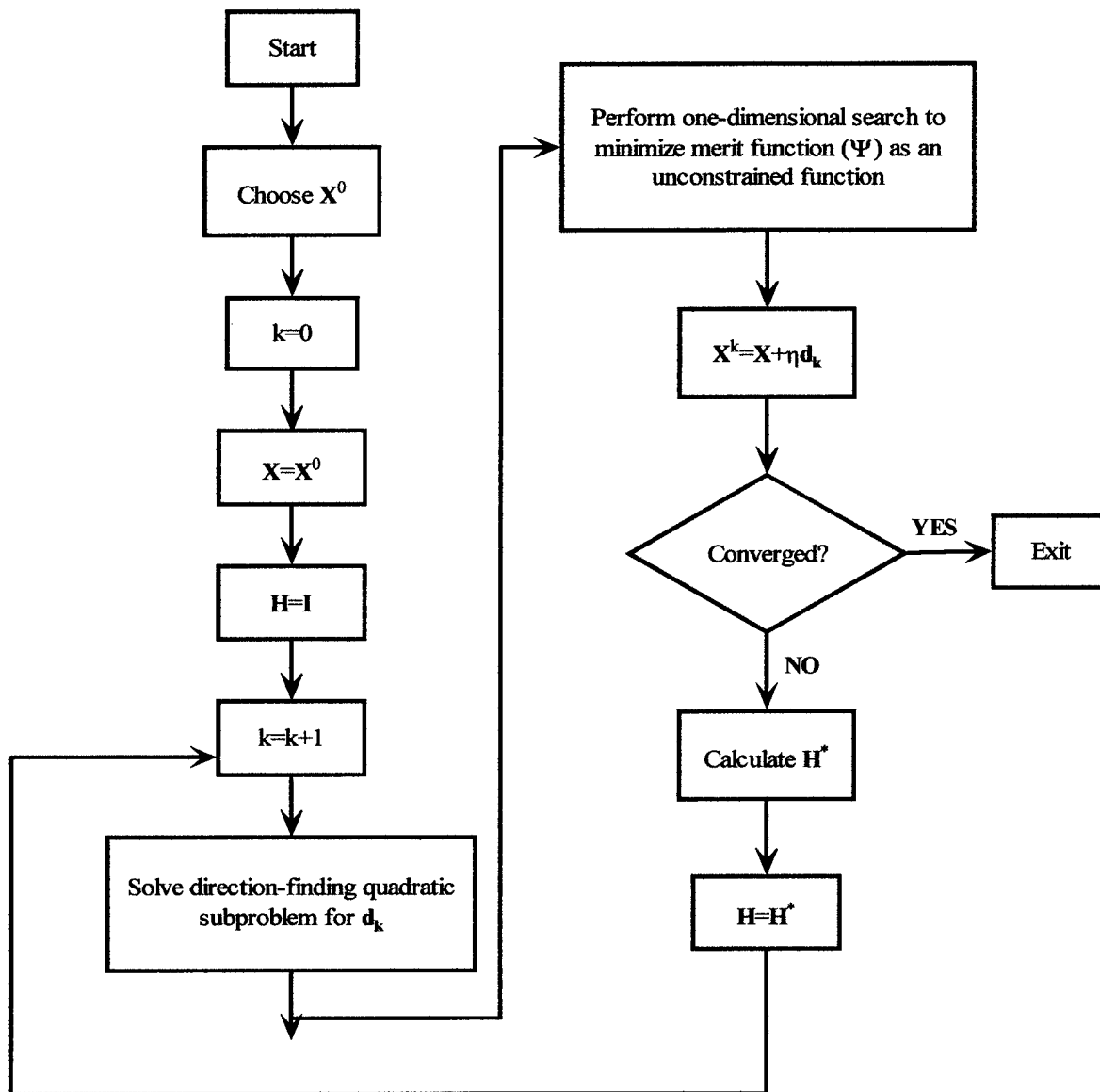


Figure 4.1 Algorithm for *Matlab's* Sequential Quadratic Programming (SQP)

2. Optimization Arguments for Use in 'CONSTR'

The inputs into the arguments of the function 'constr' are provided here in order to describe the optimization parameters used in finding the optimal values for the chosen hydrodynamic coefficients. By giving values for these arguments, a good description can be given for the limitations and options that were used to decide the values for the resulting optimum coefficients. A description of all pertinent function arguments is also listed here in order to describe all governing optimization parameters.

a. *Function String*

$x = \text{constr}('final')$ The function '*final*' is the function string that contains the name of the function that computes the objective function to be minimized at the point x . The function '*final*' returns the scalar valued function to be minimized. A listing the *Matlab* function '*final.m*' is given in Appendix A.

b. $x0$

$x = \text{constr}('final', x0)$ The $x0$ vector contains the starting values for scalar multiplication of the hydrodynamic coefficients for optimization. The starting vector for this optimization scheme is the 5x1 unity vector, [1 1 1 1 1]. This vector is multiplied by the original starting values of the hydrodynamic coefficients that were previously listed above. Having the starting values equal to unity ensures that the optimization algorithm begins with the values that were taken from the original model-based observer design. Obviously, deviation from these initial values demonstrates improved reduction in

residual generation of the model. It was also found that the use of any other starting vector did not result in improved optimization results.

c. Options

$x = \text{constr}('final', x0, \text{options})$ The *options* vector controls the parameters of the optimization algorithm. The options used in this optimization were options (1), (2), and (3). Option (1) produces a tabular display of intermediate results that include the function value, the number of function calls, and the status of the Hessian matrix. Option (2) controls the accuracy of the solution at x . Option (3) controls the accuracy of the objective function at the solution of f . Both options (2) and (3) were set at an accuracy level of 0.01. Any further increase in accuracy did not provide results of greater significance, but only prolonged the lengthy time needed to run the optimization.

d. VLB and VUB

$x = \text{constr}('final', x0, \text{options}, \text{vlb}, \text{vub})$ The vectors *vlb* and *vub* control the lower and upper bounds of the variation of the hydrodynamic coefficients. These vectors contained bounding scalar multipliers of the original coefficient values. As was the case for $x0$, these vectors were multiplied by the original coefficient values in order to obtain variation in the hydrodynamic coefficients. The values for the coefficients were allowed to be optimized over a range to $\pm 10X$ their original values.

As an example from the *Matlab* code 'final.m', the lowest value for the coefficient of hydrodynamic sway force induced by sideslip, Y_v , is obtained by multiplying the original coefficient value by the first value of the vector *vlb*:

$$Y_{v1} = \text{vlb}(1) * (-7.406e-03).$$

With an allowable +/- 10X variation from the original coefficient value, the range of optimization for Y_v is from -0.07406 to 0.07406. Limitations on the physical feasibility of this range of values is not of significant concern, as the final values found through optimization often did not exceed 4X the value of the original hydrodynamic coefficient.

e. SS and ES

$x = \text{constr}('final', x0, options, \text{vlb}, \text{vub}, [], ss, es)$ The scalar values *es* and *ss* are arguments passed to the optimization function 'final' and contain the respective starting and ending values of the data set interval over which optimization will be calculated. The values of *es* and *ss* were changed according to the maneuvering specific data interval over which it was desired to find optimized hydrodynamic coefficients.

3. Scalar Reduction by Use of a Weighting Matrix

Since the actual residual error of the model-based observer is a (3x1) vector of residuals, it is necessary to scale the residual error vector in order to provide a scalar result of the objective function for use by the optimization function 'constr'. The residual error vector consists of the state observation errors for sideslip (v), yaw rate (r), and

heading (ψ). The residual error vector and the units pertaining to each residual state are defined as:

$$\mathbf{Residuals} = \underset{(3 \times 1)}{[\mathbf{v}]} = \begin{bmatrix} v(ft/s) \\ rr(rad/s) \\ \psi(rad) \end{bmatrix}.$$

A weighting matrix, $\mathbf{P}$, was multiplied by the residual error vector in a manner to return a (1x1) scalar value that accurately weighted the combination of all three state observation residuals. In order to accomplish this task, the following equation was implemented into the optimization code to provide an updated objective function for optimization:

$$\underset{(1 \times 1)}{funmin} = \underset{(1 \times 3)}{[\mathbf{v}']} * \underset{(3 \times 3)}{\begin{bmatrix} P_1 & 0 & 0 \\ 0 & P_2 & 0 \\ 0 & 0 & P_3 \end{bmatrix}} * \underset{(3 \times 1)}{[\mathbf{v}]}.$$

The appropriate weighting values for scaling the residual error vector were chosen by attempting to non-dimensionalize the units pertaining to each residual. Analysis of the updated objective function for optimization gave insight into the values to choose for each weighting matrix coefficients.

The updated objective function can be written as follows:

$$funmin = v^2 * P_1 + rr^2 * P_2 + \psi^2 * P_3$$

where

$$P_1 \approx \left[(ft/s)^2 \right]^{-1}$$

$$P_2 \approx \left[(rad/s)^2 \right]^{-1}$$

$$P_3 \approx \left[(rad)^2 \right]^{-1}$$

Using the vehicle length of 20 feet as a magnitude parameter for the sideslip velocity, and noting that conversion between ft/s and rad/s cannot be accomplished numerically, values for the weighting matrix were chosen with the assistance of multiple executions of the optimization code. The resultant weighting matrix used for optimization of the hydrodynamic coefficients is given as:

$$P = \begin{bmatrix} .01 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

(3x3)

4. Use of a Performance Index to Increase Fault Detection

In order to adequately detect a fault in the steering subsystem, it was shown that the residuals generated during vehicle maneuvering had to be reduced to allow the residuals generated by the actual fault to be 'observed'. An inherent problem in residual reduction comes from the possibility of nullifying the maneuvering generated residuals to a degree where even residuals generated from faults would be below detection thresholds.

This case of residual reduction 'overkill' was corrected by the implementation of a 'Performance Index' into the hydrodynamic coefficient optimization code. The chosen performance index amplified the residuals generated from a fault condition while reducing the fault-free residuals generated from vehicle maneuvering. In theory, the use of the performance index would allow for adequate fault detection during vehicle maneuvers. The performance index is defined as:

$$f = error_nofault + \frac{1}{error_fault}.$$

As can be seen from the algebra of the performance index, the minimization of the no-fault residuals reduces maneuvering residuals. Conversely, the subsequent minimization of the objective function results in an increase in the value of the fault residuals. The trade-off between increasing the fault generated residuals and decreasing the fault-free generated residuals will theoretically provide the model-based observer design with the means to detect a fault.

The fault generated residuals were the result of a full-stroke fin failure as previously described in Chapter III. The fault-free generated residuals were resultant from the previously described maneuvering specific data intervals of Chapter III.

5. Root Mean Squared Error Analysis of Generated Residuals

In order to quantitatively compare the residual generation of the original model-based observer design with the residual generation of the optimized design, the root mean squared value of the scalar performance index over a given data interval was compared

between the two designs. The root mean squared value is defined as the squared summation of the scalar performance index divided by the number of data points within the evaluated data interval. This can be shown numerically as:

$$RMS = \frac{\sum_{i=t_{ss}}^{i=t_{es}} f^2}{(es - ss)}.$$

Since both optimized and original designs will incorporate residual scaling by use of the weighting matrix, P , the utilization of the root mean squared method will allow equal comparisons of the two designs with respect to magnitude of residual generation. Also, the use of the scaled performance index value alleviates the need for individual comparison of each of the three state residuals.

C. RESULTS OF HYDRODYNAMIC COEFFICIENT OPTIMIZATION

1. Tabular Results of Hydrodynamic Coefficient Optimization

Innumerable optimization runs were conducted over all data set intervals in order to obtain the best optimized hydrodynamic coefficients for the model-based observer design. Only the final results will be listed here. If further manipulation of any of the defining parameters of optimization is desired, the *Matlab* code written for these calculations and their subsequent optimized results is contained in Appendix A.

The final optimized hydrodynamic coefficients and their respective performance improvements to residual generation are first listed in tabular form for numerical

comparison purposes. Graphical representation of the optimized coefficients performance will be included later in this chapter.

a. Optimized Hydrodynamic Values for Each Data Set Interval

Optimization of the hydrodynamic coefficients over each data set interval was conducted using the 21UUV sensor data. Each data set interval had its own distinct set of coefficient values that were found to optimally minimize the scaled performance value, f , pertaining to that data interval's residual generation characteristics. As previously mentioned, the scalar multipliers for each coefficient were allowed to be optimized over a range from -10 to $+10$. The initial starting values for each coefficient were the values of the coefficients from the original model-based observer design. Table 4.1 lists the final optimized 'scalar multipliers' of each hydrodynamic coefficient for each data set interval.

Data Set	Y_v	Y_r	N_v	N_r	α (Y_δ & N_δ)
4800-5800	-0.0339	3.9459	-0.2479	-1.3745	1.5478
7500-8750	1.3507	0.5918	-0.7773	-3.1481	3.0880
9250-10250	-0.0460	3.9509	-0.4077	-1.3938	0.5505
10750-11500	0.2156	2.5048	0.0247	-0.6599	2.3958
18500-20500	3.9994	3.8151	-10.000	-1.4565	0.2213

Table 4.1 Optimized Scalar Multipliers for Each Hydrodynamic Coefficient

Notes for Table 4.1: Source Code Name – "side_perf_sets.m"
Function File Name – "final_perf.m"
 $x0=[1 \ 1 \ 1 \ 1 \ 1]$, $vlb=[-10 \ -10 \ -10 \ -10 \ -10]$, $vub=[10 \ 10 \ 10 \ 10 \ 10]$

b. Performance Characteristics of Optimized Coefficients

The resulting performance of each optimized coefficient set is given in Table 4.2. The performance values for the coefficients from the original model-based observer design are included in order to gauge the improvement in model performance by optimization of the hydrodynamic coefficients. Also included in Table 4.2 is the number of function calls necessary to reach optimal values for the coefficients. This information would become of importance in future work incorporating on-board, adaptive optimization as a means to improve vehicle performance. The larger the value of the function call, the longer the CPU time necessary to calculate residuals. The Root Mean Squared (RMS) data that is included in this table was calculated over the entire data set from data =4000:21900. This calculation gives insight into the actual performance of each hydrodynamic coefficient set as if it were to be used continuously over all maneuvering situations. This valuable insight will allow direct comparisons between the different coefficient sets for best performance in residual generation during maneuvering.

Data Set	Original <i>f</i> -value	Optimized <i>f</i> -value	Orig.RMS (All data)	Opt. RMS (All data)	% Δ RMS	# Fun. Calls
4800- 5800	557.088	2.531	0.0193	0.0057	70.47% (↓)	209
7500- 8750	139.204	25.010	0.0193	0.0119	38.34% (↓)	125
9250- 10250	13998.5	2047.6	0.0193	0.0056	70.98% (↓)	119
10750- 11500	13.642	0.5197	0.0193	0.0096	50.25% (↓)	284
18500- 20500	1656420	204593	0.0193	0.0354	83.41% (↑)	168

Table 4.2 Performance Characteristics of Optimized Coefficients

Notes for Table 4.2: Source Code Name – “side_perf_sets.m” and “RMS_obs.m”
Data Interval for RMS Calculations – 10000 to 21900

2. Graphical Results of Hydrodynamic Coefficient Optimization

Although numerical comparisons between the performance characteristics of each optimized coefficient set is significant, actual graphical analysis of the residual reduction of the optimized coefficient sets is necessary in order to fully grasp the benefits of attempting to decrease the difference in the measured and estimated dynamic and control matrixes of the model-based observer design. By evaluating the numerical performance characteristics of the hydrodynamic coefficients, the initial starting point for analyzing the graphical nature of residual reduction by optimization can be found. Coefficient data set THREE (Data: 9250-10250) produced the best results for residual reduction over the entire data interval. The percentage decrease in RMS residual value over the entire data set was slightly greater than the decrease resulting from coefficient data set ONE. Although coefficient data set THREE did not result in the largest percentage decrease in scalar performance value, f , over its respective data set, it did produce the greatest decrease in residual generation over the entire data run when compared to the other four coefficient sets. With this in mind, Figures 4.2 through 4.6 are included here in order to give a representation of the actual residual reduction that occurs when optimization of the hydrodynamic coefficients takes place. Only one characteristic state residual will be shown for each of the five coefficient sets. Figures 4.7 and 4.8 are included to provide graphical analysis of residual generation over a large data interval (Data: 10000-20000) using the both coefficient data sets THREE and ONE, due to their superior residual reduction performance.

Residual reduction over maneuvering specific data intervals can be accomplished by optimizing the hydrodynamic coefficients that constitute the dynamic and control

matrixes of the model-based observer design. By optimizing the hydrodynamic coefficients of these matrixes, a better model is formed that more accurately reflects the values of the measurement dynamic and control matrixes. The estimated control and input matrixes can utilize the newly optimized hydrodynamic coefficients in order to reduce the difference between the measured and estimated state values that produce residuals.

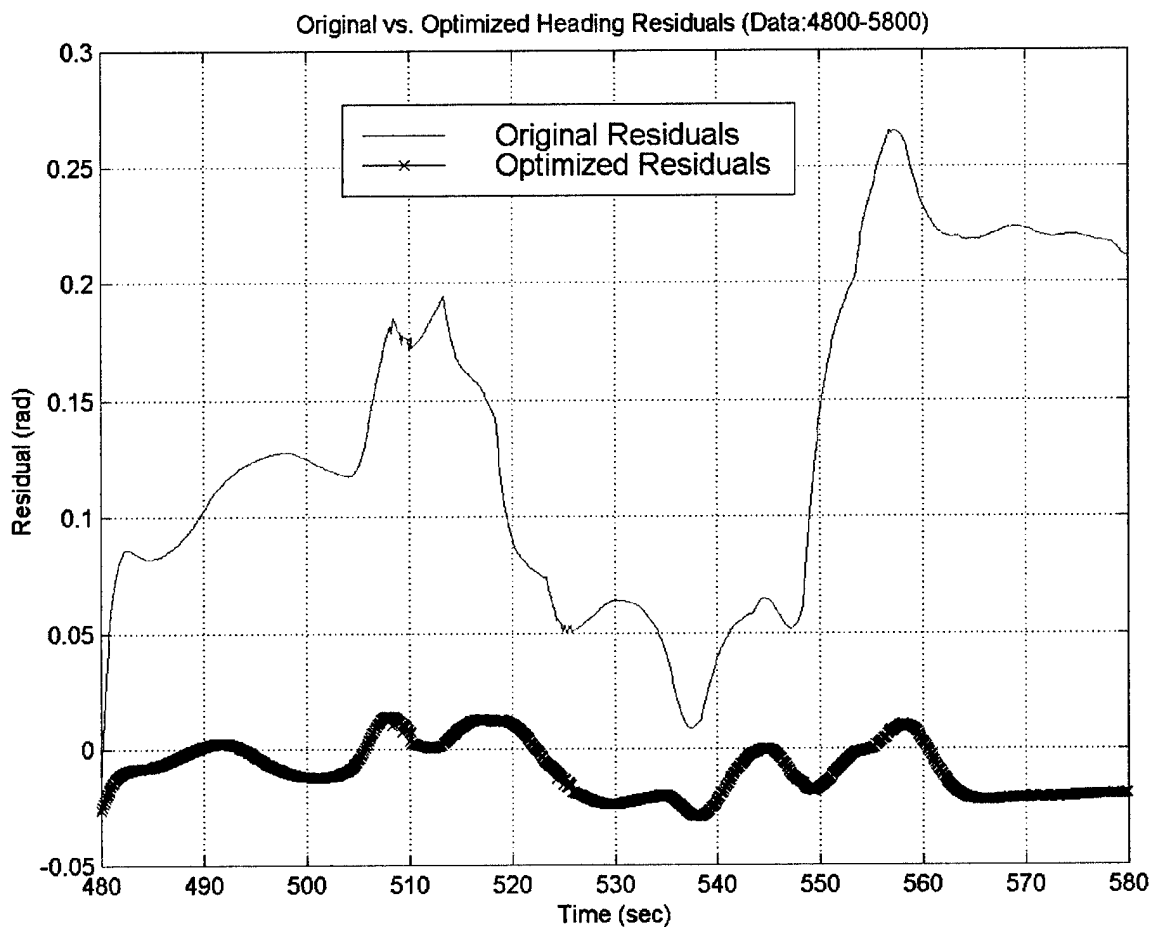


Figure 4.2 Residual Reduction by Hydrodynamic Coefficient Optimization
(Data Interval: 4800-5800)

Notes for Figure 4.2: Source Code Name – “opt_res_reduc.m”
Scalar Coeffs. Used – [-0.0339 3.946 -0.248 -1.375 1.548]

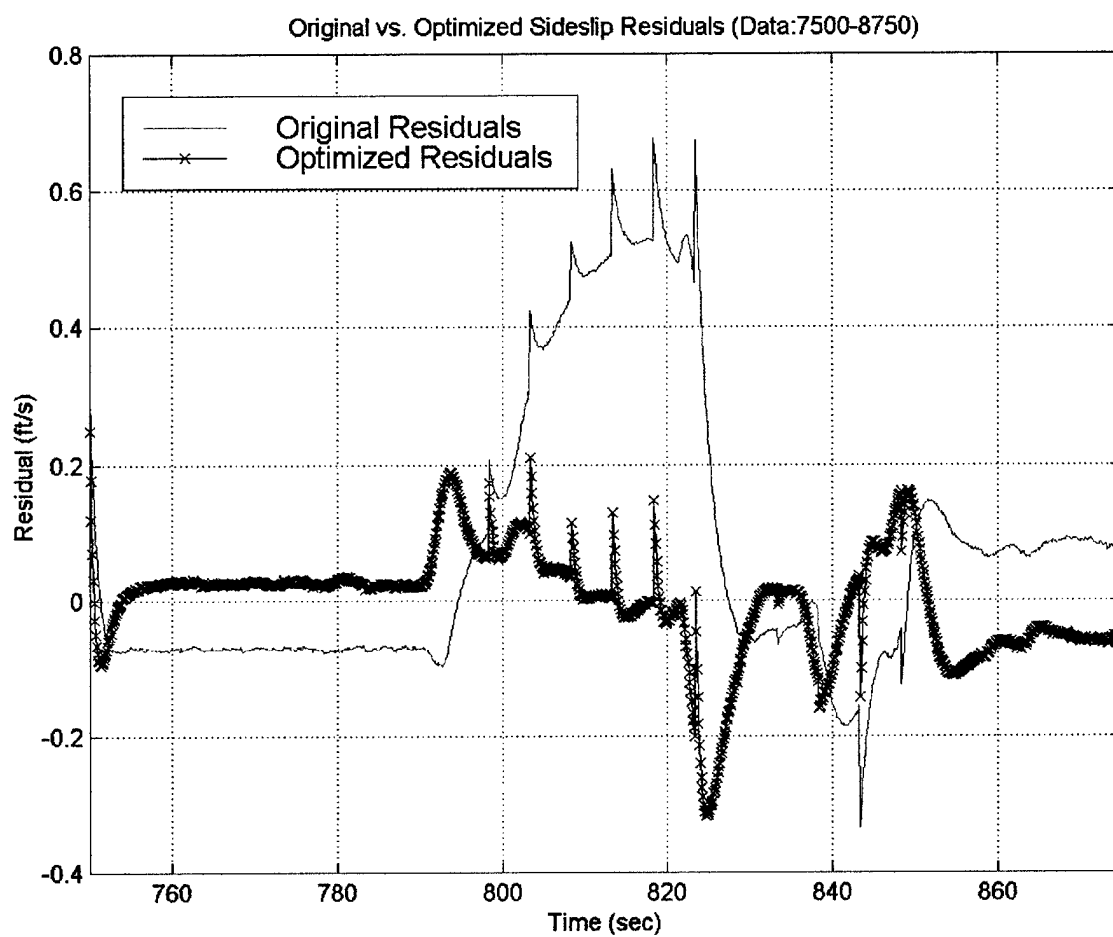


Figure 4.3 Residual Reduction by Hydrodynamic Coefficient Optimization
(Data Interval: 7500-8750)

Notes for Figure 4.3: Source Code Name – “opt_res_reduc.m”
Scalar Coeffs. Used – [1.351 0.5918 -0.777 -3.148 3.088]

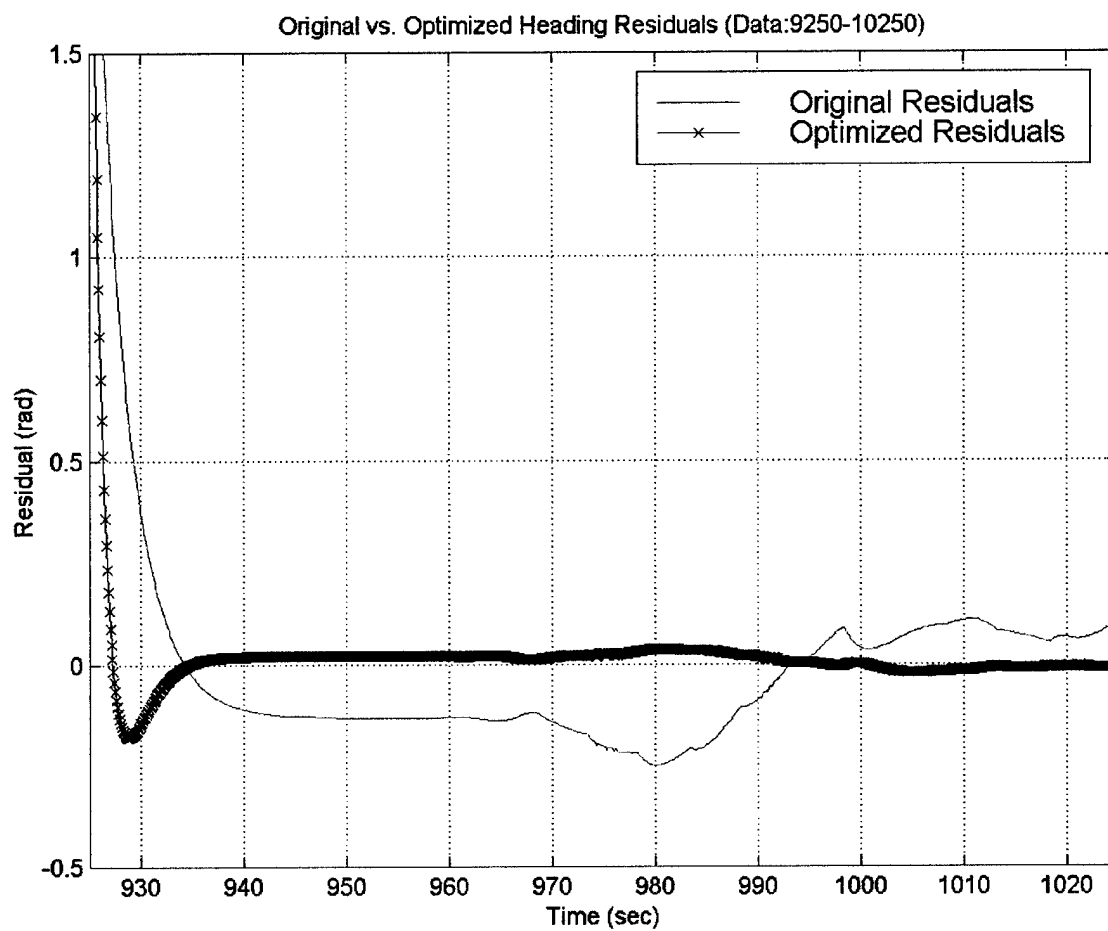


Figure 4.4 Residual Reduction by Hydrodynamic Coefficient Optimization
(Data Interval: 9250-10250)

Notes for Figure 4.4: Source Code Name – “opt_res_reduc.m”
Scalar Coeffs. Used – [-0.046 3.951 -0.408 -1.394 0.551]

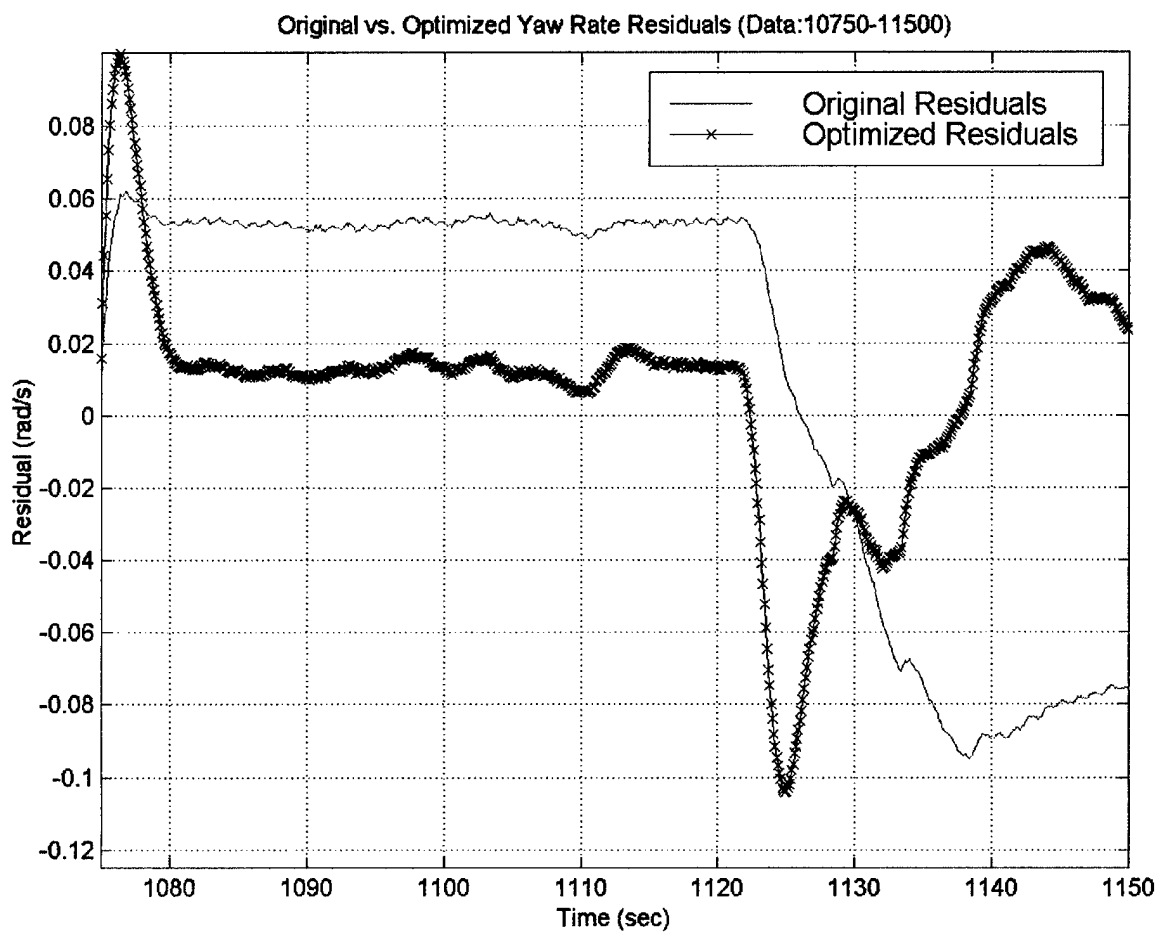


Figure 4.5 Residual Reduction by Hydrodynamic Coefficient Optimization
(Data Interval: 10750-11500)

Notes for Figure 4.5: Source Code Name – “opt_res_reduc.m”
Scalar Coeffs. Used – [0.216 2.505 0.0247 -0.660 2.396]

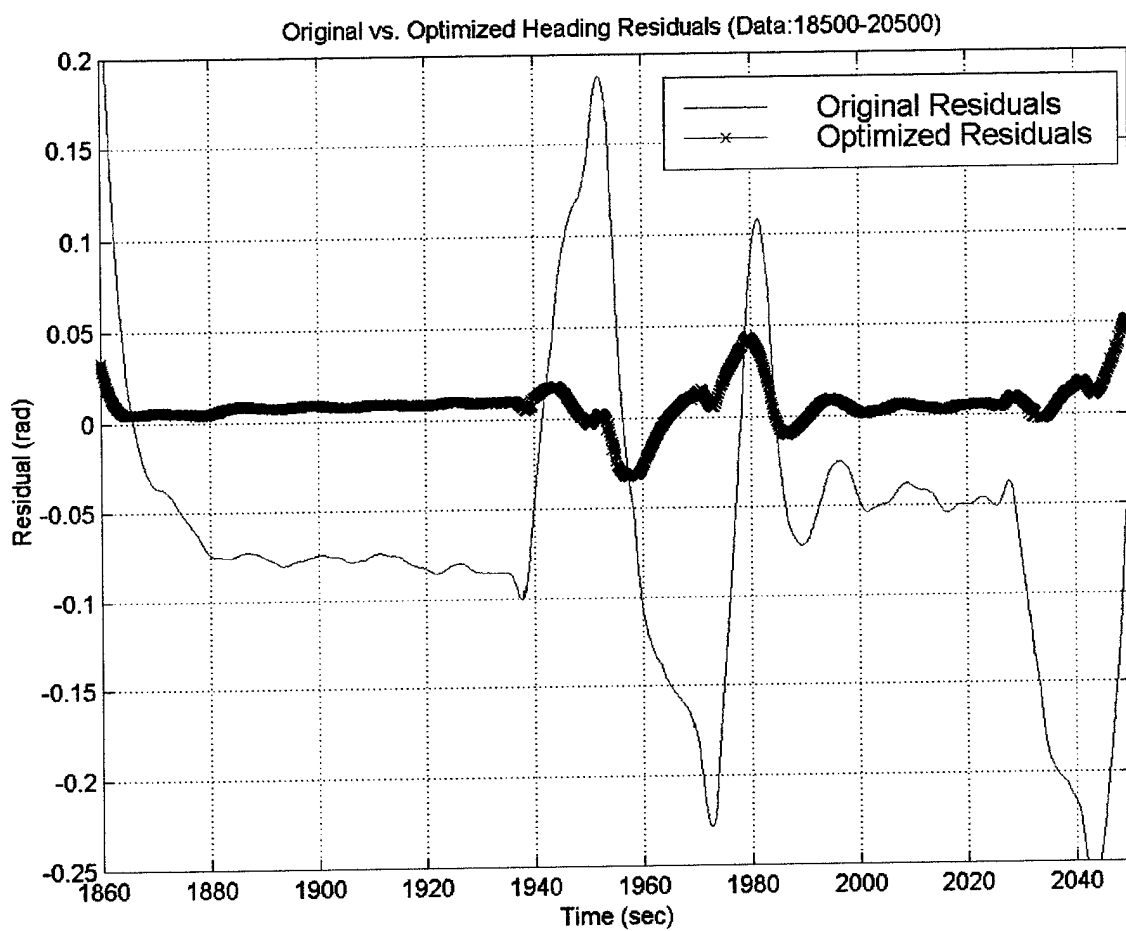


Figure 4.6 Residual Reduction by Hydrodynamic Coefficient Optimization
(Data Interval: 18500-20500)

Notes for Figure 4.6: Source Code Name – “opt_res_reduc.m”
Scalar Coeffs. Used – [3.999 3.815 -10.00 -1.457 0.221]

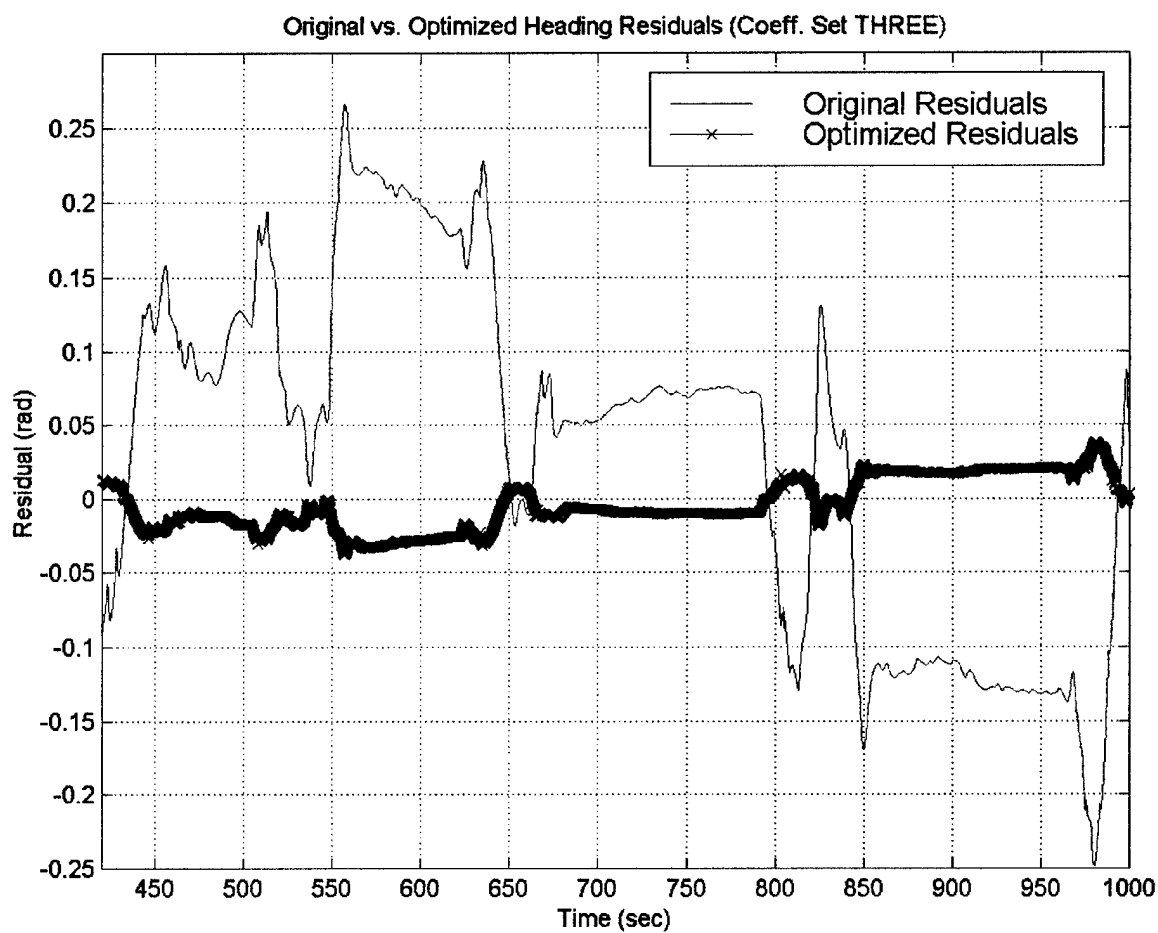


Figure 4.7 Coefficient Set THREE Performance Over Extended Interval
Data: 4000-10000, Heading Residual Response

Notes for Figure 4.7: Source Code Name – “opt_res_reduc.m”
Scalar Coeffs. Used – [-0.046 3.951 -0.408 -1.394 0.551]

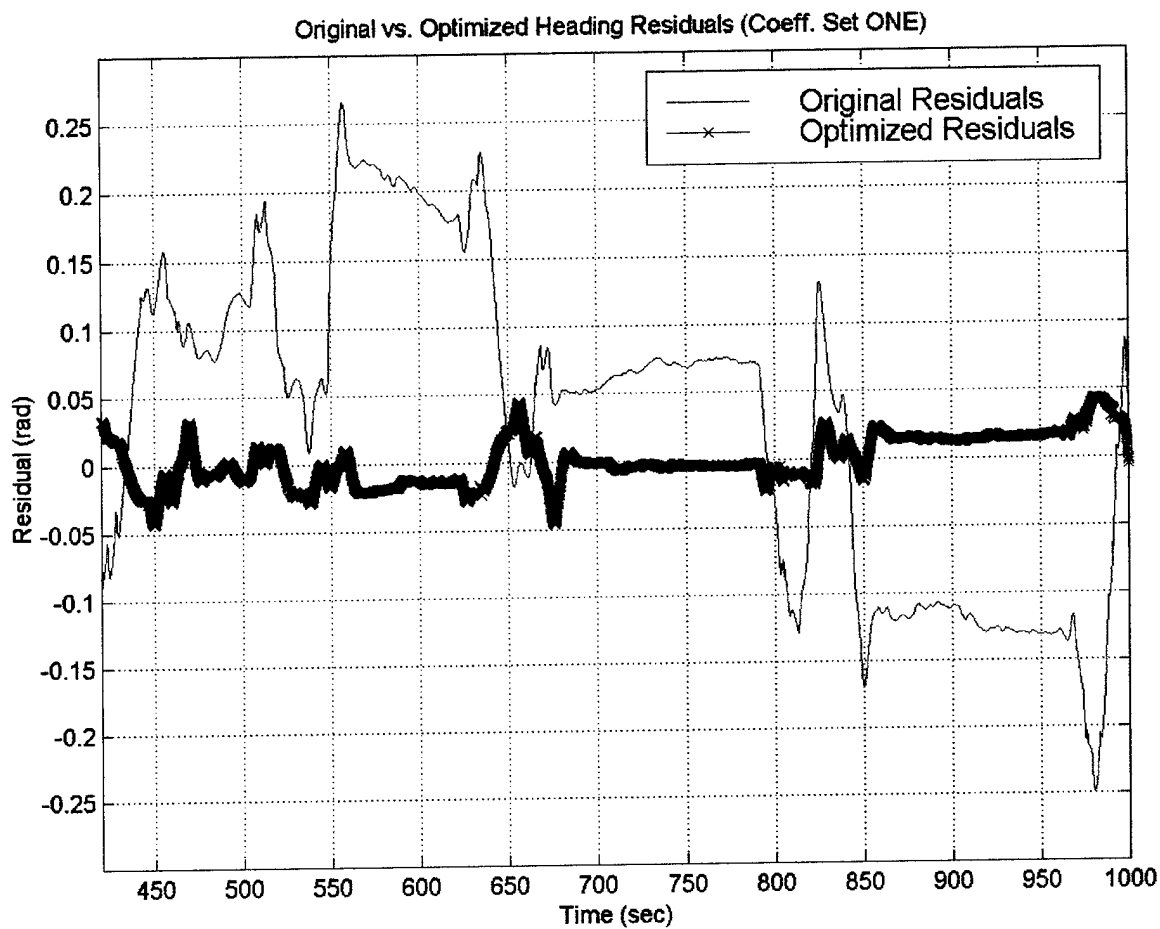


Figure 4.8 Coefficient Set ONE Performance Over Extended Interval
Data: 4000-10000, Heading Residual Response

Notes for Figure 4.8: Source Code Name – “opt_res_reduc.m”
Scalar Coeffs. Used – [-0.0339 3.946 -0.248 -1.375 1.548]

Figures 4.2 through 4.7 display excellent residual reduction characteristics for the five coefficient sets. Although residual reduction is very important in detecting faults during maneuvering, it is still vital to be able to 'see' the fault-induced residuals through the reduced maneuvering residuals.

D. MANUAL FAULT INTRODUCTION AND DETECTION

As previously accomplished in Chapter III, a manual full-stroke fin fault was introduced into the newly optimized model-based observer design. The optimized hydrodynamic coefficients from data set THREE were used in the improved observer design due to their residual reduction characteristics as shown in Table 4.2 and Figures 4.4 and 4.7. In order for the model-based observer design to accurately detect faults, the model-based observer must adequately amplify the residuals due to the implemented fault to a degree where threshold tolerance levels would be exceeded. The fault scenarios from Chapter III are again used here for analysis and they are listed again for familiarization:

- *Data Set: 5200:5800* – Fault occurs during the maneuver and remains constant beyond maneuver completion. (Figure 4.9)
- *Data Set: 7500:8750* – Fault occurs before the maneuver and remains constant beyond maneuver completion. (Figure 4.10)
- *Data Set: 9250:10250* – Fault occurs during the maneuver and is corrected prior to maneuver completion. (Figure 4.11)
- *Data Set: 10750:11500* – Fault occurs before the maneuver and is corrected prior to maneuver completion. (Figure 4.12)
- *Data Set: 18500:20500* – Fault occurs during the maneuver and is corrected prior to maneuver completion. (Figure 4.13)

Figures 4.9 through 4.13 display the model-based observer results when a fault is manually introduced into the steering subsystem. Graphical analysis of these figures shows that there is a considerable increase in the clarity between the fault-free residuals and the fault residuals by using the optimized observer design over the original design. Optimization of the hydrodynamic coefficients provided a better estimate of the coefficient values that constitute the estimated dynamic and control matrixes. The model-based observer design was significantly improved by utilizing optimization, but reliable and robust fault detection is still questionable due to the irregularities in the optimally generated fault-free residuals. It would be a difficult task to develop a fault detection algorithm that could reliably detect a fault as shown in Figures 4.9 through 4.13. The lack of a constant baseline residual value to use for fault residual comparison makes fault detection difficult even with the use of the optimized model-based observer design.

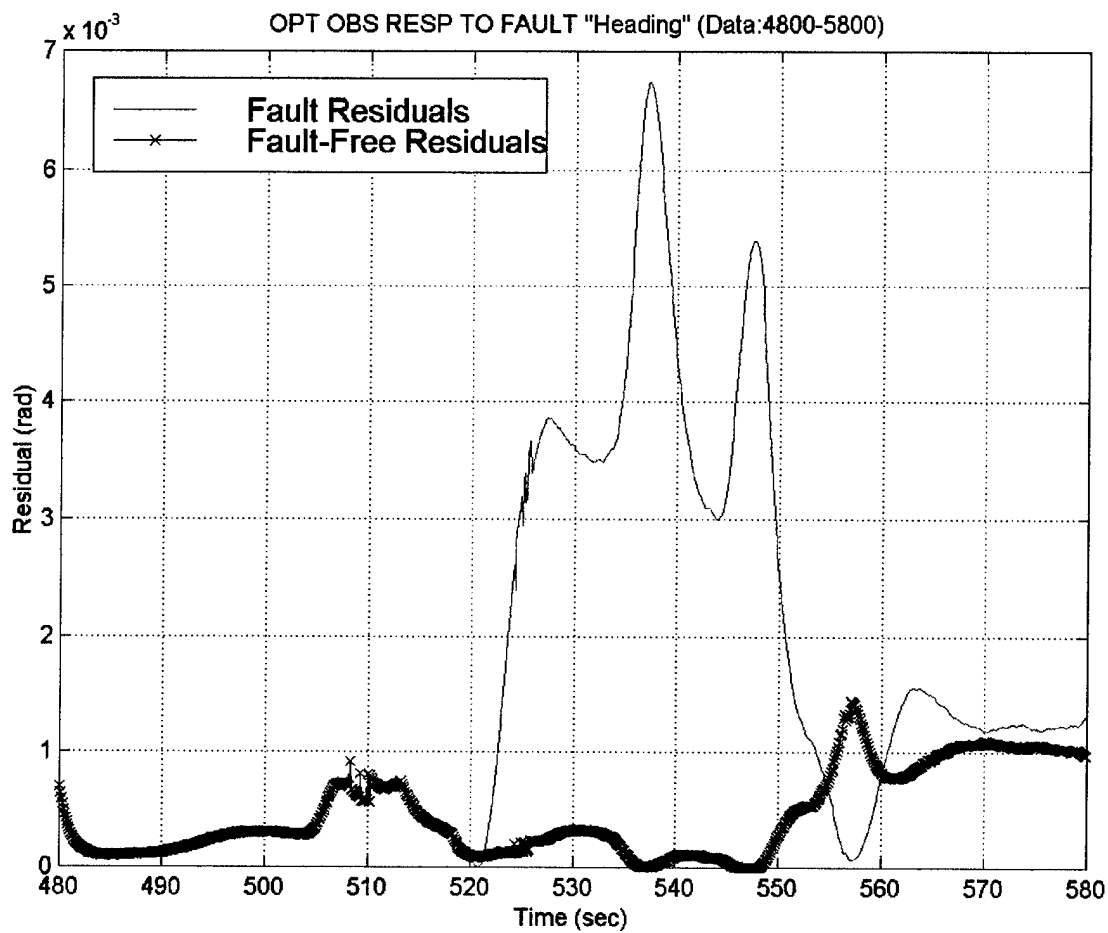


Figure 4.9 Optimized Observer Design Response to Fin Fault “Sideslip”
(Data Interval: 4800-5800)

Notes for Figure 4.9: Source Code Name – “opt_faults.m”

Scalar Coeffs. Used – [-0.046 3.951 -0.408 -1.394 0.551]

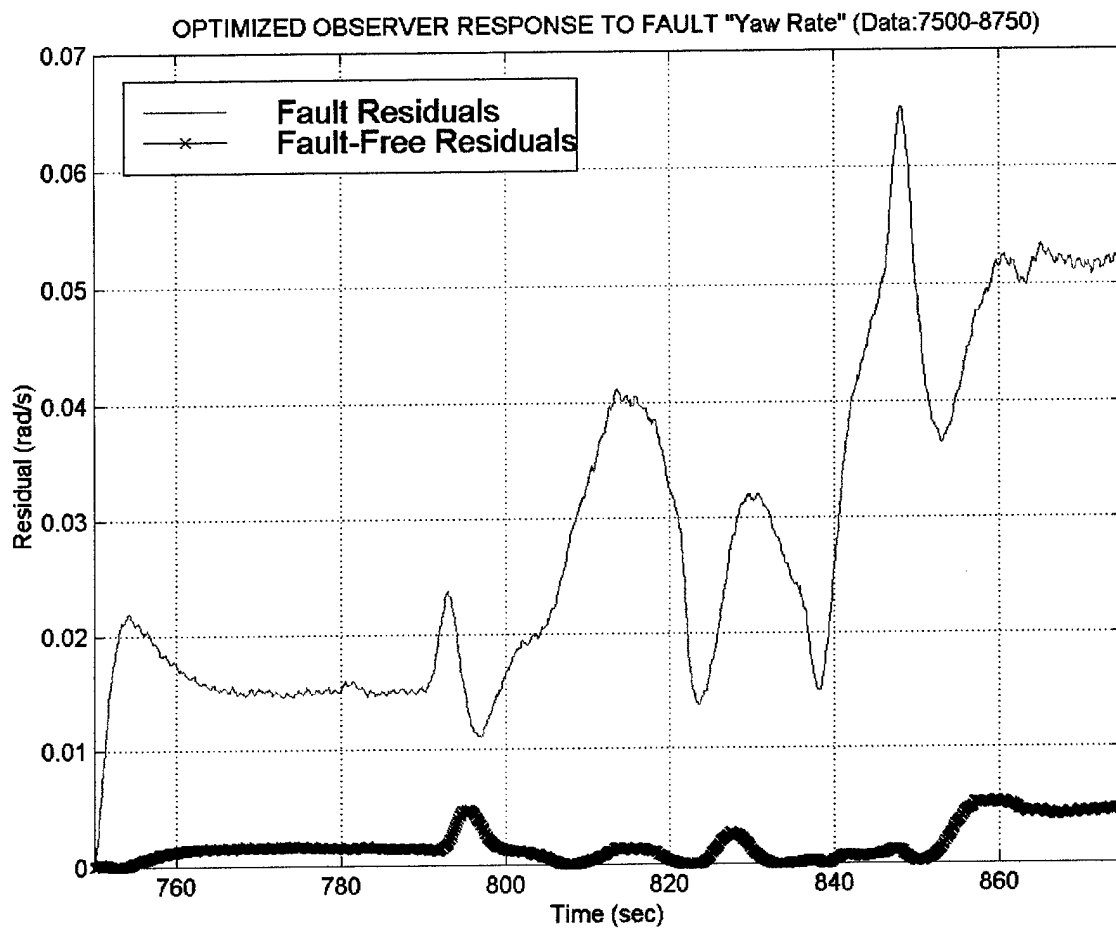


Figure 4.10 Optimized Observer Design Response to Fin Fault "Yaw Rate"
(Data Interval: 7500-8750)

Notes for Figure 4.10: Source Code Name – "opt_faults.m"

Scalar Coeffs. Used – [-0.046 3.951 -0.408 -1.394 0.551]

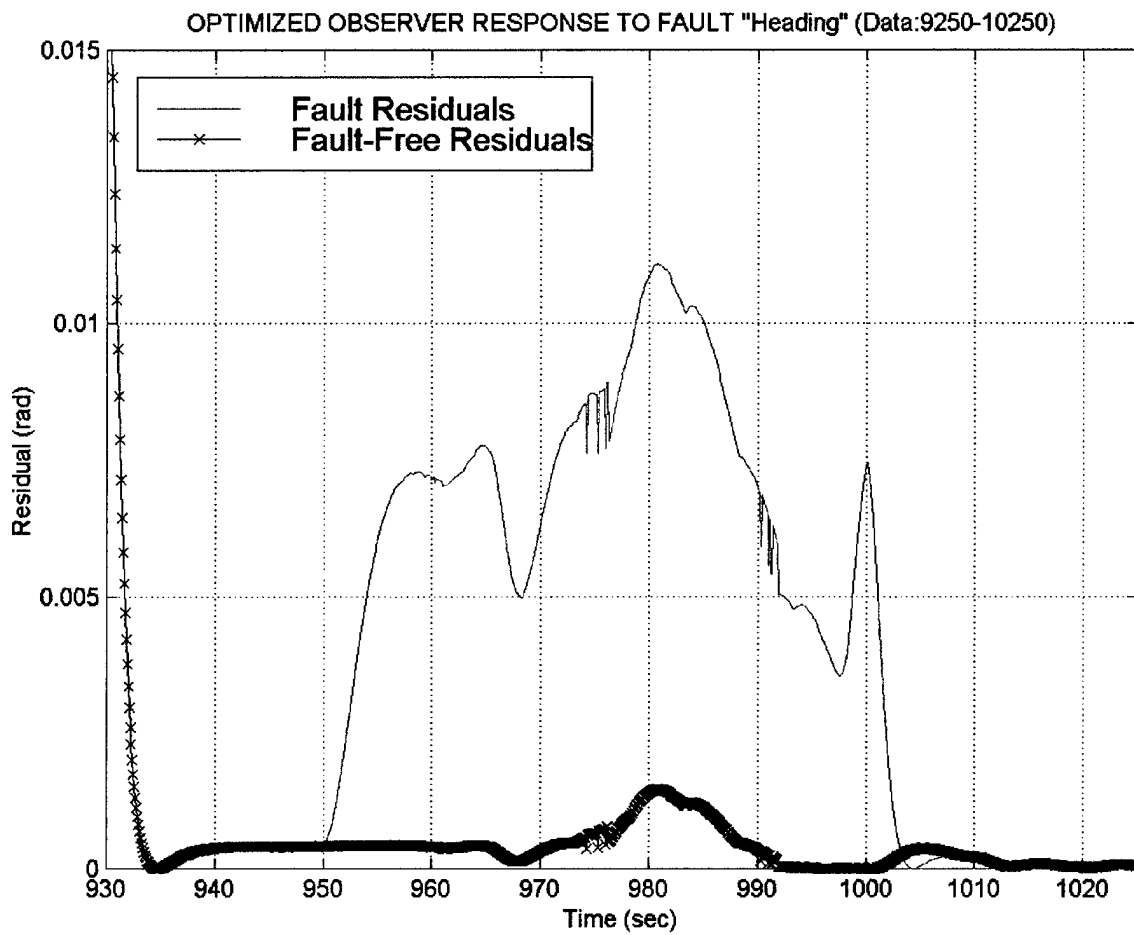


Figure 4.11 Optimized Observer Design Response to Fin Fault "Heading"
(Data Interval: 9250-10250)

Notes for Figure 4.11: Source Code Name – "opt_faults.m"
Scalar Coeffs. Used – [-0.046 3.951 -0.408 -1.394 0.551]

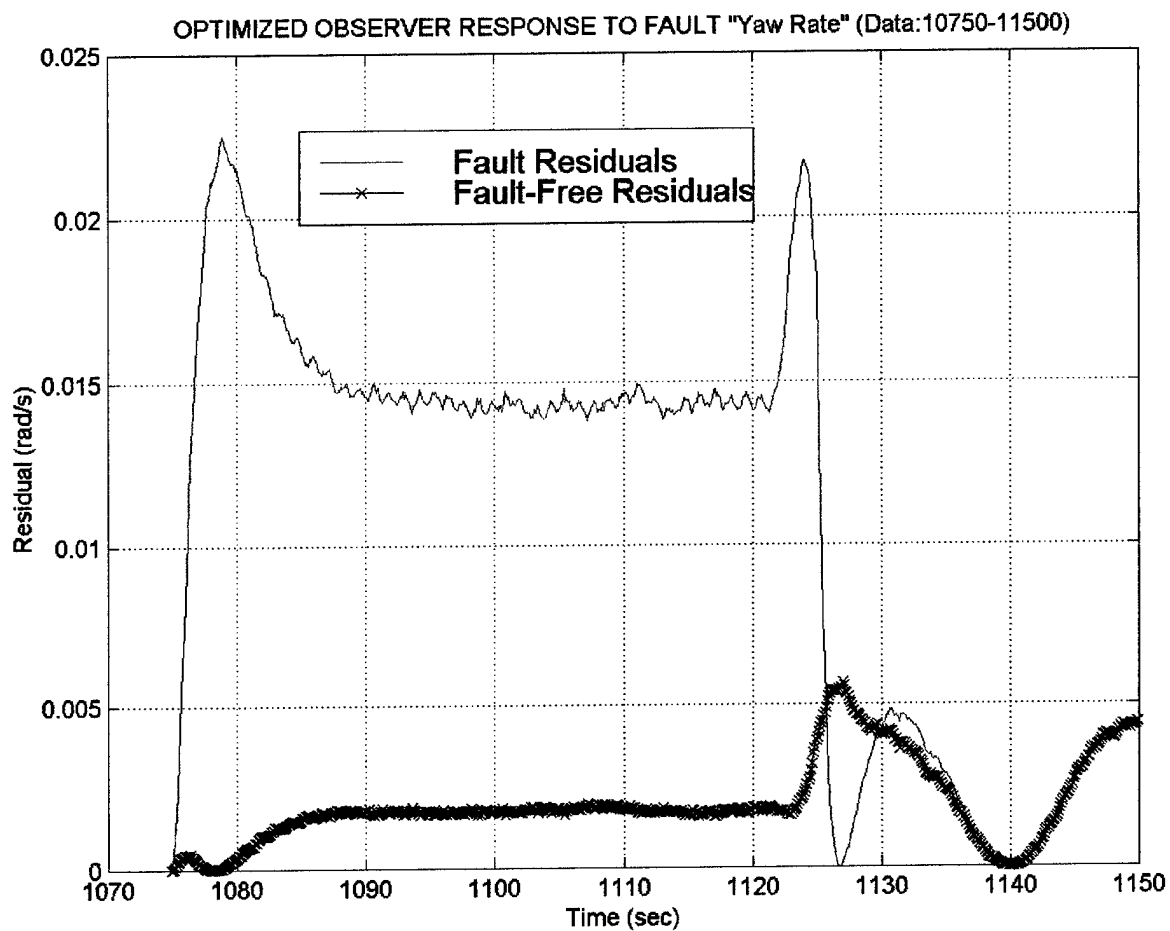


Figure 4.12 Optimized Observer Design Response to Fin Fault "Yaw Rate"
(Data Interval: 10750-11500)

Notes for Figure 4.12: Source Code Name – "opt_faults.m"

Scalar Coeffs. Used – [-0.046 3.951 -0.408 -1.394 0.551]

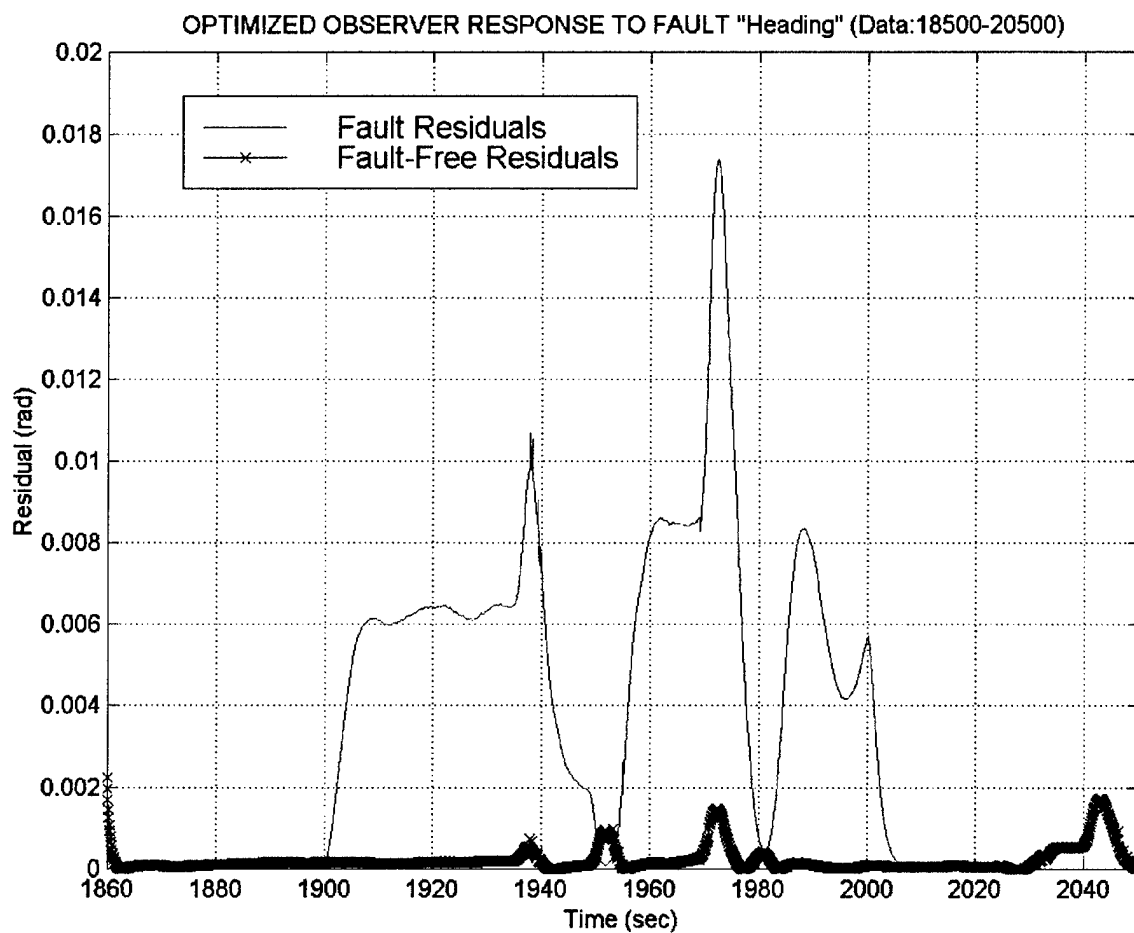


Figure 4.13 Optimized Observer Design Response to Fin Fault "Heading"
(Data Interval: 18500-20500)

Notes for Figure 4.13: Source Code Name – "opt_faults.m"

Scalar Coeffs. Used – [-0.046 3.951 -0.408 -1.394 0.551]

E. CONCLUSIONS

This chapter provided many insights into the use of a model-based observer for fault detection during vehicle maneuvers. By implementing real 21UUV sensor data into the original model-based observer as designed by Melvin (1998), it was shown that this design produced large magnitude residuals throughout the length of the vehicle's run. More importantly, the residuals produced by this observer design during vehicle maneuvers were extremely large and irregular in periodicity. It was also shown that manually introduced full-stroke fin faults were not detectable throughout the five data set intervals that were previously analyzed in Chapter II. Failure to detect these faults led to the investigation into the inaccuracies between the measured and estimated dynamic (**A**) and control (**B**) matrixes of the steering subsystem model that are largely responsible for residual generation.

It was proposed that certain hydrodynamic coefficients that constitute the estimated **A** and **B** matrixes of the steering subsystem model contained inaccuracies that led to the increase in residual generation during vehicle maneuvering. The inaccuracies in these matrixes resulted in improper modeling of the vehicle's steering subsystem as the vehicle maneuvered through large angles of heading throughout its run. Optimization of five chosen hydrodynamic coefficients resulted in significant residual reduction during vehicle maneuvers. Residual reductions of up to 71% were achieved over the length of the entire data set. Substituting the optimized hydrodynamic coefficients into the original model-based observer design proved that significant numerical and graphical improvements in residual reduction were attainable by attempting to accurately model the **A** and **B** matrixes of the steering subsystem model by use of optimization.

Upon implementation of a full-stroke fin fault into the steering subsystem, it was shown that the optimized model-based observer design had an improved capability to detect faults during maneuvering data intervals. Although the clarity and recognition of the actual faults increased by use of optimization, the reliability to detect the faults in a timely manner was suspect due to the lack of constant baseline residuals generated during fault-free time intervals. Without the ability to accurately compare fault-free residuals to fault-induced residuals, detection of a full-stroke fin fault is unattainable with the current model-based observer design. This is due to the inability to set fault threshold levels based upon constant residual values generated over the time of a vehicle maneuver.

It is proposed that the reason for limitations in the original model-based observer design to reliably detect faults due to inconsistent baseline residual values is twofold:

- The model-based observer design is based upon a linear model assumption. With the large angles of heading experienced by the vehicle throughout the length of its data run, the linear model breaks down because of the model calculations based upon assumption of small angles of incidence. Without the inclusion of large angles into the model design, the existing system model is flawed and incapable of accurately estimated state values that accurately define the model.
- The gain matrix that is calculated for the observer model is based upon a nominal model, where variances in the noise and measurement matrixes are neglected.

Due to this nominal assumption, the gain matrix calculated upon nominal values is inherently flawed and does not satisfactorily represent the actual measurement model during maneuvering.

Solutions to these two reasons for the limiting performance of the model-based observer design must be found in order for reliable and robust fault detection to take place during vehicle maneuvers.

V. UTILIZATION OF THE EXTENDED KALMAN FILTER FOR FAULT DETECTION

It was shown in previous chapters that robust and reliable fault detection in the steering subsystem of 21UUV was not entirely attainable using prior linear designs and model-based observer techniques. Optimization of certain hydrodynamic coefficients that constitute the dynamic and control matrixes of the system model significantly reduced residual generation from model uncertainty but not to a degree where fault residuals could be detected with 100% certainty due to insuppressible residuals that continued be generated by the model. Investigation into the remaining uncertainties of the steering model design leads to the inevitable conclusion that the nominalizations of the system and measurement errors present in the model add to the generation of residuals. A technique perfectly tailored to the problem of filtering out these errors is the optimal linear estimator, the Kalman filter. By implementation of an often-used Kalman filter algorithm, it will be shown that the model-based observer design may be improved further, and to a degree where reliable and robust fault detection in the steering subsystem of the 21UUV is attainable.

A. BASIC INTRODUCTION TO KALMAN FILTERING

A Kalman filter is a data processing algorithm that optimally and recursively updates the values of state variables given input measurements corrupted by noise and a model with uncertainty. The state variables that define the steering subsystem of 21UUV cannot be measured directly, but must be calculated based upon sensor measurement data taken by the vehicle. There exists an amount of uncertainty between the calculated states

and the actual measured states that is identified as modeling inaccuracies and is defined as system noise, or system error. Also, measurements taken by the vehicle are degraded by the presence of noise, biases, and instrument inaccuracies. These measurement uncertainties are defined as measurement noise, or measurement error. Maybeck (1996) describes the abilities of the Kalman filter to combine all available measurement data with knowledge of system and measurement devices, to produce an estimation of desired variables in such a manner as to statistically minimize error.

Figure 5.1 was adapted from Maybeck (1996) and depicts the basic architecture in which Kalman filtering would be used to improve steering subsystem performance by minimizing errors in the system. The steering subsystem is being driven by the inputs fed to it by the steering controller, while measuring devices provide actual state variable data taken from the vehicle. Knowledge of the system inputs, measurements, and respective noise is utilized to provide optimal estimates of the system, as shown.

A Kalman filter utilizes all available information that can be provided to it in order to produce the best possible, or optimal, estimate available. The Kalman filter uses three types of information to process available measurements to estimate desired state variables. These three types of information are (Maybeck, 1996):

- Knowledge of the measurement devices and system dynamics
- Information concerning initial conditions of variables of interest
- Statistical description of system noises, measurement errors, and uncertainty in the system model.

The recursive nature of the Kalman filter means that the filter does not require storage and processing of large amounts of previous data. This aspect of the Kalman

filter prevents excessive requirements for CPU calculation time. Maybeck (1996) defines 'filter' as actually being a 'data processing algorithm'. Essentially, the filter is just a computer program used in the CPU that incorporates discrete-time measurement data rather than continuous time data.

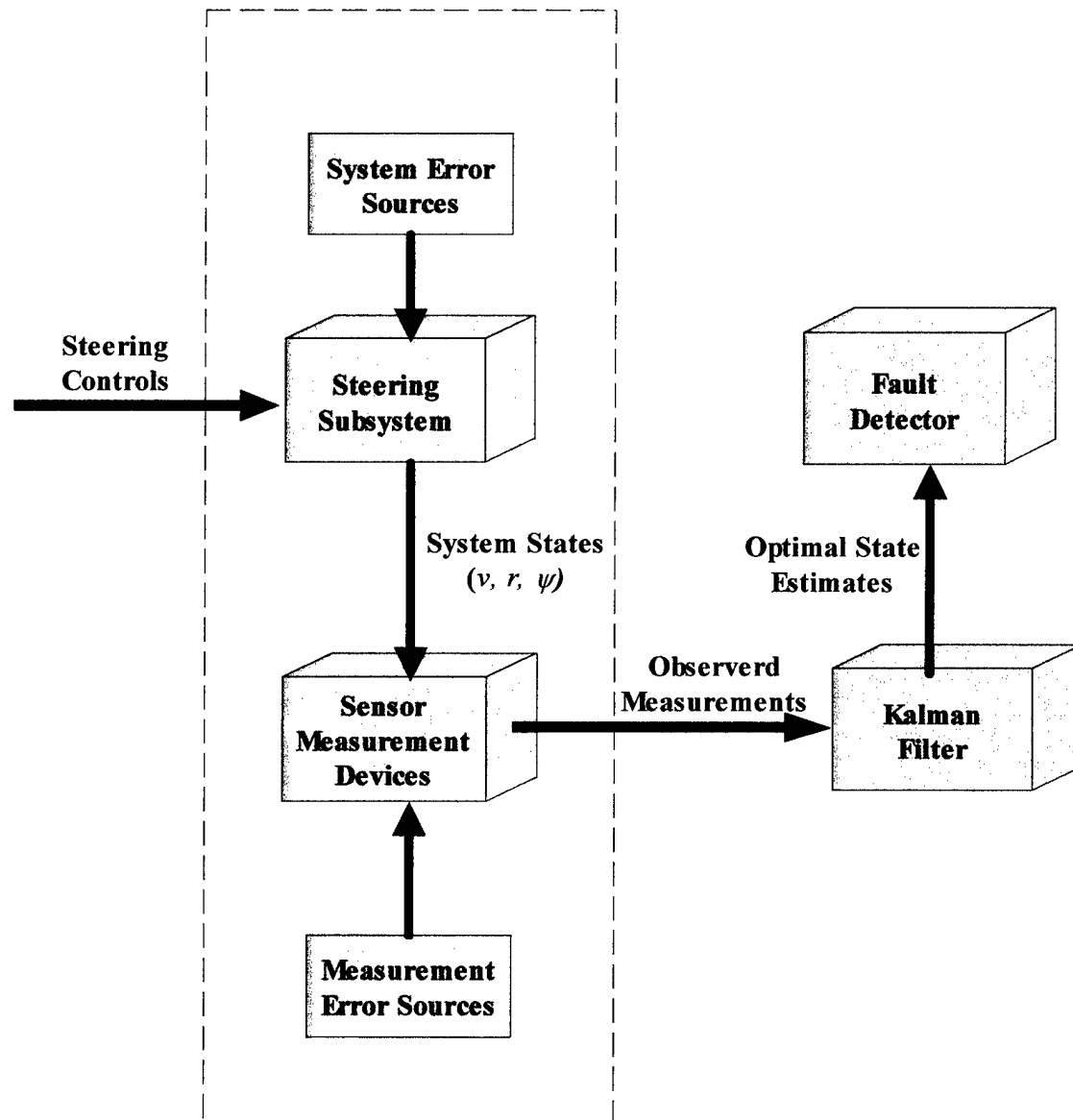


Figure 5.1 Steering Subsystem Kalman Filter Architecture

The Kalman filter algorithm used in this work was adapted from Gelb (1974) and can be viewed in any of the referenced code that was written to generate the graphs shown in the remainder of this chapter. If further information or study into Kalman filtering concepts is desired, the reader is strongly encouraged to seek out the paper written by Maybeck (1996).

B. ANALYSIS OF BASIC KALMAN FILTER FAULT DETECTION

A very important feature of the Kalman filtering technique is the Kalman calculation of a normalized relative error. This normalized relative error provides a correctly squared, scaled, and weighted measure of error calculated over a given data interval by use of the combination of all three state variable residual values. Normalized error (NE) is given as:

$$NE = v' S v,$$

$$S = [CPC' + R]^{-1},$$

$$P = E\{\tilde{x}\tilde{x}'\} \Rightarrow \text{Error Covariance Matrix}$$

P^{-1} is related to the Information Matrix and is high when the estimation error is low (FIM, Bar Shalom). Essentially, it is an accurate representation of the scaled weighting equation formulated in Chapter IV, but with no uncertainty in dimensional equivalency. This normalized relative error value becomes a very powerful tool for residual generation and fault detection for the model-based observer design. Observer performance may be singularly evaluated by evaluation of the normalized relative error, without much need to view independent state residual values. The evaluation of the normalized relative error

relied upon heavily in this work, but subsequent evaluation of individual state residuals will also be included.

An initial graphical study of the residual reduction characteristics of the Kalman filter shows significant reduction in residual values over all data set intervals. Figures 5.2 through 5.5 display the residual reduction performance of implementing a basic Kalman filter into the original model-based observer design.

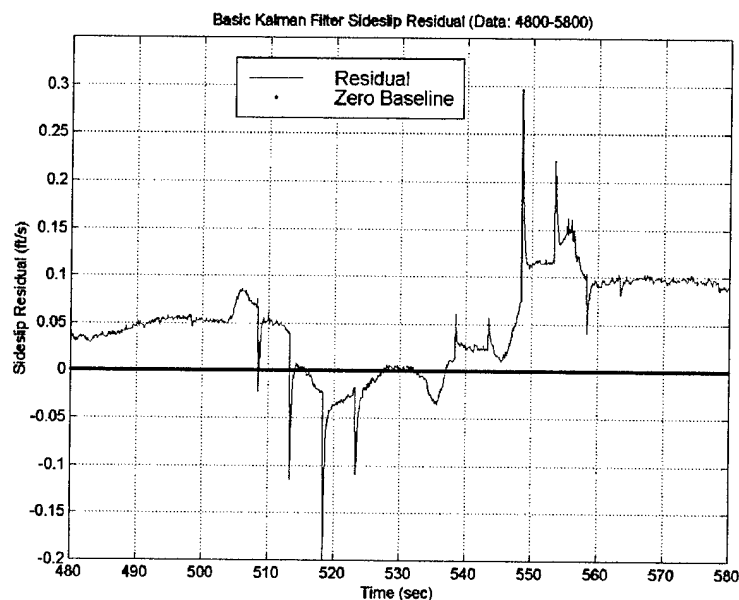


Figure 5.2 Basic Kalman Filter Sideslip Residual Generation (Data: 4800-5800)

Note on Figure 5.2: Source Code Name – “res_test.m”, All Original Values Used

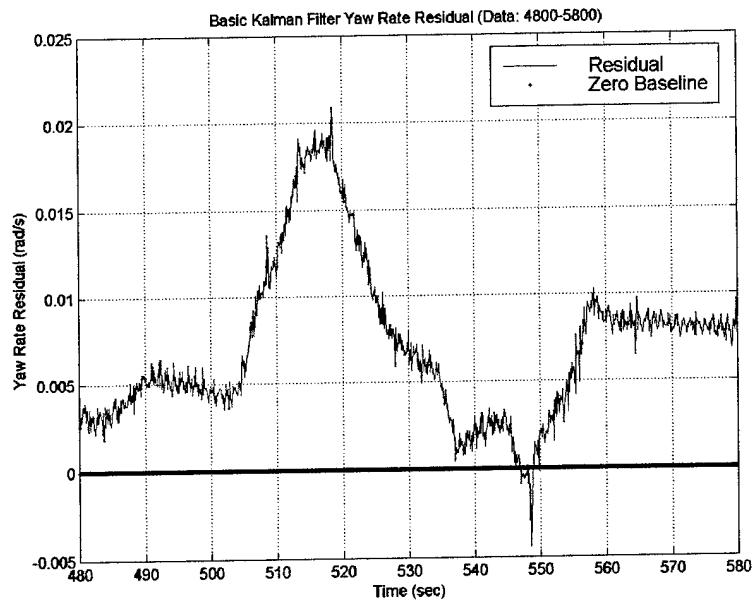


Figure 5.3 Basic Kalman Filter Yaw Rate Residual Generation (Data: 4800-5800)

Note on Figure 5.3: Source Code Name – “res_test.m”, All Original Values Used

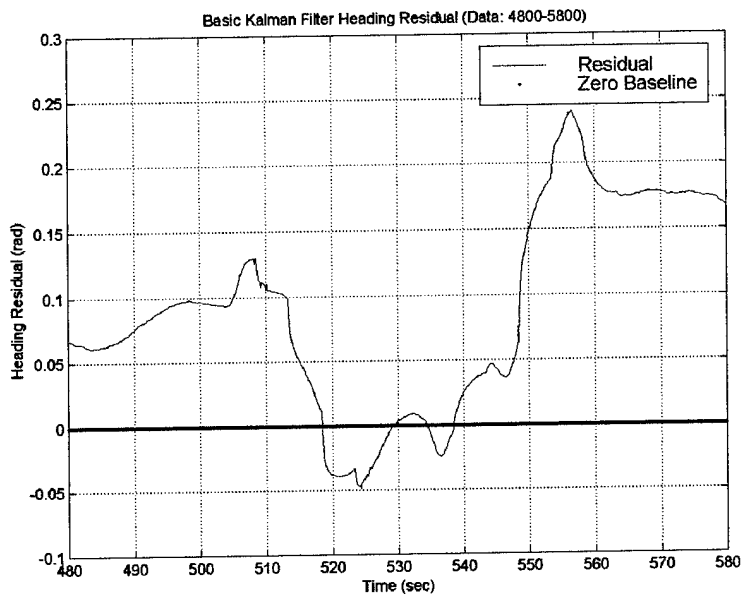


Figure 5.4 Basic Kalman Filter Heading Residual Generation (Data: 4800-5800)

Note on Figure 5.4: Source Code Name – “res_test.m”, All Original Values Used

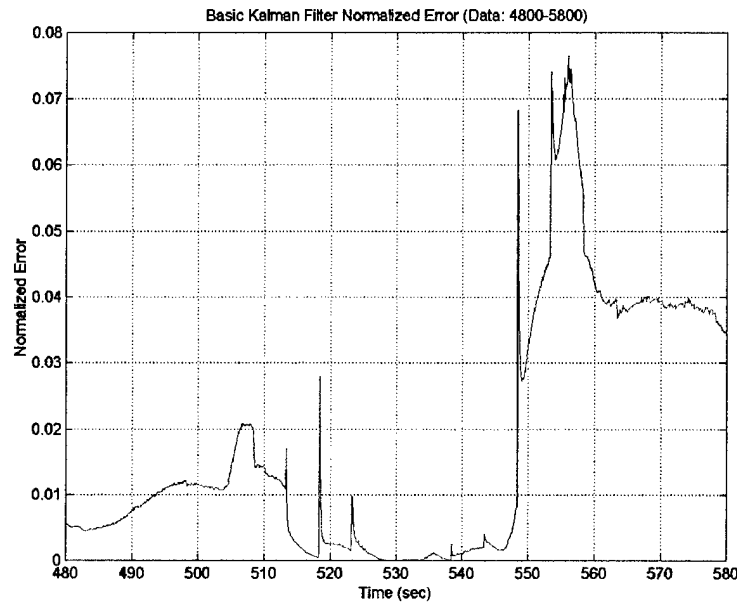


Figure 5.5 Basic Kalman Filter Relative Error Values (Data: 4800-5800)

Note on Figure 5.5: Source Code Name – “res_test.m”, All Original Values Used

The data evaluated in these figures was from data set interval ONE, without fault introduction into the steering subsystem. Although significant residual magnitude was reduced over this maneuvering data interval, there still exist significant fluctuations in the residuals in comparison to a zero baseline reference value. The normalized relative error over the data interval is small, but fluctuations in its magnitude may prevent proper fault detection.

In order to actually gauge the fault detection performance of the basic Kalman filter, a full-stroke fin fault was introduced over data set interval ONE. As performed previously, the fault was introduced at the beginning of the data set and remained ‘on’ until the end of the data set. The resultant performance of the basic Kalman filter is shown in Figures 5.6 through 5.9. Unfortunately, the basic Kalman filter was not capable

of improving the detection possibility of a full-stroke fin fault by residual generation analysis over the maneuvering interval. Although detection of a fault by residual analysis is not reliably possible using this Kalman filter, partial fault detection is capable by analyzing the relative error produced due to the fin fault. Figure 5.9 shows the relative error of the steering subsystem due to a fin fault and the distinction between the fault-free residual and the fault residual is prominent. Designating the basic Kalman filter as the answer to fault detection challenges at this point would be premature because continued analysis of the relative error plot reveals that between time 550 and 580 seconds, the normalized relative error of the fault falls below the fault-free relative error. This situation is an example of a 'false fault-detect'. Clearly, as evaluated in the previous eight figures, the Kalman filtering technique must be improved if it is to be proved a reliable technique for detecting faults in the steering subsystem of the 21 UUV.

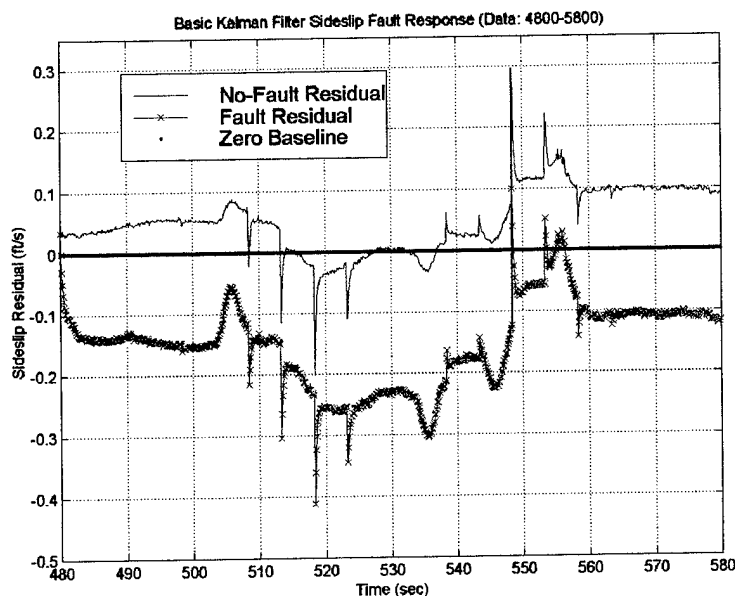


Figure 5.6 Basic Kalman Filter Sideslip Residual Fault Detection (Data: 4800-5800)

Note on Figure 5.6: Source Code Name – “res_test_both.m”, All Original Values Used

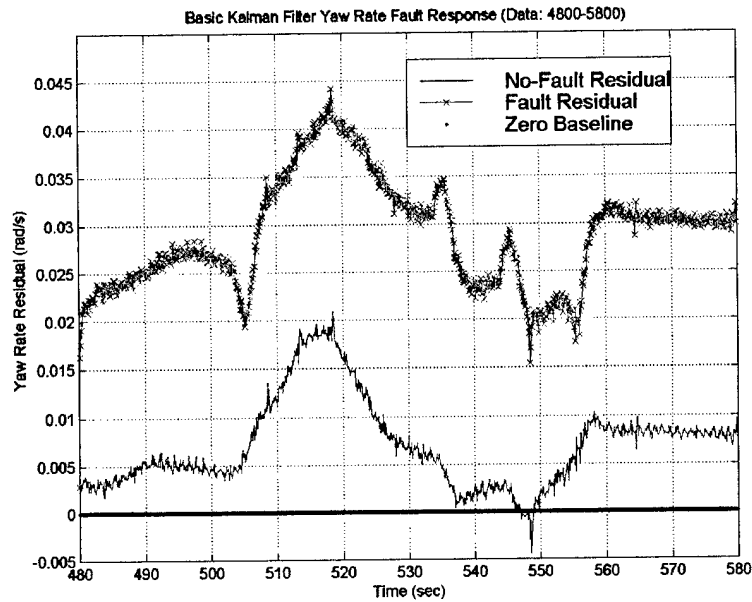


Figure 5.7 Basic Kalman Filter Yaw Rate Residual Fault Detection (Data: 4800-5800)

Note on Figure 5.7: Source Code Name – “res_test_both.m”, All Original Values Used

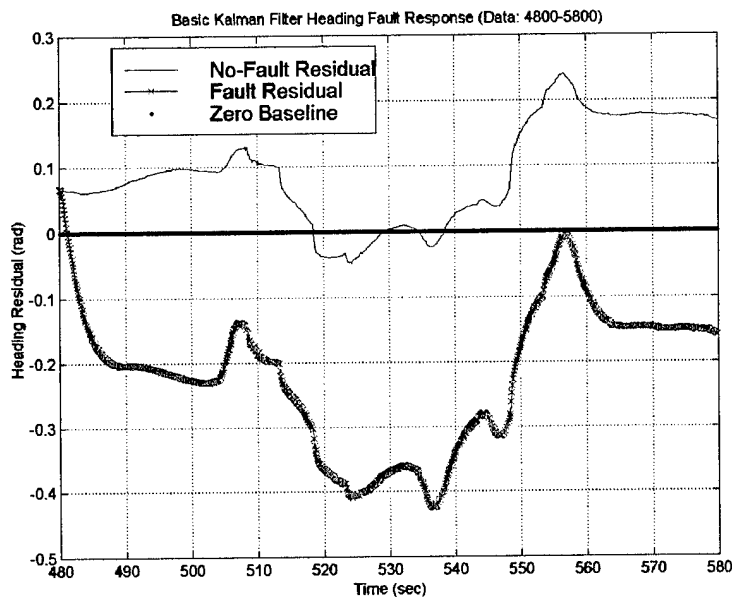


Figure 5.8 Basic Kalman Filter Heading Residual Fault Detection (Data: 4800-5800)

Note on Figure 5.8: Source Code Name – “res_test_both.m”, All Original Values Used

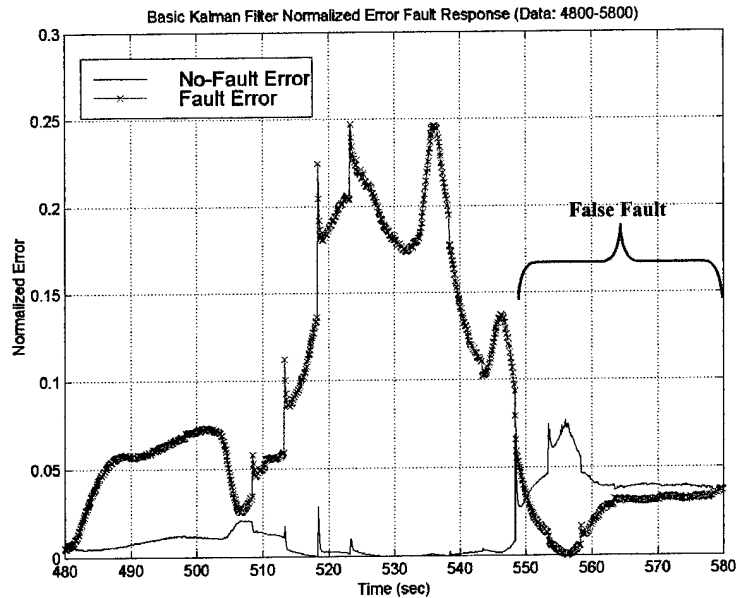


Figure 5.9 Basic Kalman Filter Relative Error Fault Detection (Data: 4800-5800)

Note on Figure 5.9: Source Code Name – “res_test_both.m”, All Original Values Used

C. ANALYSIS OF UNCERTAIN KALMAN PARAMETERS

In an attempt to improve the Kalman filter for use in steering subsystem fault detection, an analysis of the algorithm was conducted and three areas of uncertainty were designated in which it was believed residual generation might be effected significantly.

These three areas of uncertainty were the:

- System noise matrix, Q
- Measurement noise matrix, R
- Scalar Gain Multiplier, designated β .

Analysis of the system noise matrix and the measurement noise matrix revealed that their values were based upon nominal assumptions. The exact values of these two matrixes are unknown because they represent unknown noise and bias quantities inherent in the

system and are dependent upon operational and environmental conditions. The system noise matrix and the measurement noise matrix that were used in the previous evaluation of the basic Kalman filter performance were, respectively:

$$Q = \begin{bmatrix} 0.0478 & 0.07917 & 0 \\ 0.7917 & 16.31 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$R = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}.$$

A scalar gain multiplier was identified as being a possible improvement to fault detection by aiding in amplifying the residual response due to a fault. Essentially, if the residuals due to maneuvering were driven to near-zero magnitude and the residuals due to a fault were present, a scalar gain multiplier would greatly amplify the fault residuals, but at the same time, be ineffective for amplifying the fault-free residuals due to their small magnitude. The scalar gain multiplier would be used to multiply the gain matrix, Γ , taken from the state space representation of the steering observer model. If written in state space form, the scalar gain multiplier, β , would appear as:

$$x_{k+1} = \Phi x_k + \beta * \Gamma u_k.$$

The matrixes Φ and Γ represent the discrete-time forms of the continuous-time state matrixes **A** and **B**, respectively. In order to improve upon the performance of the Kalman

filter by attempting to find better values for **Q**, **R**, and β , optimization of the model-based observer containing the Kalman filter was performed.

D. OPTIMIZATION OF **Q**, **R**, AND β

1. Parameters For Optimization of the Kalman Filter

Optimization of **Q**, **R**, and β was accomplished by identifying the values for which optimization might improve the Kalman filter's performance. Once again, the *Matlab* function 'constr' was utilized for the optimization function associated with this process. The final variables for optimization were chosen after innumerable executions of scheme combinations, in which individual analysis was conducted in order to determine which optimization parameter set produced the most reliable and robust fault detection for the steering subsystem of the 21UUV. The optimization values for the **Q** and **R** matrixes are scalar multipliers of the nominal **Q** and **R** values. The scalar gain multiplier, β , was optimized around the nominal value of one. The final vector (**alpha**) containing all the optimization variables for **Q**, **R**, and β is defined as:

$$\mathbf{alpha} = [\mathbf{alpha}(1) \quad \mathbf{alpha}(2) \quad \mathbf{alpha}(3) \quad \mathbf{alpha}(4) \quad \mathbf{alpha}(5) \quad \mathbf{alpha}(6) \quad \mathbf{alpha}(7)],$$

$$\mathbf{R} = \begin{bmatrix} \mathbf{alpha}(1) & 0 & 0 \\ 0 & \mathbf{alpha}(2) & 0 \\ 0 & 0 & \mathbf{alpha}(3) \end{bmatrix},$$

$$\mathbf{Q} = \begin{bmatrix} \mathbf{alpha}(4) * .0478 & \mathbf{alpha}(5) * .7917 & 0 \\ \mathbf{alpha}(5) * .7917 & \mathbf{alpha}(6) * 16.31 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\beta = \mathbf{alpha}(7).$$

The range for optimization was set from 0.0001 to 20.0 for the scalar multiplier of each **alpha** component pertaining to the values of **R**, whereas the range for optimization for the remaining values of **alpha** was set from 0.0001 to 10.0 for the scalar multiplier of their nominal values. Initial starting values for the components of **alpha** were each unity. This ensured that the optimization would begin about the original values of the basic Kalman filter design and without influence from the scalar gain multiplier.

In view of the significant residual reduction characteristics obtained from using previously optimized hydrodynamic coefficients from the model-based observer design, the optimized hydrodynamic coefficients from data set interval ONE were implemented into the optimization of the Kalman filter design. Since the Kalman filter design still incorporates the modeling dynamics of the 21UUV, the utilization of the optimized hydrodynamic coefficients significantly reduced the residuals generated with the Kalman filter. Data set interval ONE was chosen for inclusion into the Kalman filter design due to its superior fault-free residual reduction characteristics and its superior fault residual amplification tendencies.

As utilized in the optimization of hydrodynamic coefficients, a performance index will be the objective function for optimization. The performance index for the optimization of **alpha** is of the same design as the previous, but the Kalman filter normalized relative error, with and without a fault, will constitute the index. Since the Kalman filter calculates this squared, scalar relative error that takes into account the three state residuals, there is no reason to formulate a weighting matrix or to attempt to accurately non-dimensionalize the residual values.

The performance index that is the objective function for the optimization of **alpha** is defined as:

$$f = relerr_nofault + \frac{1}{relerr_fault}.$$

This performance index once again amplifies the residuals generated from a fault condition while reducing the fault-free residuals generated from vehicle maneuvering.

2. Tabular and Graphical Results for Q, R, and β Optimization

Again, innumerable optimization runs were conducted over all data set intervals in order to obtain the optimized values for **alpha** that provided the greatest improvement in model design. Only the final results taken from the best **alpha** will be listed here. If further manipulation or evaluation of any of the defining parameters of optimization is desired, the *Matlab* code written for these calculations and their subsequent optimized results is contained in Appendix C. Also, each graph contained in this chapter includes the name of the code that generated the graph's respective plot.

The final optimized components of **alpha** and their respective performance improvements to residual reduction are first listed in tabular form for numerical comparison purposes. Graphical representations of the performance of the optimized components of **alpha** will be included later in this chapter.

a. *Optimized Alpha Components for Each Data Set Interval*

Optimization over each data set of the components of vector **alpha** was conducted using the 21UUV sensor and measurement data. Each data set interval had its

own distinct **alpha** component values that were found to optimally minimize the normalized relative error, *rel*, pertaining to that data interval's residual generation characteristics. Table 5.1 lists the final optimized scalar multipliers for the chosen values of the **Q** matrix, the **R** matrix, and the scalar gain multiplier β .

Data Set	Alpha(1) R(1,1)	Alpha(2) R(2,2)	Alpha(3) R(3,3)	Alpha(4) Q(1,1)	Alpha(5) Q(1,2),Q(2,1)	Alpha(6) Q(2,2)	Alpha(7) β
4800-5800	20.00	2.2282	1.6919	10.00	1.2232	8.1224	1.4964
7500-8750	20.00	4.7403	4.3996	10.00	0.0700	8.9135	1.6679
9250-10250	20.00	20.00	20.00	10.00	0.0001	10.00	1.3128
10750-11500	20.00	6.0862	6.4989	10.00	1.9825	5.8062	1.5968
18500-20500	20.00	20.00	20.00	10.00	2.5477	10.00	1.3552

Table 5.1 Optimized Scalar Multipliers for Alpha

Notes for Table 5.1: Source Code Name – “opti_kalm_sets.m”

Function File Name – “opti_call.m”

$x_0 = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$, $v_{lb} = [.0001 \ .0001 \ .0001 \ .0001 \ .0001 \ .0001 \ .0001]$,

$v_{ub} = [20 \ 20 \ 20 \ 10 \ 10 \ 10 \ 10]$

b. Performance Characteristics of Optimized Alpha

The resulting performance of each of the optimized **alpha** sets is given in Table 5.2. An additional column in Table 5.2 contains the percent change in the objective function value as calculated over each respective data set interval. Due to the Kalman filter's ability to calculate an accurately scaled and normalized relative error for the residuals, it is important to analyze the change in the optimized objective function since it directly relates to the optimization of the three state variables in the steering

model for fault detection purposes. In order to numerically gauge the performance of each **alpha** set over multiple vehicle maneuvering scenarios, the objective function was calculated for each **alpha** set over data ranging from 4000 to 21900 data points. This extended analysis will allow for comparisons between the **alpha** sets on overall model performance improvement. Performance values for the basic Kalman filter were calculated using original model-based observer hydrodynamic coefficients and are included here in order to gauge the improvement in model performance by optimization of **Q**, **R**, and β and inclusion of the optimized hydrodynamic coefficients from the model-based observer design.

Data Set	Original f -set	Opti. f -set	Orig f extended	Opt f extended	% Δ f -set	% Δ f -extend	# Fun Calls
4800-5800	2.7133	0.1796	247.664	10.932	93.4% (↓)	95.6% (↓)	129
7500-8750	5.2271	0.1677	247.664	7.2479	96.8% (↓)	97.1% (↓)	129
9250-10250	46.800	4.3174	247.664	3.0656	90.7% (↓)	98.8% (↓)	169
10750-11500	3.3110	0.2182	247.664	6.0700	93.4% (↓)	97.6% (↓)	138
18500-20500	447.48	43.199	247.664	3.2057	90.3% (↓)	98.7% (↓)	217

Table 5.2 Performance Characteristics of Optimized Alpha

Notes for Table 5.2: Source Code Name – “RMS_test_sets.m”, “f_calc.m”, and “f_calc_orig.m”

Data Interval for Extended f Calculations – 4000 to 21900

3. Graphical Results for Q, R, and β Optimization

The numerical comparisons of the performance of each data set's **alpha** components is very significant, but graphical analysis of residual reduction and fault detection are necessary in order to judge which data set produced the best performing Kalman filter design. Each data set produced fantastic reduction in objective function value over its respective data interval. Due to such great performance of each data set's optimized **alpha**, the normalized relative error over each data set is included here in order to gauge the severity of residual reduction resulting from **alpha** optimization. Figures 5.10 through 5.14 display the comparison between normalized relative error with the basic Kalman filter and normalized relative error with the optimized Kalman filter. Each figure utilizes the specific **alpha** calculated by optimization of the Kalman filter's **Q**, **R**, and β respective to its data set. As can be seen from these plots, optimization of the system noise matrix, measurement noise matrix, and gain scalar multiplier produces incredible results in residual reduction.

Figures 5.10 through 5.14 show that the normalized relative error over each data set is driven nearly to a magnitude of zero by optimization of the basic Kalman filter. Since normalized relative error is a weighted measure of all three state residuals, it is apparent each data set experiences a severe reduction in maneuvering residuals after optimization. Evaluation of a fault implemented into the newly optimized Kalman filter will define the actual performance of the new design to detect faults within the steering subsystem of the 21UUV.

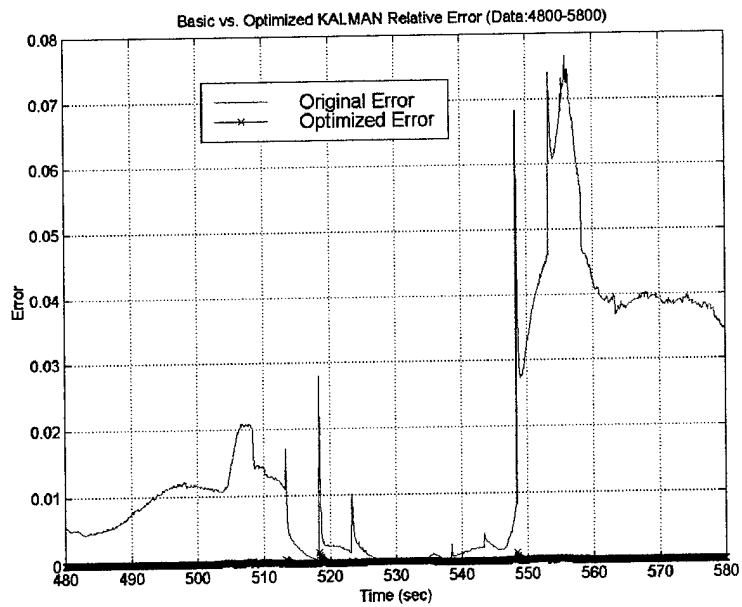


Figure 5.10 Optimized Kalman Filter Residual Reduction (Data: 4800-5800)

Notes for Figure 5.10: Source Code Name – “res_comp_sets.m”
Alpha Set Used: [20 2.23 1.69 10 1.22 8.12 1.50]

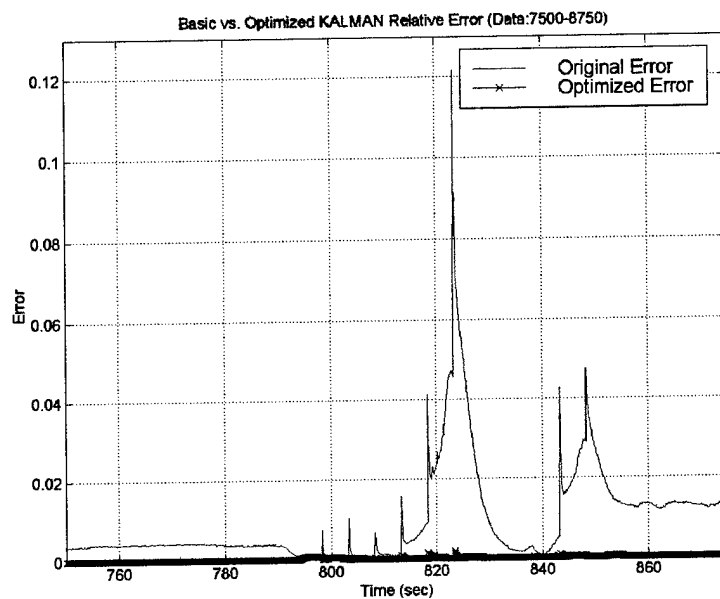


Figure 5.11 Optimized Kalman Filter Residual Reduction (Data: 7500-8750)

Notes for Figure 5.11: Source Code Name – “res_comp_sets.m”
Alpha Set Used: [20 4.74 4.40 10 0.070 8.91 1.67]

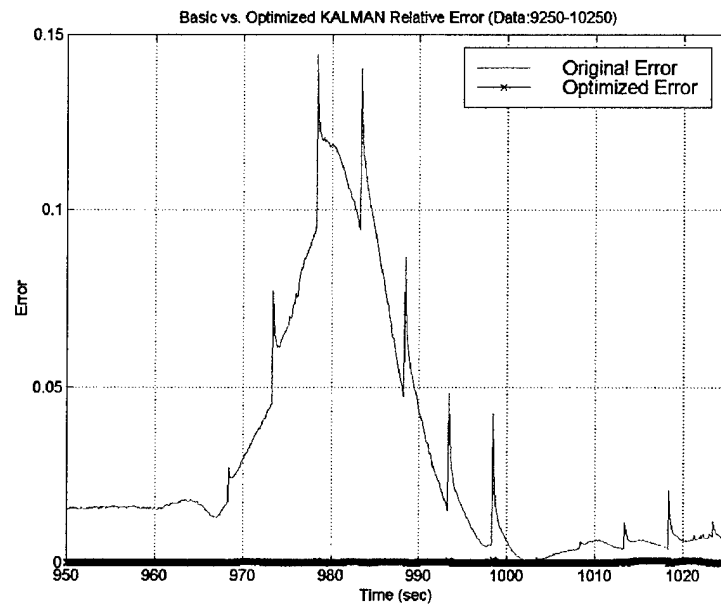


Figure 5.12 Optimized Kalman Filter Residual Reduction (Data: 9250-10250)

Notes for Figure 5.12: Source Code Name – “res_comp_sets.m”

Alpha Set Used: [20 20 20 10 0.0001 10 1.31]

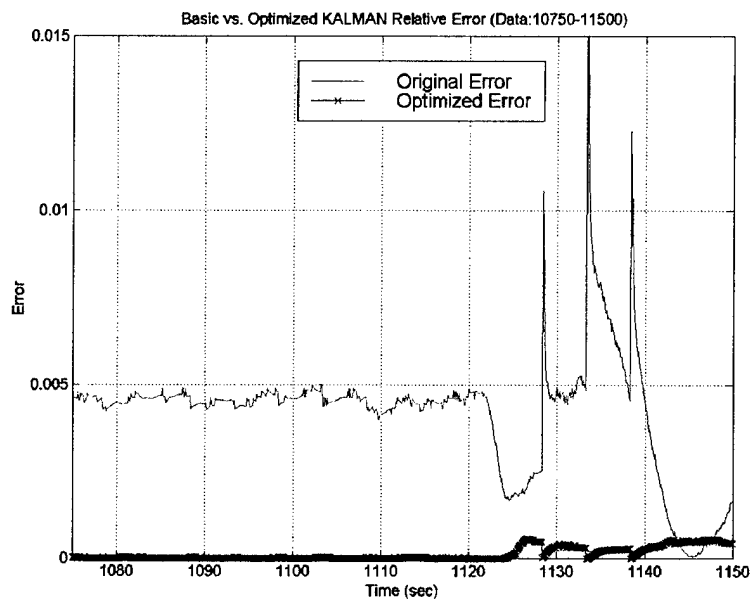


Figure 5.13 Optimized Kalman Filter Residual Reduction (Data: 10750-11500)

Notes for Figure 5.13: Source Code Name – “res_comp_sets.m”

Alpha Set Used: [20 6.09 6.50 10 1.98 5.81 1.60]

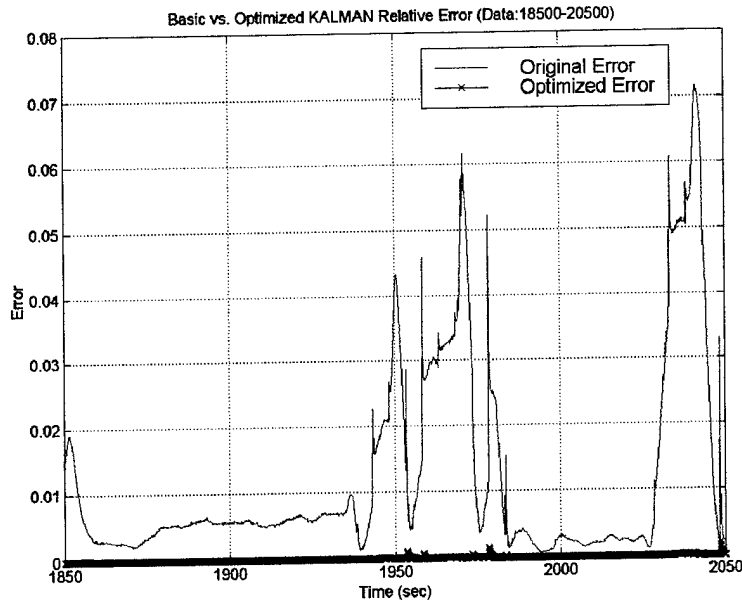


Figure 5.14 Optimized Kalman Filter Residual Reduction (Data: 18500-20500)

Notes for Figure 5.14: Source Code Name – “res_comp_sets.m”
Alpha Set Used: [20 20 20 10 2.55 10 1.36]

E. MANUAL FAULT INTRODUCTION AND DETECTION

Due to the major residual reduction characteristics of each **alpha** set, it was necessary to evaluate fault detection performance of all **alpha** sets over each data set interval. Each **alpha** set reduced fault-free maneuvering residuals to the same approximate magnitudes for each data set interval. Since each **alpha** set adequately suppressed the residuals due to maneuvering to the same level, fault amplification characteristics for each **alpha** set were compared to decide greatest performance improvement of the Kalman filter design. **Alpha** set FOUR was found to produce the greatest fault detection performance improvement of the Kalman filter design over every data set interval. This decision was based upon **alpha** FOUR’s ability to magnify fault residuals to a greater degree than any other **alpha** set. All further evaluations of fault

detection performance by the Kalman filter design will include **alpha FOUR**'s component values within the filter.

A full-stroke fin fault was introduced into the optimized Kalman filter design in order to evaluate fault detection capabilities of the new design. The 0.4 radian fault was introduced at the *beginning* of each data set interval. Each data set interval includes a particular maneuver by the 21UUV. Prominent and quick fault declaration must occur if this optimized design is to be utilized for further implementation into UUV and AUV technology. In order to judge if a fault is present within each data set interval, the normalized relative error is plotted against time. Generated relative errors without a fin fault are included in the plot to judge the severity of the magnitude for the fault-generated relative error. Finally, a constant line for the maximum fault-free normalized relative error in that data interval is plotted throughout the data set interval in order to compare how far displaced the fault relative errors are from the fault-free relative errors. This graphical displacement gives an indication as to how robust and reliable the new design will be for detecting a full-stroke fin fault in the steering subsystem. The greater the disparity between the fault-free error line and the fault error response, the greater the robustness of the fault detector. Figures 5.15 through 5.19 show the fault detection response of the optimized Kalman filter using component values from **alpha FOUR**.

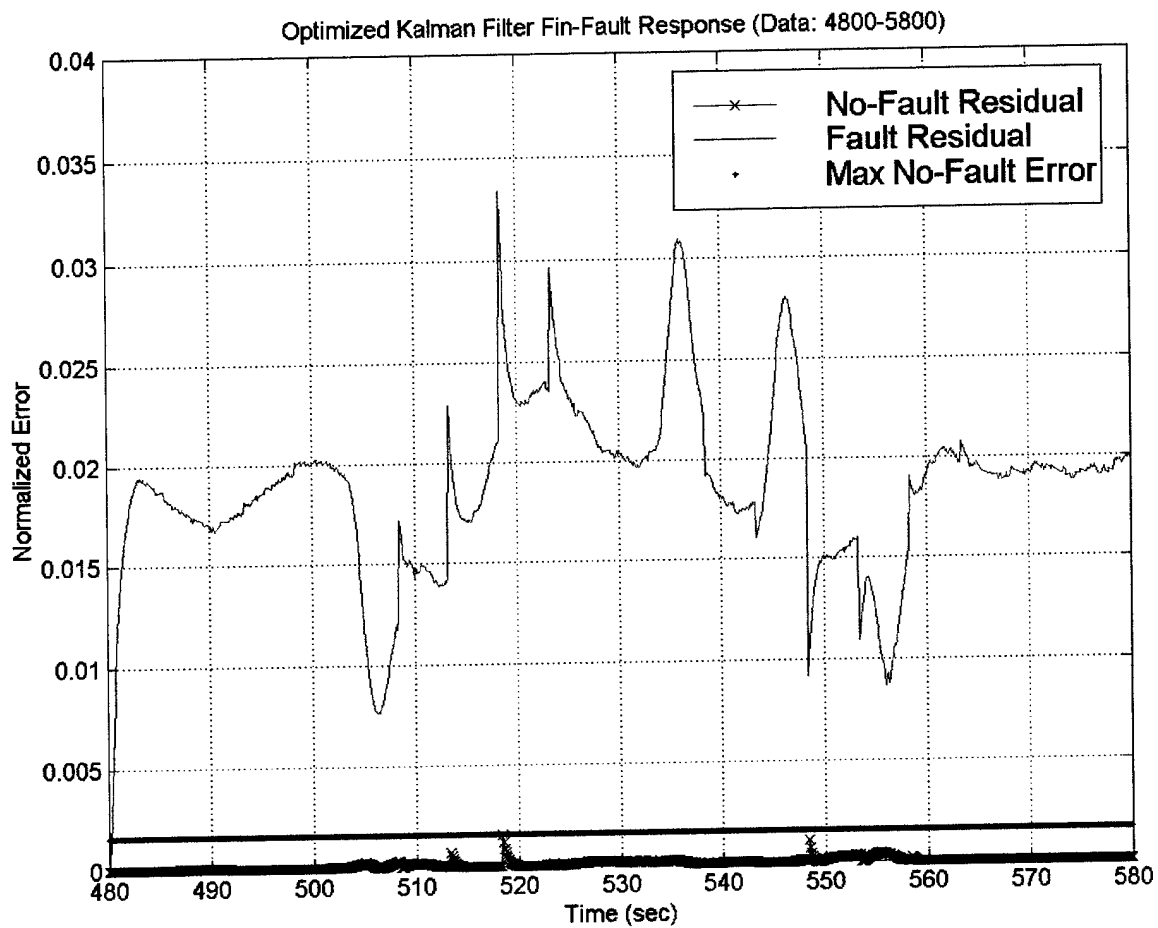


Figure 5.15 Fin Fault Detection by Optimized Kalman Design (Data: 4800-5800)

Note for Figure 5.15: Source Code Name – “kalm_faults.m”

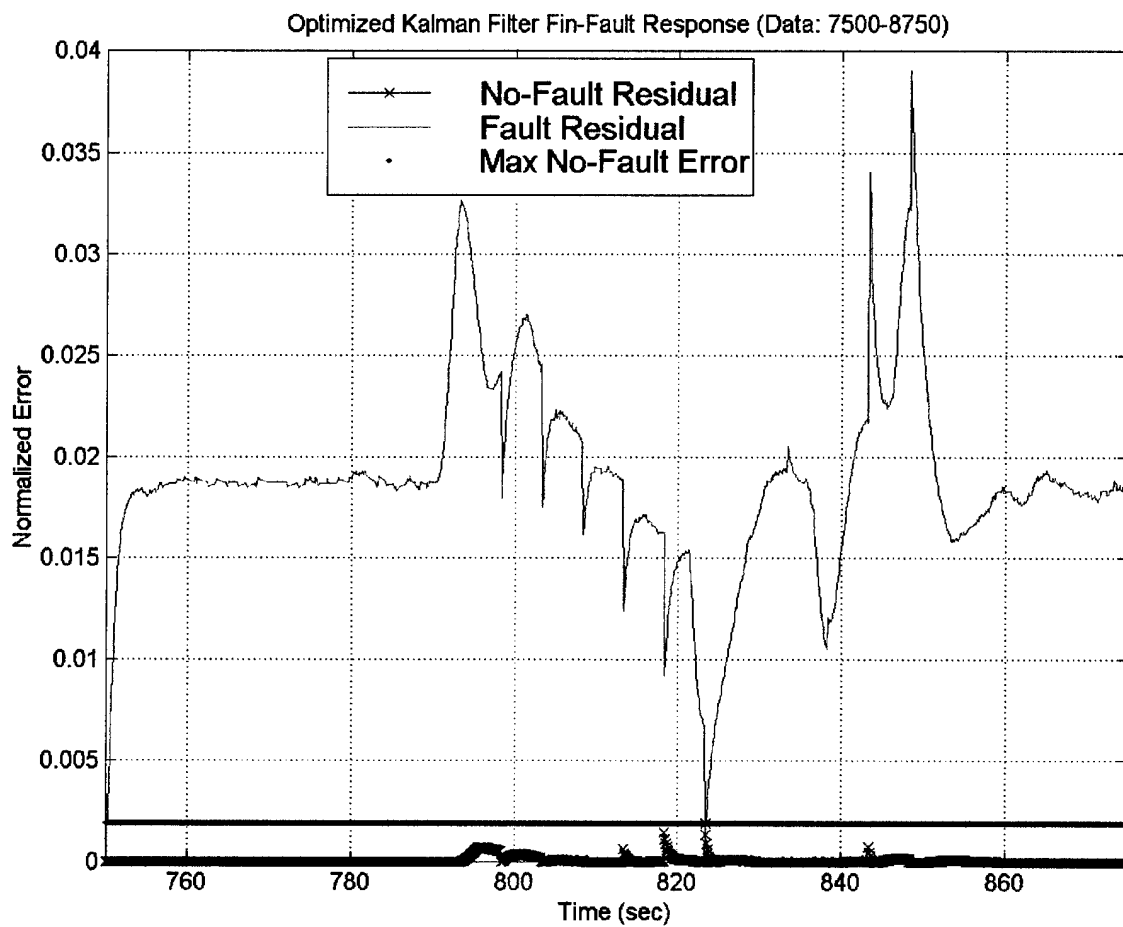


Figure 5.16 Fin Fault Detection by Optimized Kalman Design (Data: 7500-8750)

Note for Figure 5.16: Source Code Name – “kalm_faults.m”

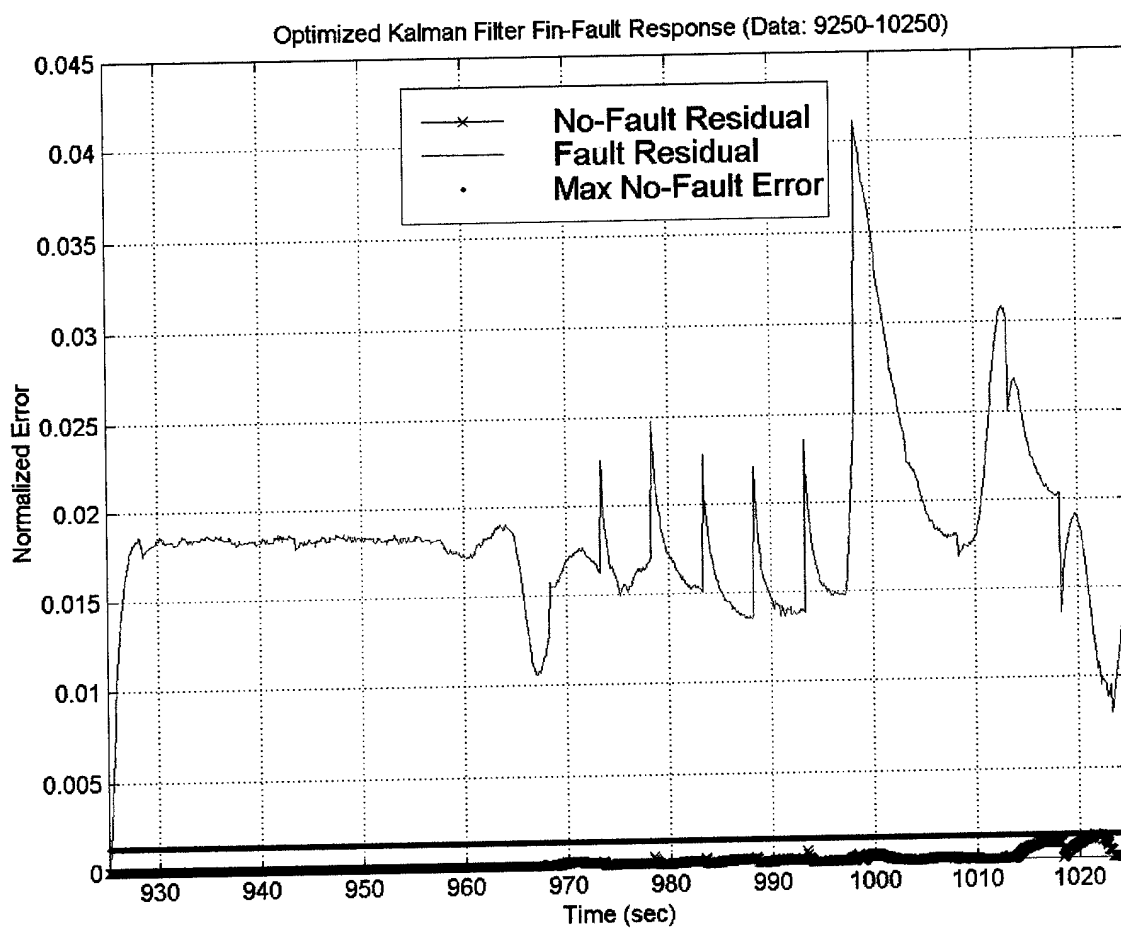


Figure 5.17 Fin Fault Detection by Optimized Kalman Design (Data: 9250-10250)

Note for Figure 5.17: Source Code Name – “kalm_faults.m”

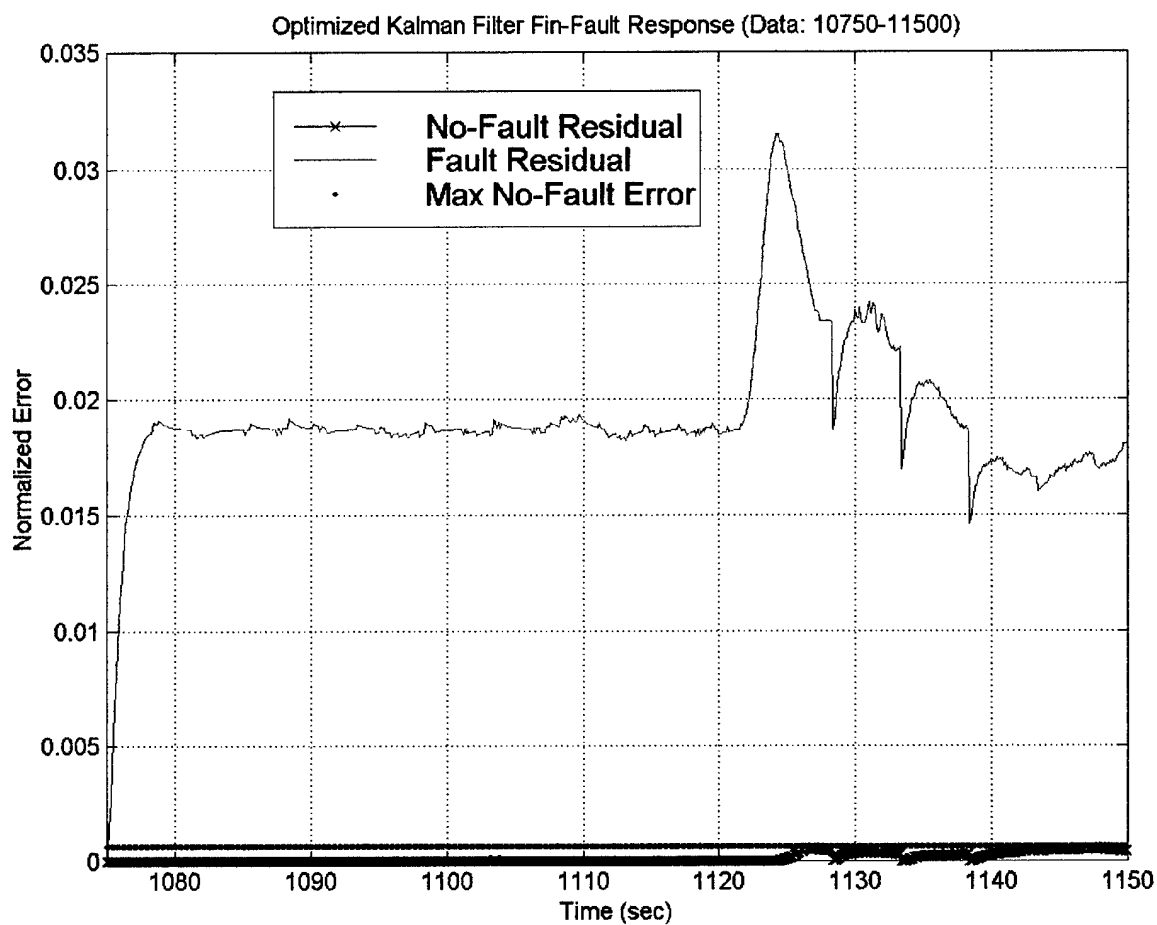


Figure 5.18 Fin Fault Detection by Optimized Kalman Design (Data: 10750-11500)

Note for Figure 5.18: Source Code Name – “kalm_faults.m”

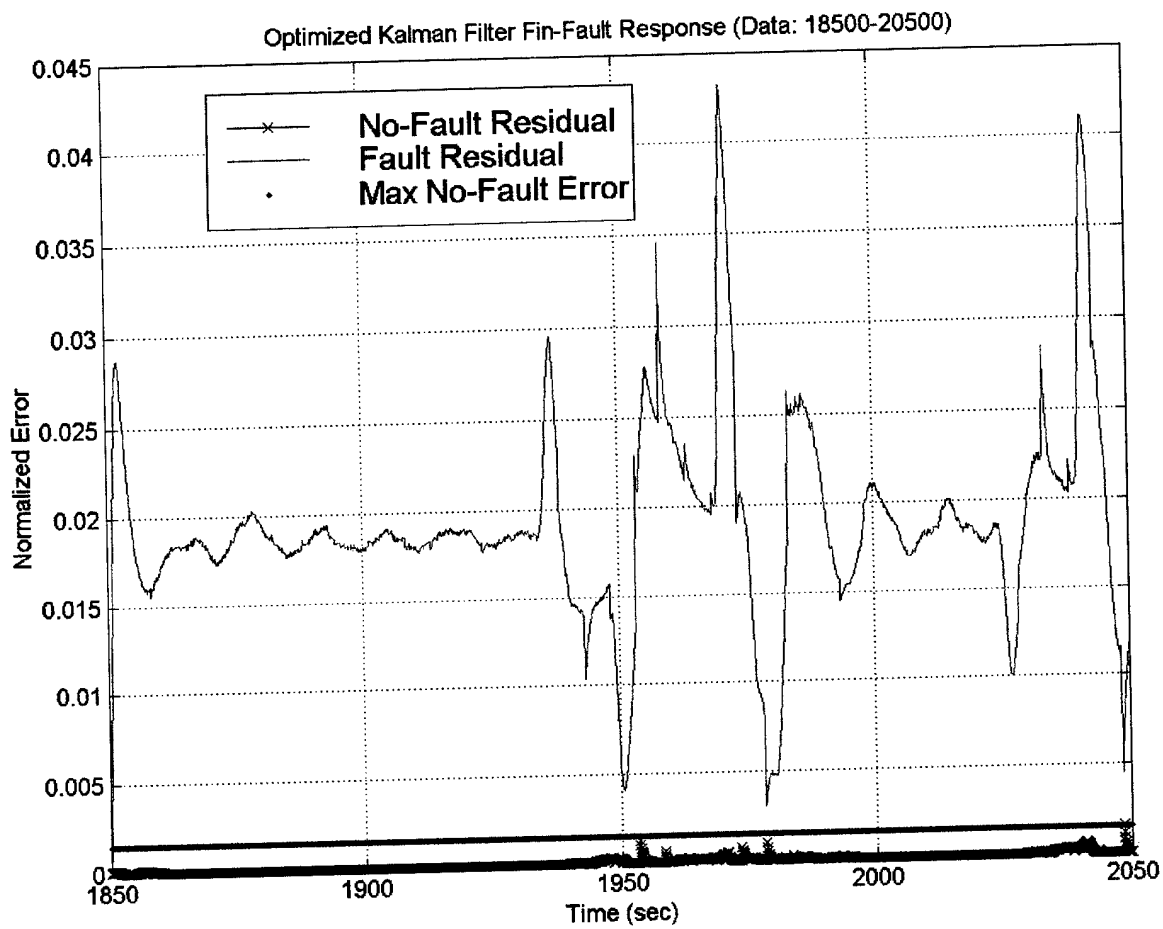


Figure 5.19 Fin Fault Detection by Optimized Kalman Design (Data: 18500-20500)

Note for Figure 5.19: Source Code Name – “kalm_faults.m”

Figures 5.15 through 5.19 display five plots that accurately and easily depict a 0.4-radian fin fault by use of an optimized Kalman filter. Each data interval contained a clearly detectable and resolute full-stroke fin fault. The enormous disparity between the maximum fault-free normalized error and the fault-normalized error provides a large degree of robustness for the detection of a full-stroke fin fault. By optimizing the $\mathbf{Q}$, $\mathbf{R}$, and β of the Kalman filter and by utilizing the previously optimized hydrodynamic coefficients of the model-based observer, the 21UUV's maneuvering error was suppressed to near zero values. Although the error due to maneuvering was driven towards zero, the optimization about the performance index of the Kalman filter resulted in design amplification of a system fault. The final values for $\mathbf{Q}$, $\mathbf{R}$, and β that have provided a useable fault detection algorithm are:

$$\mathbf{R} = \begin{bmatrix} 20 & 0 & 0 \\ 0 & 6.0862 & 0 \\ 0 & 0 & 6.4989 \end{bmatrix},$$

$$\mathbf{Q} = \begin{bmatrix} 0.478 & 1.570 & 0 \\ 1.570 & 94.70 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\beta = 1.5968.$$

The clarity of the 0.4-radian fin fault in Figures 5.15 through 5.19 indicates that a fault of lesser value may be distinguishable during vehicle maneuvering. The resolution of lesser values of fault will be discussed in Chapter VI.

F. CONCLUSIONS

Chapter V introduced the idea of improving residual reduction and fault resolution characteristics of the model-based observer design by implementation of a Kalman filter into the design's algorithm. The Kalman filter is a superior algorithm that improves prediction accuracy of state variables by filtering out inherent noise in the system. The utilization of a basic Kalman filter was improved by the introduction of the optimized hydrodynamic coefficients that were found to more accurately represent the dynamic characteristics of a maneuvering 21UUV. Seven values of the Kalman filter algorithm were identified as having the potential of improving filter performance by reducing maneuvering residual values. These seven values originated from the system noise matrix, Q , the measurement noise matrix, R , and the implementation of a scalar gain multiplier, β . Optimization of these seven values over each maneuvering data set produced values that greatly increased residual reduction during maneuvers. The optimization of Q and R produced matrixes that accurately modeled the noise within the system. In order to ensure proper amplification of faults within the system, a performance index, consisting of a normalized relative error value for the state residuals, was designed and utilized as the objective function for optimization. The optimization of this objective function produced optimized values for each data set that reduced maneuvering error to near zero values and significantly amplified errors due to system faults. The optimized values associated with data set interval FOUR were chosen to be used for the optimized Kalman filter due to their superior fault amplification characteristics. Figures 5.15 through 5.19 graphically display the phenomenal fault

detection capability of an optimized Kalman filter design that utilizes the optimized hydrodynamic coefficients of the model-based observer.

The utilization of this design into the 21UUV will provide the vehicle with a robust and reliable fault detection algorithm that is capable of eliminating residual errors due to maneuvering, while having the capability to amplify and detect errors due to system faults.

THIS PAGE LEFT INTENTIONALLY BLANK

VI. ANALYSIS OF FAULT RESOLUTION OF FINAL DESIGN

A. DEVELOPMENT OF A FAULT DETECTION ALGORITHM

In order to simulate the fault detection capability of the optimized Kalman filter design, a fundamental fault detection algorithm was written to identify faults within the normalized relative error value. Two methods for detecting a fault were included in the algorithm. The first method for detecting a fault simply entailed the classification of a fault if the normalized error exceeded a peak threshold value. The peak threshold was chosen through the evaluation of the maximum fault-free normalized errors developed by the vehicle throughout each maneuvering interval. The largest error was isolated and chosen as the threshold value. This peak threshold value was set at 0.0016. The second method for detecting fault errors included the interrogation of past error values after a lower threshold error value was exceeded by the system. Once the lower level is exceeded, the algorithm recalls four previous error values and evaluates their magnitudes. If the four previous errors had magnitudes greater than the lower threshold value, then a fault is said to exist. The past time history length evaluated by the algorithm depends on the sensitivity requirements of the vehicle operator. A large time history length minimizes false-detects within the system by overlooking anomalous error spikes that may be developed within the measurement data. Although the minimization of the probability of false-detects is beneficial to unintentional mission aborts and unnecessary system reconfigurations, large time histories also lengthen the time between the initiation of an actual fault and the positive detection of the fault. Conversely, if the time history length is chosen too small, then the algorithm quickly detects faults, but may mistake

anomalous error spikes as actual faults. This scenario increases the likelihood of mission aborts and unnecessary system reconfigurations. The trade-offs between the two extremes of fault detection must be properly weighted by the operator in order for a correct time history length to be chosen. After many evaluations of the sensitivities of differing time history lengths, the final time history chosen for fault detection evaluation of the 21UUV was 20 data steps, or 2.0 seconds. Unless the normalized errors due to a fault exceed the peak threshold value under two seconds, the time to detection of a fault will be in excess of 2.0 seconds. Optimally, the length of time between fault initiation and fault detection is approximately 5.0 seconds. Hopefully, fault detection under five seconds will allow the vehicle time enough to mitigate the fault or receive instructions from higher authority prior to vehicle endangerment or loss. Obviously, the time allotted to correct or mitigate a fault is dependent on the actual fault itself. The occurrence of a fin failure during deep-water operations is much less dangerous than losing a fin while the vehicle conducts shallow water operations!

B. ANALYSIS OF FAULT DETECTION FOR EACH DATA INTERVAL

The algorithm developed for detecting a full-stroke fin fault was evaluated over each data set interval. Since each interval contained maneuvering data, it was important to determine whether the optimized Kalman design was capable of detecting faults during non-steady state conditions. The final Kalman filter design of Chapter V will be utilized here for fault detection and resolution evaluation. The 0.4-radian fin fault was introduced at the *beginning* of each data set interval. Figures 6.1 through 6.5 display the resultant fault detection performance of the optimized Kalman design. As seen in these figures,

detection of a full-stroke fin fault is very reliable and quick. The faults for these simulated fin failures were detected within 0.5 seconds of their initiation into the steering subsystem. The coupling of the optimized Kalman filter with the fault detection algorithm produced a successful fault detection system that is robustly capable of detecting the full-stroke fin fault that wasn't detectable using prior model-based observer designs.

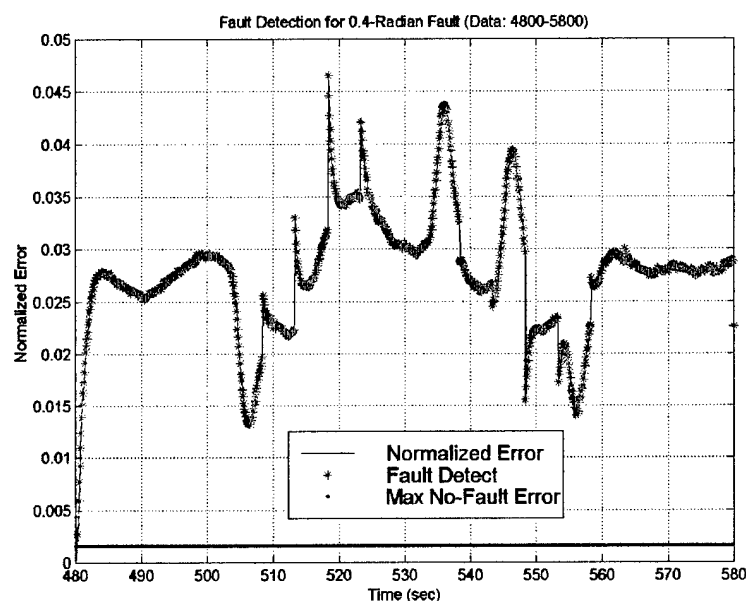


Figure 6.1 0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 4800-5800)

Notes on Figure 6.1: Source Code Name – "fault_test.m"

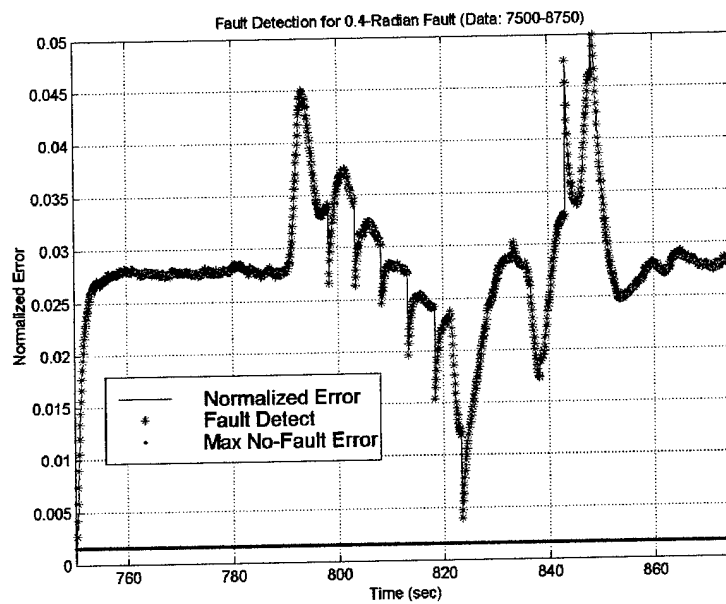


Figure 6.2 0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 7500-8750)

Notes on Figure 6.2: Source Code Name – “fault_test.m”

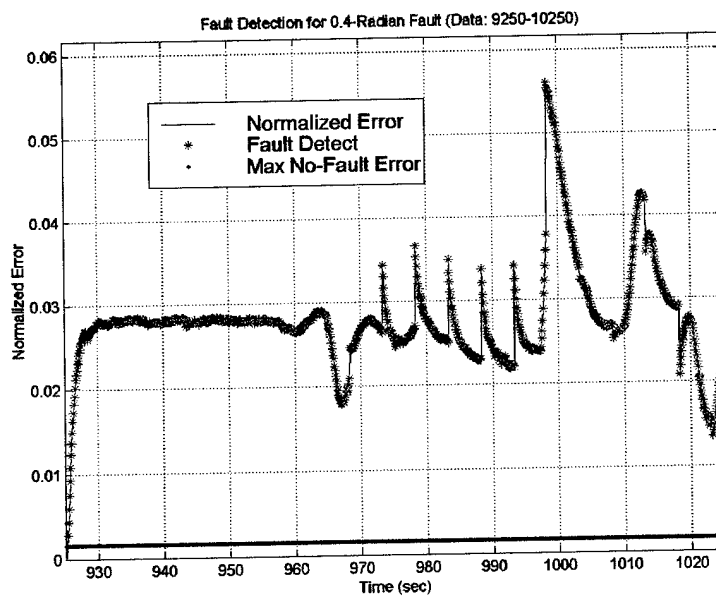


Figure 6.3 0.4-Radian Fault Detection w/Optimized Kalman Design (Data: 9250-10250)

Notes on Figure 6.3: Source Code Name – “fault_test.m”

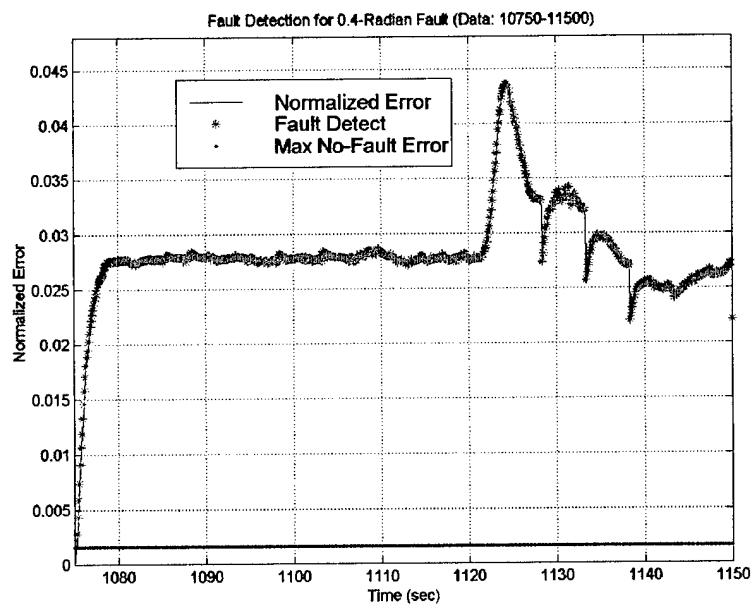


Figure 6.4 0.4-Radian Fault Detection w/Optimized Kalman Design
(Data: 10750-11500)

Notes on Figure 6.4: Source Code Name – “fault_test.m”

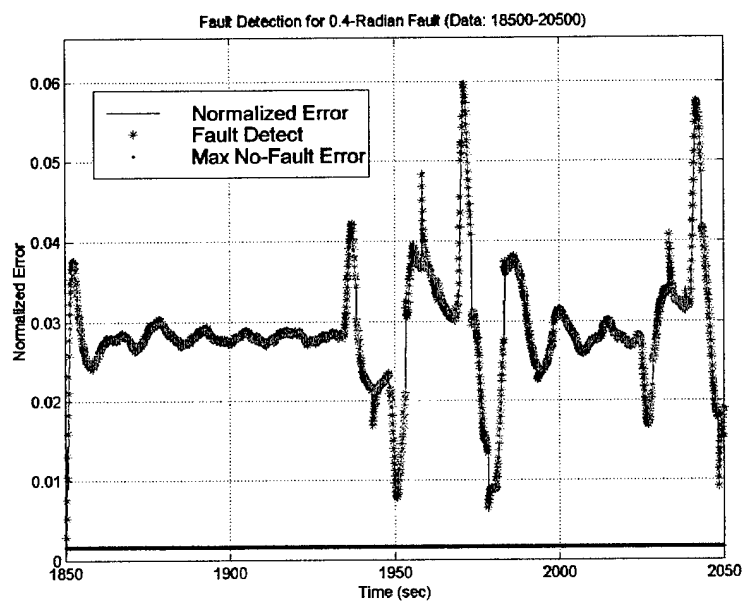


Figure 6.5 0.4-Radian Fault Detection w/Optimized Kalman Design
(Data: 18500-20500)

Notes of Figure 6.5: Source Code Name – “fault_test.m”

C. FAULT MAGNITUDE SENSITIVITY FOR DETECTION

The clear indication of a fault in the previous five figures leaves little doubt as to the ability of this work's final fault detection design to detect a full-stroke fin fault. Although the ability to detect a fault of this magnitude was the motivation for this work, it is necessary to evaluate the sensitivity of the fault detection design to lesser magnitude fin faults. A full-stroke fin fault would be an 'ideal' failure for a fault detection design due to the large amount of normalized relative error produced within the steering subsystem. A more probabilistic scenario for a fin failure would be for failure of the fin between the maximum and minimum range of its stroke. The final fault detection design of this work was evaluated over the range of stroke deflections for a 21UUV fin. The data set interval used to analyze the sensitivity of the fault detection design was taken from interval FOUR. Interval FOUR includes the largest and most dynamic maneuver taken by the 21UUV during its 21900 data point run. The graph of the track for this interval is reproduced here from Chapter III in order to display the unusual maneuvering aspects of this data interval set. This maneuver is a good representation of a highly dynamic maneuver where generation of state residuals and subsequent normalized errors would be very abundant. The analysis of the fault detection design's sensitivity to lesser stroke fin faults over this data set gives an accurate representation of the sensitivity of the design throughout all maneuvers.

The peak threshold value for the fault detection design remained at 0.0016 and the lower value for time history interrogation was set at .00005. These values for threshold levels produced very good results for fault detection sensitivity of the final design.

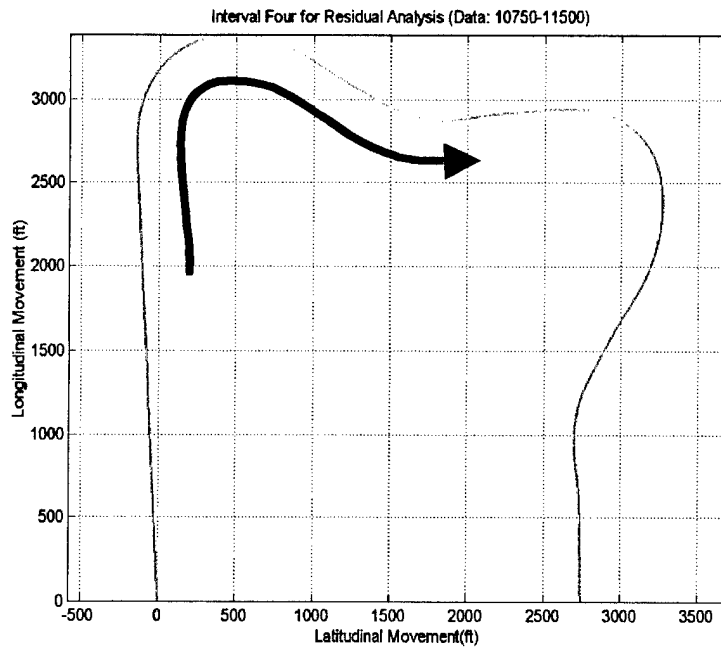


Figure 3.10 Data Set Interval Four

Note on Figure 3.10: Source Code Name – “dead_int4.m”

In order for the fault detection design to accurately detect a fault, the fault must be maintained through a majority of the data interval. Scattered fault-detects at low stroke magnitudes were not associated with actual faults. In actuality, the lowest error value for a fault-detect would be 0.0005, since this value is defined as the lower threshold value for fault detection.

Figures 6.6, 6.7, and 6.8 display the fault sensitivity of the final design to faults of magnitude 0.2-radian, 0.1-radian, and 0.06-radian, respectively. The final value of a fin stroke that was detected was 0.055-radian. This would relate to a failed fin at an angle of 3.15° off centerline. The final sensitivity of the fault detection design exceeds all initial criteria for failure detection. In essence, the final fault detection design developed by this work performs better than originally required or desired.

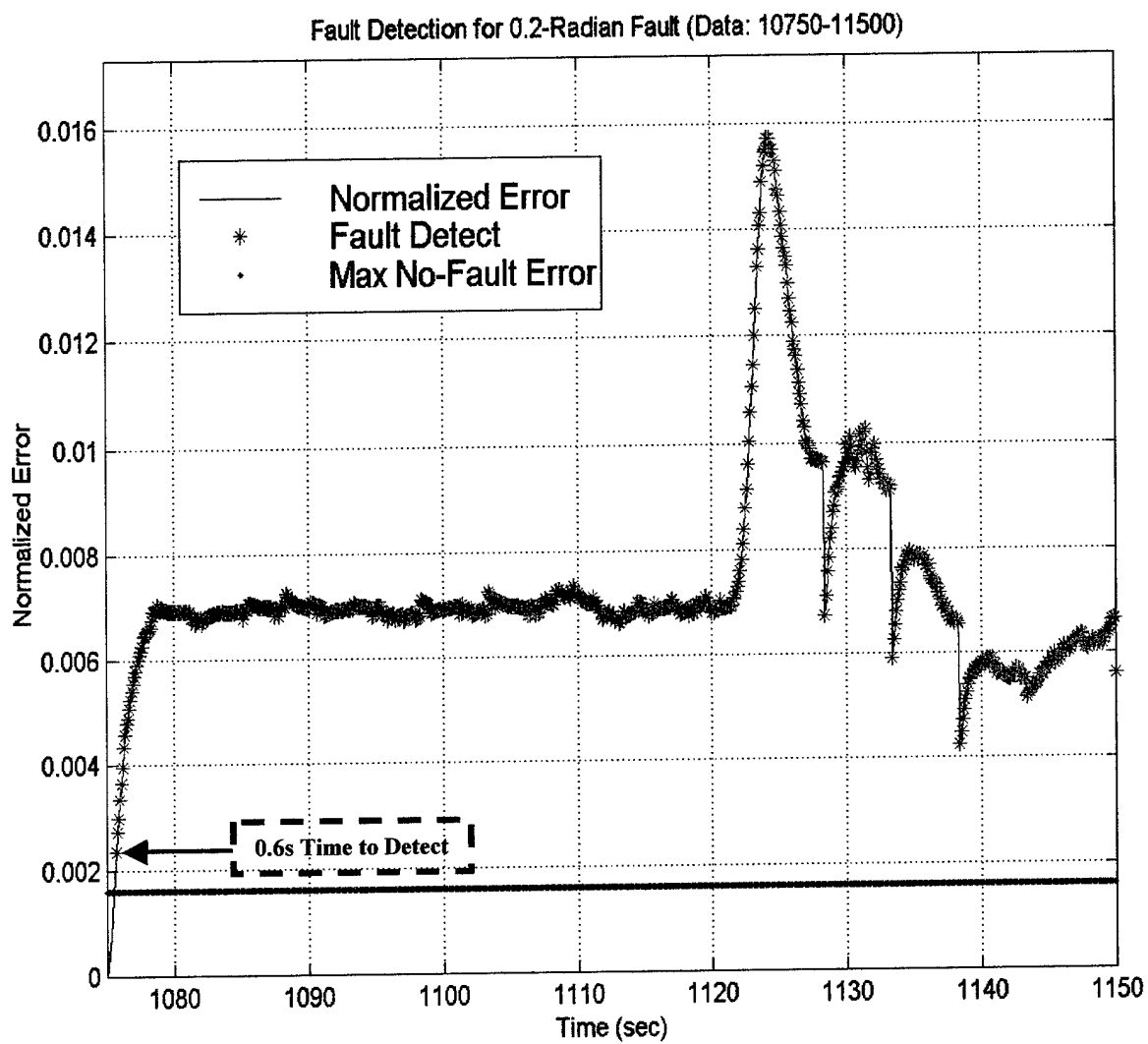


Figure 6.6 Fault Sensitivity to 0.2-radian Fault (Data: 10750-11500)

Note for Figure 6.6: Source Code Name – “fault_test.m”

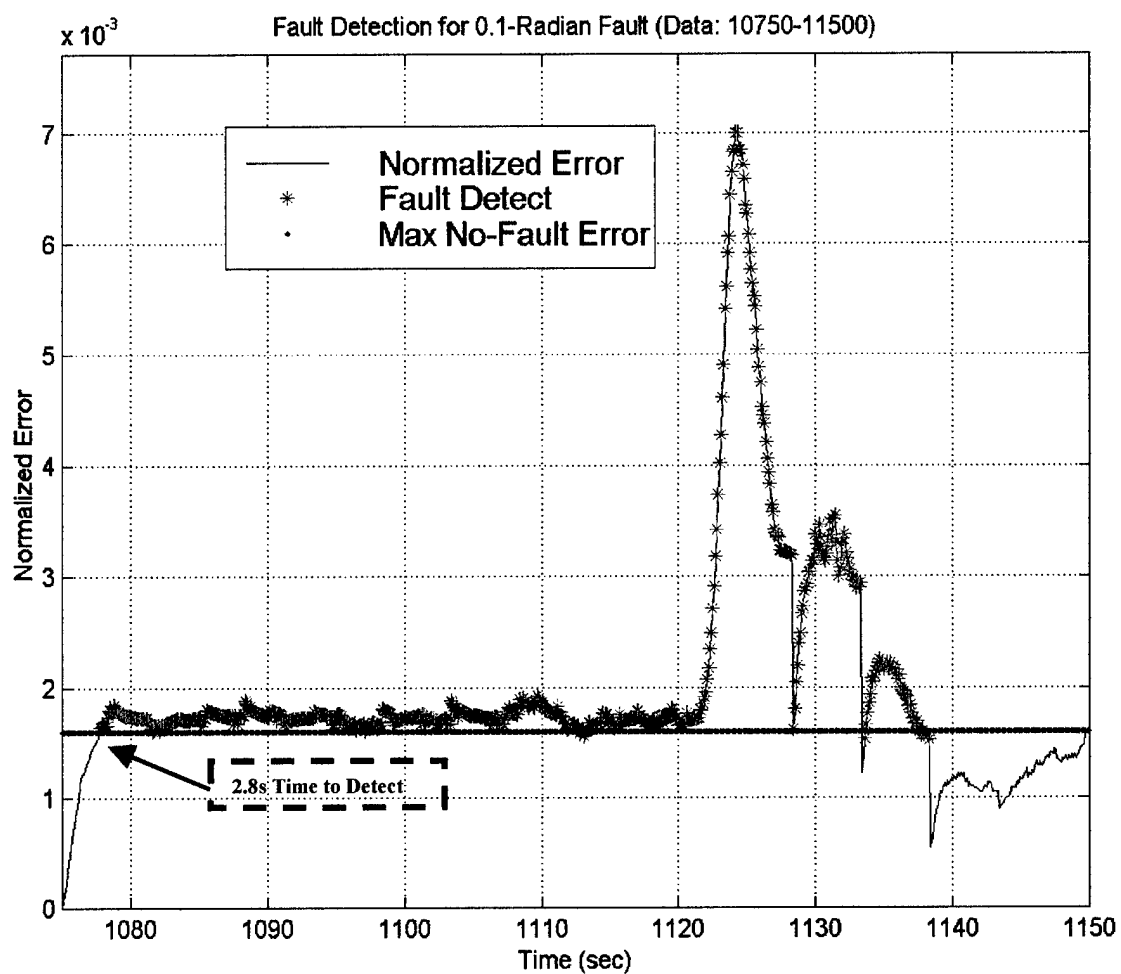


Figure 6.7 Fault Sensitivity to 0.1-radian Fault (Data: 10750-11500)

Note for Figure 6.7: Source Code Name – “fault_test.m”

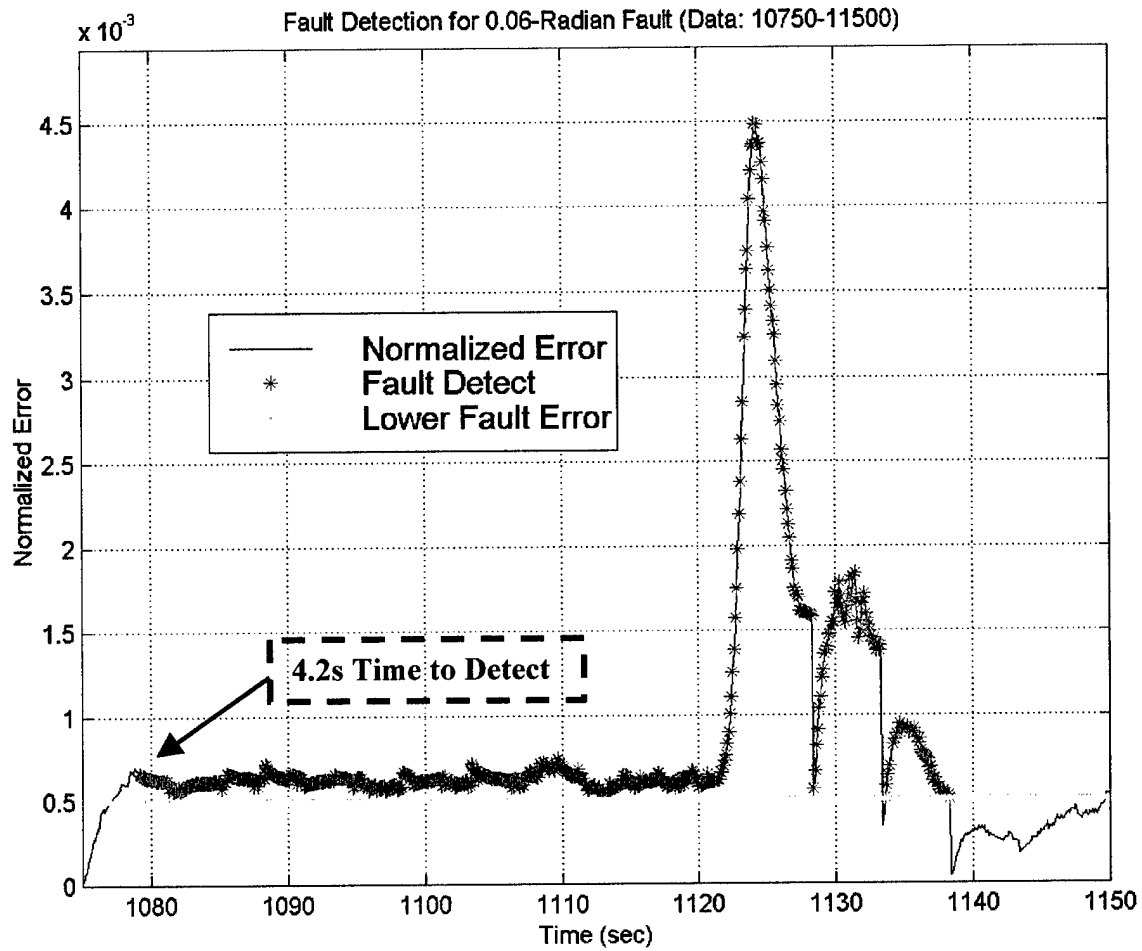


Figure 6.8 Fault Sensitivity to 0.06-radian Fault (Data: 10750-11500)

Note for Figure 6.8: Source Code Name – “fault_test.m”

Figures 6.6 and 6.7 display fault detections that occurred due to normalized error exceeding the peak threshold value. As seen by these two plots, detection of the fault occurs very quickly. In comparison, Figure 6.8 displays a fault detected by time history errors exceeding the lower threshold value. As can be seen from this plot, the time to detect the fault increased due to the time length input into the time history portion of the fault detection algorithm. A plot of fault magnitude versus the time to fault detection is included in Figure 6.9. As annotated on the plot, 'time to detect' increases at an exponential rate for decreasing fault magnitude. Fault magnitude values greater than 0.22-radian reach a constant 'time to detect' of 0.5 seconds.

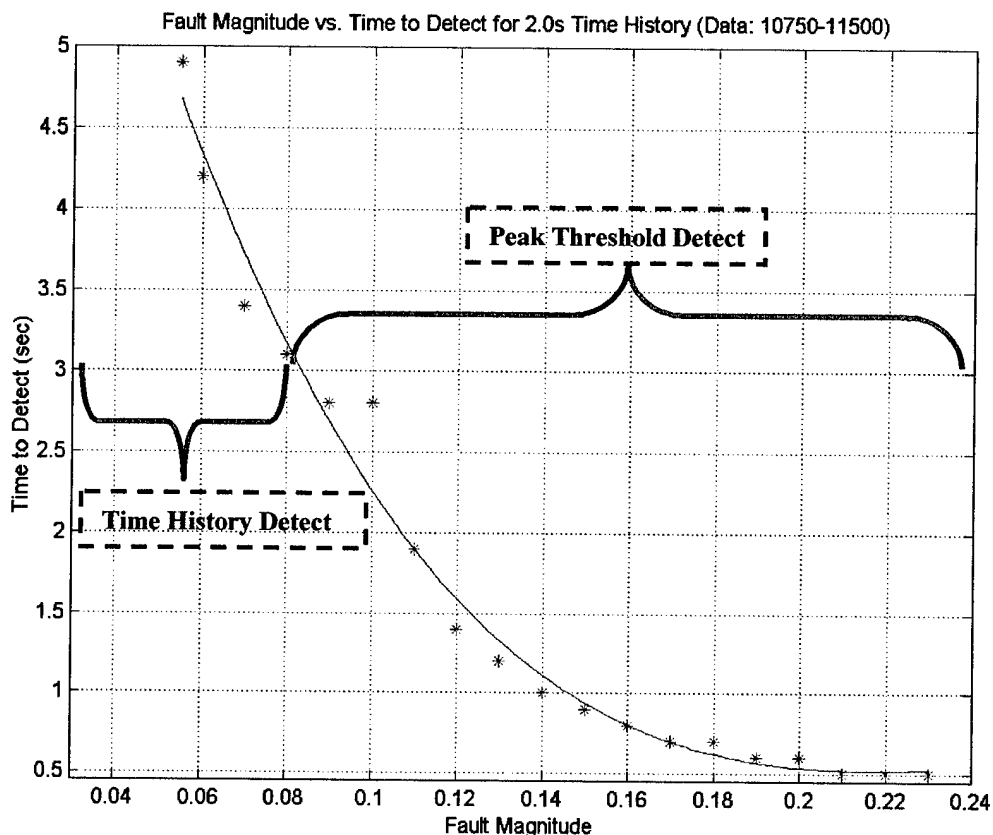


Figure 6.9 Fault Magnitude vs. Time to Detect

Note for Figure 6.9: Source Code Name – “sense.m”

D. METHODOLOGY FOR FAULT DETECTION DESIGN TAILORING

The procedure to obtain the final fault detection design of this work can be adapted to fit other UUV's or AUV's. The fault detection design developed in this work is specifically oriented towards the characteristics and individuality of the 21UUV. Adaptation and subsequent implementation of a vehicle-specific fault detection design may be developed by following the basic procedures as outlined in Figure 6.10. This figure gives a broad procedural guideline to follow if a fault detection design of this type is to be adapted to another autonomous or unmanned underwater vehicle. If greater clarification of developmental procedures is necessary, one can refer back to this work in order to obtain all manners of fault detection specifics pertaining to this design.

E. CONCLUSIONS

A fault detection algorithm was developed that utilized peak and time history error interrogation in order to determine if a fault existed in the 21UUV steering subsystem. The sensitivity of the final fault detection design of this work was found to be as low as .055-radians for a fin fault. The sensitivity range of this fault detection design enables the 21UUV to detect a fin fault with a minimum stroke angle of 3.15° to the maximum stroke range of the fin, which is 23.0° . The time to detect the smallest detectable fin fault was 4.9 seconds. The minimum time to detect a fault was 0.5 seconds and this occurred for faults of magnitudes greater than 0.22 radians.

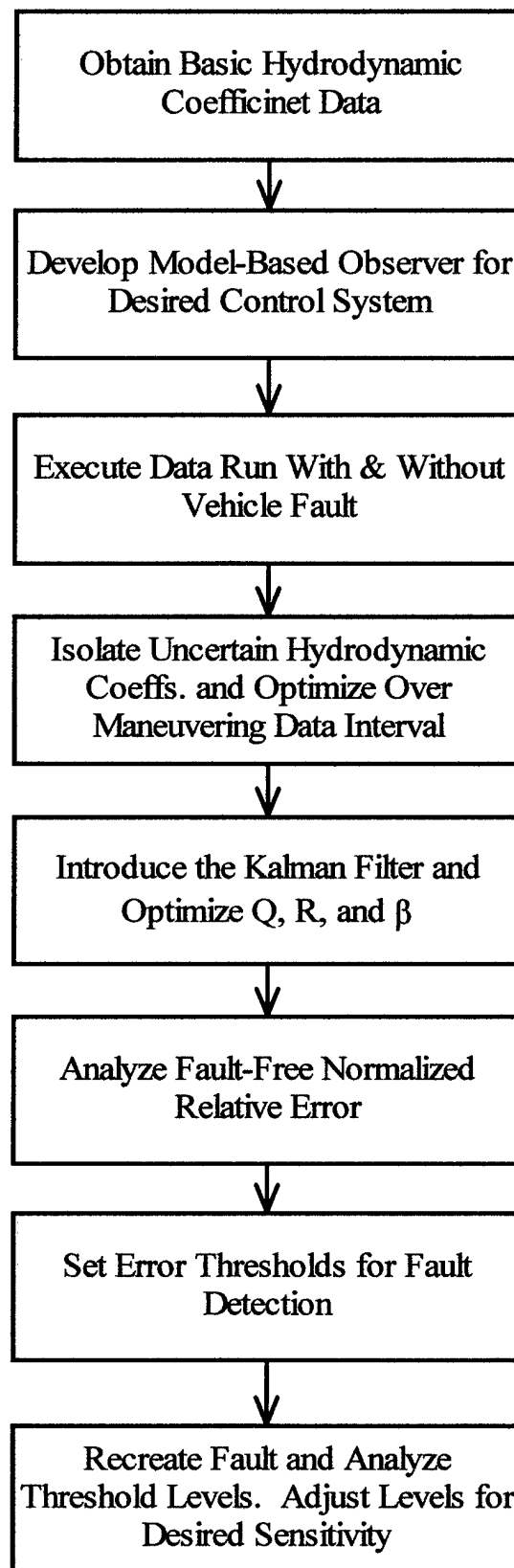


Figure 6.10 Individual Vehicle Adaptation Procedure

The methodology used in this work can be adapted and implemented into other various unmanned or autonomous underwater vehicles. A basic procedural outline was provided which described the steps necessary to develop a similar fault detection system for another vehicle of choice.

VII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

As with all autonomous systems or machines tasked with carrying out complex mission assignments in extreme environments, AUV's may experience unforeseen problems. These threaten the mission reliability and completeness of operational goals. To maximize the possibility of mission completeness, AUV control systems are being developed with the capability of detecting a variety of failures within their subsystems and autonomously correcting for such failures.

This thesis has examined the design of a model-based observer and an extended Kalman filter for detecting faults in the steering subsystem of the 21UUV. The utilization of real sensor measurement data from the 21UUV has enabled the filters to be accurately tuned for the characteristics of this particular vehicle. Specifically, the problem of fault detection in the presence of vehicle maneuvers has been studied in depth. This work has shown that optimization of the filter design has allowed for fault induced residuals to be distinguished from residuals induced by maneuvering alone. Further detailed conclusions for the development and optimization of filter design are contained at the end of each respective chapter.

B. RECOMMENDATIONS

The further recommended work on the fault detection design produced in this work is substantial. The methodology used in the development of the final fault detection algorithm should be applied to the diving control and the roll control subsystems of the

21UUV. Adaptation of this methodology for use in another control subsystem can be easily accomplished by utilization and reconfiguration of the *Matlab* code developed for this work, which is included in Appendices A, B, and C.

Also of importance to the further improvement of fault detection in unmanned or autonomous underwater vehicles is the ability to process sensor measurement data taken from a vehicle with an actual fault within its steering subsystem. By having this data to use for the detection of actual faults, the fault detection design developed in this work can be adjusted to reliably detect an actual fault within the system.

APPENDIX A. *MATLAB* CODE ASSOICIATED WITH ORIGINAL MODEL-BASED OBSERVER DESIGN

The *Matlab* code associated and developed for the original model-based observer is contained on CD-ROM and is obtainable through request from Professor A. J. Healey.

Appendix A is contained on the CD-ROM and can be found under the directory, assuming D is the letter representing the CD-ROM drive, D:\Gibbons_thesis\app_A.

The files contained in Appendix A and within this directory are as follows:

- dead.m
- dead_int1.m
- dead_int2.m
- dead_int3.m
- dead_int4.m
- dead_int5.m
- O_d1.m
- O_d1_faults.m
- rap_count.m
- state_resp.m

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. *MATLAB* CODE ASSOCIATED WITH OPTIMIZATION OF MODEL-BASED OBSERVER DESIGN

The *Matlab* code associated and developed for the optimization of the hydrodynamic coefficients of the model-based observer design is contained on CD-ROM and is obtainable through request from Professor A. J. Healey. Appendix B is contained on the CD-ROM and can be found under the directory, assuming D is the letter representing the CD-ROM drive, D:\Gibbons_thesis\app_B. The files contained in Appendix B and within this directory are as follows:

- final_perf.m
- opt_faults.m
- opt_res_reduc.m
- RMS_obs
- side_perf_sets.m

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. *MATLAB* CODE ASSOCIATED WITH OPTIMIZED KALMAN FILTER DESIGN

The *Matlab* code associated and developed for optimization of the extended Kalman filter design is contained on CD-ROM and is obtainable through request from Professor A. J. Healey. Also included in this appendix is the code developed for the fault detection algorithm used in the final design. Appendix C is contained on the CD-ROM and can be found under the directory, assuming D is the letter representing the CD-ROM drive, D:\Gibbons_thesis\app_C. The files contained in Appendix C and within this directory are as follows:

- f_calc.m
- f_calc_orig.m
- fault_test.m
- kalm_faults.m
- opti_call.m
- opti_kalm_sets.m
- res_comp_sets.m
- res_test.m
- res_test.both.m
- RMS_test_sets.m
- sense.m

The necessity to place Appendices A, B, and C on CD-ROM arises from the large amount of code written for this work. By using a CD-ROM, anyone may access the files developed for this work and alter them as need be to further the research in this area. Also included on the CD-ROM is this entire text, written in MICROSOFT WORD 2000.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

- An, E.P., 1998, Proceedings IEEE AUV98 Workshop on Underwater Navigation, Draper Laboratories, Cambridge, Massachusetts, Aug 21-22.
- Bar-Shalom, Y. and Li, X., 1993, *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Boston, Massachusetts.
- Bell, D. et al., 1992, "Using Causal Reasoning for Automated Failure Modes & Effects Analysis (FMEA)," Proceedings of the Annual Reliability and Maintainability Symposium, Las Vegas, Nevada, pp. 343-353.
- Bellingham, J.G., 1997, "Sampling Strategies for Ocean Observation with Mobile Platforms," Marine Society Journal, Vol. 31, No. 3, pp. 34-37.
- Branch, M.A. and Grace, A., 1996, *Optimization Toolbox*, The Math Works, Inc.
- Gelb, A., 1974, "Applied Optimal Estimation," Cambridge, Massachusetts, September.
- Healey, A.J., 1998, "Analytical Redundancy and Fuzzy Inference in AUV Fault Detection and Compensation," Naval Postgraduate School, Monterey, California.
- Healey, A.J., 1995, Course Notes, Dynamics of Marine Vehicles, Naval Postgraduate School, Monterey, California.
- Healey, A.J., 1994, "Coordination of the Hovering Behaviors of the NPS AUV II Using Onboard Sonar Servoing," MBARI NSF Workshop, April.
- Healey, A.J., 1993, "Towards and Automatic Health Monitor for Autonomous Underwater Vehicles Using Parameter Identification", Naval Postgraduate School, Monterey, California.
- Healey, A.J. and Lienard, D., 1993, "Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles," IEEE Journal of Oceanic Engineering, Vol. 18, No. 3, pp. 327-339.
- Healey, A.J., 1992, "A Neural Network Approach to Failure Diagnostics for Underwater Vehicles," Proceedings of 1992 IEEE Conference, Washington, D.C., June, pp. 131-134.
- Hurni, M.A., 1997, "Development of an On-Line Failure Mode Detection and Resolution Algorithm for the Phoenix AUV," Master's Thesis, Naval Postgraduate School, Monterey, California.
- Isermann, R., 1984, "Process Fault Detection Based on Modeling and Estimation Methods – A Survey," Automatica, Vol. 20, No. 4, pp. 387-404.

- Mangoubi, R., 1998, *Robust Estimation and Failure Detection: a Concise Treatment*, New York, Springer.
- Mangoubi, R.S., Appleby, B.D., Vergese, G.C., and Vandervelde, W.E., 1995, "A Robust Failure Detection and Isolation Algorithm," Proceedings of 34th Conference on Decision and Control, New Orleans, Louisiana, December, pp. 2377-2382.
- Marco, D.B., 1996, "Autonomous Control of Underwater Vehicles and Local Area Maneuvering", Doctoral Dissertation, Naval Postgraduate School, Monterey, California, September.
- Marco, D.B. and Healey, A.J., 1996, "Local Area Navigation Using Sonar Feature Extraction and Model Based Predictive Control," Autonomous Underwater Vehicles Laboratory, Naval Postgraduate School, Monterey, California.
- Maybeck, P.S., 1996, "The Kalman Filter: An Introduction to Concepts", Symposium on Autonomous Underwater Vehicle Technology, Monterey, California, June.
- Melvin, J.E., 1998, "AUV Fault Detection Using Model Based Observer Residuals," Master's Thesis, Naval Postgraduate School, Monterey, California.
- Patton, R.J., 1997, "Fault Tolerant Control: Fault Detection and Compensation," ONR Workshop, Naval Postgraduate School, Monterey, California.
- Patton, R.J. and Chen, J., "Robust Fault Detection Using Eigenstructure Assistance: A Tutorial Consideration and Some New Results," Proceedings of 30th Conference on Decision and Control, pp. 2242-2247.
- Peng, et. al., 1997, "A Complete Procedure for Residual Generation and Evaluation with Application to a Heat Exchanger," IEEE Transactions on Control Systems Technology, Vol. 5, No. 7, pp. 351-370.
- Powell, M.J.D., "Variable Metric Methods for Constrained Optimization," *Mathematical Programming: The State of the Art*, (A.Bachem, M.Groetschel and B.Korte, eds.) Springer Verlag, pp.288-311.
- Powell, M.J.D., 1997, "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations, no. DAMTP77/NA, 2," University of Cambridge, England.
- Smith, M.S., 1995, "The Ocean Explorer AUV: A Modular Platform for Coastal Sensor Deployment," Autonomous Vehicles in Mine Countermeasures Symposium, Naval Postgraduate School, Monterey, California.
- Vaanderplaats, G.N., 1999, *Numerical Optimization Techniques For Engineering Design*, Vanderplaats Research & Development, Inc., Colorado Springs, Colorado.

INITIAL DISTRIBUTION LIST

	No. of Copies
1. Defense Technical Information Center..... 8725 John J. Kingman Road STE 0944 Ft. Belvoir, VA 22060-6218	2
2. Dudley Knox Library Naval Postgraduate School 411 Dyer Road Monterey, California 93943-5101	2
3. Professor Anthony J. Healey, Code ME/He..... Naval Postgraduate School 700 Dyer Road Monterey, California 93943-5101	2
4. Mechanical Engineering Department Chairman, Code ME..... Chairman, Mechanical Engineering Department Naval Postgraduate School Monterey, California 93943-5101	1
5. Engineering and Technology Curricular Officer, Code 34..... Naval Postgraduate School Monterey, California 93943-5101	1
6. LT Andrew S. Gibbons 1120 Third Street Paonia, Colorado 81428	3
7. Mr. Mike Keegan, Underwater Vehicles..... NUWC, Code 8221 Newport, Rhode Island 02841-5047	1
8. Mr. Chris Hillenbrand NUWC, Code 8291 Newport, Rhode Island 02841-5047	1
9. Dr. Ravi Mangoubi The Charles Stark Draper Laboratories Inc. 555 Technology Square, MS-77 Cambridge, Massachusetts 02139-3563	1