



Calhoun: The NPS Institutional Archive

Theses and Dissertations

Thesis Collection

2000-03

UH-60 Black Hawk disturbance rejection study for hover/low speed handling qualities criteria and turbulence modeling/ Steven J. Labows.

Labows, Steven J.

Monterey, California. Naval Postgraduate School



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>

NAVAL POSTGRADUATE SCHOOL
Monterey, California



THESIS

**UH-60 BLACK HAWK DISTURBANCE REJECTION
STUDY FOR HOVER/LOW SPEED HANDLING
QUALITIES CRITERIA AND TURBULENCE MODELING**

by

Steven J. Labows

March 2000

Thesis Co-Advisors:

Mark B. Tischler
E. Roberts Wood

Approved for public release; distribution is unlimited.

DTIC QUALITY INSPECTED 4

20000615 022

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 UH-60 Black Hawk Disturbance Rejection Study For Hover/Low Speed Handling Qualities Criteria And Turbulence Modeling			5. FUNDING NUMBERS	
6. AUTHOR(S) Labows, Steven J.				
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) U.S. Army/NASA Rotorcraft Division, Ames Research Center, Moffett Field, CA 94035-1000			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) <p>Helicopters operate in an environment where task performance can easily be affected by atmospheric turbulence. This paper discusses the airborne flight test of the Sikorsky UH-60 Black Hawk helicopter in turbulent conditions to determine disturbance rejection criteria and to develop a low speed turbulence model for helicopter simulation. A simple approach to modeling the aircraft response to turbulence is described by using an identified model of the Black Hawk to extract representative control inputs that replicate the aircraft response to disturbances. This parametric turbulence model is designed to be scaled for varying levels of turbulence and utilized in ground or in-flight simulation. Flight control cutoff frequency data are also analyzed to support design criteria for gust rejection handling qualities.</p>				
14. SUBJECT TERMS Low Speed Helicopter Handling Qualities, Turbulence Modeling, Cutoff Frequency, Flight Control Design Requirements.			15. NUMBER OF PAGES 123	
			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

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**UH-60 BLACK HAWK DISTURBANCE REJECTION STUDY FOR
HOVER/LOW SPEED HANDLING QUALITIES CRITERIA AND
TURBULENCE MODELING**

Steven J. Labows
Lieutenant Commander, United States Navy
B.S., Cornell University, 1988


Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the


**NAVAL POSTGRADUATE SCHOOL
March 2000**

Author:




Steven J. Labows

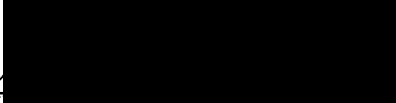
Approved by:




Mark B. Tischler, Thesis Co-Advisor



E. Roberts Wood, Thesis Co-Advisor



Chris L. Blanken, Second Reader



Max F. Platzer, Chairman
Department of Aeronautics and Astronautics

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Helicopters operate in an environment where task performance can easily be affected by atmospheric turbulence. This paper discusses the airborne flight test of the Sikorsky UH-60 Black Hawk helicopter in turbulent conditions to determine disturbance rejection criteria and to develop a low speed turbulence model for helicopter simulation. A simple approach to modeling the aircraft response to turbulence is described by using an identified model of the Black Hawk to extract representative control inputs that replicate the aircraft response to disturbances. This parametric turbulence model is designed to be scaled for varying levels of turbulence and utilized in ground or in-flight simulation. Flight control cutoff frequency data are also analyzed to support design criteria for gust rejection handling qualities.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
	A. GENERAL	1
	B. SCOPE OF RESEARCH	1
II.	BACKGROUND.....	3
	A. SUMMARY OF PAST RESEARCH	3
	B. TECHNICAL APPROACH.....	3
	C. FLOW AROUND BLUNT OBJECTS	6
III.	RESEARCH AIRCRAFT AND FLIGHT TEST	11
	A. UH-60 BLACK HAWK AIRCRAFT DESCRIPTION	11
	B. FLIGHT TEST DATA	13
	C. FLIGHT TEST CONDITIONS.....	14
	D. FLIGHT TEST TECHNIQUE	16
IV.	DATA ANALYSIS TOOLS	19
	A. CIFER®	19
	1. Input Data.....	19
	2. Frequency Response Calculation	21
	3. Transfer-Function Modeling	22
	B. MATLAB®	22
	C. TRENDS	23
V.	ADS-33 DISTURBANCE REJECTION CRITERIA.....	25
	A. CUTOFF FREQUENCY ANALYSIS	25
	1. Longitudinal	26
	2. Lateral.....	28
	3. Directional.....	30
	B. FLIGHT CONTROL REQUIREMENTS	32
VI.	GUST MODEL IDENTIFICATION METHOD/DATA ANALYSIS	39
	A. OVERVIEW.....	39
	B. AIRCRAFT MODEL DETERMINATION.....	40
	1. Gen Hel Advanced Rotor/Helicopter Model.....	41
	2. On-Axis Transfer Function Model	42
	3. Identified Coupled Model	45
	4. Comparison of Various UH-60 Models	46
	C. DETERMINATION OF REMNANT AIRCRAFT RATES	47
	D. INVERSE MODEL.....	48
	E. RESULTANT GUST MIXER INPUT	53
	F. FORMATION OF GUST MODEL	57

G.	CHECK OF MODELING PROCESS	61
H.	UTILIZATION OF GUST MODEL.....	64
VII.	CONCLUSIONS AND RECOMMENDATIONS.....	67
A.	CONCLUDING REMARKS	67
B.	RECOMMENDATIONS FOR FOLLOW-ON RESEARCH.....	68
	LIST OF REFERENCES	69
	APPENDIX A. SIMULINK® MODELS	71
	APPENDIX B. MATLAB® SCRIPTS.....	75
1.	SCRIPT SSSS.M.....	77
2.	SCRIPT FORMDATA.M	79
3.	SCRIPT PLOTDIFFS.M.....	80
4.	SCRIPT PZMOD.M.....	81
5.	SCRIPT INVCHECKSS.M.....	84
6.	SCRIPT SSSS_COUP.M.....	88
7.	SCRIPT INVCHECKSS_COUP.M.....	91
8.	SCRIPT SSTF.M.....	95
9.	SCRIPT INVCHECKTF.M	97
10.	SCRIPT BODECHECK.M	100
11.	SCRIPT PITCHCHECKER.M.....	104
12.	SCRIPT ROLLCHECKER.M.....	105
	INITIAL DISTRIBUTION LIST	107

LIST OF FIGURES

Figure 1. Schematic of overall approach of disturbance rejection study.	4
Figure 2. Isometric sketch of flow around a cube. From Ref. [6].	8
Figure 3. Flow on the centerline of a cube. From Ref. [6].	8
Figure 4. Roof flow behavior. Recirculation region length in deep boundary layer flows with normal wind. From Ref. [6].	9
Figure 5. Turbulent intensity variations. From Ref. [7].	10
Figure 6. Helicopter data acquisition system.	13
Figure 7. Control system and position sensors.	14
Figure 8. UH-60A hovering on leeward side of CGAS hangar.	15
Figure 9. Software and Database Components Used in the System Identification Procedure. From Ref. [10].	20
Figure 10. Longitudinal cutoff frequency differences for strong turbulence – Flight 2. ..	27
Figure 11. Longitudinal cutoff frequency differences for strong turbulence – Flight 1. ..	28
Figure 12. Lateral cutoff frequency differences for strong turbulence – Flight 2.	29
Figure 13. Lateral cutoff frequency differences for strong turbulence – Flight 1.	29
Figure 14. Directional cutoff frequency differences for strong turbulence – Flight 2.	31
Figure 15. Directional cutoff frequency differences for strong turbulence – Flight 1.	31
Figure 16. Definition of bandwidth.	33
Figure 17. ADS-33D small amplitude pitch attitude requirement - hover and low speed. From Ref. [2].	34
Figure 18. ADS-33D small amplitude roll attitude requirement - hover and low speed. From Ref. [2].	34
Figure 19. RASCAL model following control system.	36
Figure 20. UH-60 disturbance bandwidth vs. control crossover frequency.	36
Figure 21. Utilization of gust excitation model.	39
Figure 22. Computation of remnant aircraft rate.	41
Figure 23. Pitch doublet response of UH-60A and on-axis pitch transfer function model.	43
Figure 24. Roll doublet response of UH-60A and on-axis roll transfer function model. ...	44
Figure 25. Roll doublet response of UH-60A and 14-DOF model.	45
Figure 26. Comparison of pitch rate to longitudinal mixer frequency response characteristics of various models.	46
Figure 27. Comparison of roll rate to lateral mixer frequency response characteristics of various models.	47
Figure 28. Simulink [®] model of differencing process with state-space model.	48
Figure 29. Derivation of disturbance control positions.	49
Figure 30. Check solution for inverse aircraft model options.	51
Figure 31. Comparison of check solution with aircraft data for on-axis and coupled inverse models.	53
Figure 32. Longitudinal gust mixer autospectrum for varying pilot gain.	54
Figure 33. Coherence function for pilot mixer input to identified gust remnant.	55

Figure 34. Longitudinal gust mixer autospectrum for varying turbulence levels.	55
Figure 35. Longitudinal gust mixer autospectrum for varying aircraft heading orientation.	56
Figure 36. Gust excitation model for flight 1 (light turbulence).....	62
Figure 37. Gust excitation model for flight 2 (strong turbulence).	62
Figure 38. Gust excitation model characteristics for varying levels of turbulence.....	63
Figure 39. Comparison of longitudinal gust remnants for light turbulence (flight 1).....	64
Figure 40. Comparison of lateral gust remnants for strong turbulence (flight 2).	64
Figure 41. Example gust model implementation in RASCAL analysis.....	65
Figure A-1. Model of differencing process (extraction of gust remnant) with state-space model.....	71
Figure A-2. On-axis inverse model for obtaining mixer input to simulate gust.	71
Figure A-3. Coupled inverse model for obtaining mixer input to simulate gust.	72
Figure A-4. Check of inverse process by summing	72
Figure A-5. Check of gust excitation model using random number generator.....	73

LIST OF TABLES

Table 1. UH-60A General Specifications	12
Table 2. Flight Test Conditions.....	15
Table 3. Spectral filter fit coefficients for gust mixer model; $K/(s+\alpha_w)$	59

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENT

The author would like to acknowledge the support of the many members of the Flight Control and Cockpit Integration Branch of the Army/NASA Rotorcraft Division in this research effort. Financial support, camaraderie, and willingness to educate a helicopter test pilot in the ways of the flight test/research engineer made me feel as though I were a valuable member of their team. The AFDD Flight Projects Office, in particular LTC Chris Sullivan, CWO3 Rich Huber and MAJ Dave Arterburn, was instrumental in completing the flight test associated with this project and enabling the author to be a member of the flight test aircrew. The assistance of Jason Colbourne is appreciated in generating the CONDUIT results for RASCAL. The support of Kurt Long, of Naval Air Warfare Center, Aircraft Division, Patuxent River, in recording flow field parameters was outstanding.

Special thanks to my thesis advisors, Dr. Mark Tischler, for his patience and guidance in the ways of frequency domain testing and introduction to the world of research, and Dr. E.R. Wood, for his steadfast support of this cooperative research effort. Thanks also to Mr. Chris Blanken, the lead engineer on the project, for introducing me to the research team and providing support and guidance throughout. The author would also like to thank Professor Ronald A. Hess, University of California, Davis, for his insight and consultation in reviewing the results of this research.

Lastly, I would like to thank my wife Emily and our four children, for their patience, support and understanding during the long hours and time away from home required by this project.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. GENERAL

The effects of wind/turbulence on helicopter handling qualities are not well developed nor quantified. It is widely recognized that as the level of wind or turbulence increase, task performance may be compromised and/or pilot workload increased leading to a degradation in the handling qualities and safety. Although Baillie and Morgan [Ref. 1] have performed a disturbance-rejection handling qualities study, very few parametric wind or turbulence handling qualities studies exist. In fact, there is no supporting data for the disturbance rejection requirements in Aeronautical Design Standard – 33 (ADS-33D-PRF), the US Army adopted handling qualities requirements for military rotorcraft [Ref. 2]. One reason for the lack of wind/turbulence effects in the rotorcraft handling qualities data base has been that wind/turbulence models have been inadequate, not validated, and are difficult to implement and use.

B. SCOPE OF RESEARCH

The focus of this disturbance rejection project is: 1) to provide support for disturbance rejection requirements in ADS-33; 2) to provide a simple wind/turbulence model for ground and in-flight handling qualities simulations; and 3) to provide data for a gust response model of a UH-60A Black Hawk. Flight test data was recorded from an instrumented aircraft flying in turbulent conditions with the goal of providing supporting data for ADS-33 gust rejection criteria. In the process of meeting this goal, an analysis of

pilot cutoff frequency data for varying levels of pilot gain and turbulence level was conducted in order to correlate the results of this study to the existing criteria in ADS-33. Development of an empirical low speed turbulence model for the UH-60 was also pursued in order to facilitate future flight control design studies with an accurate turbulence model for use in a ground or in-flight simulator.

This helicopter disturbance rejection research was conducted within the combined Army/NASA Rotorcraft Division under the NASA SAFOR program (Safe All Weather Flight Operations for Rotorcraft), and the rotary wing segment of the Army's Flight Control Technology Development Approach (TDA) program.

II. BACKGROUND

A. SUMMARY OF PAST RESEARCH

The spectrum of helicopter turbulence modeling ranges from the simpler frozen field turbulence model [Ref. 3] to the complex rotating frame model [Ref. 4]. Traditional fixed-wing turbulence modeling has assumed a spatially frozen gust pattern, through which an aircraft flies, similar to a car driving down a bumpy road. The frequency content of the "frozen field" is often quoted in two forms, von Karman and the simpler Dryden approximation. In Hess' analysis of rotorcraft handling qualities in turbulence [Ref. 5], he utilizes a "frozen" turbulence field convecting at a given velocity toward the stationary vehicle. In other words, the bumpy road is moving past the car. He studied the effects of flight control characteristics on handling qualities in turbulence by including the effects of the turbulence gradients in approximate fashion in the rotorcraft equations of motion. This approach serves as a first-order correction to the approach typically used in the modeling of aerodynamic forces and moments due to turbulence for fixed-wing aircraft.

B. TECHNICAL APPROACH

In a desire to address the objectives of the research, the technical approach overview is portrayed in Figure 1. The parametric modeling effort required establishing the flight test environment to gather the aircraft data in an area of substantial turbulent airflow. A hover task was conducted on the leeward side of a cube-like aircraft hangar

during moderate to strong wind conditions. The intricate flow field in this location is difficult to model but provides the desired level of turbulent air flow. The hover location was initially targeted based on flow theory and refined by the aircrew to maximize the turbulent effects on the aircraft.

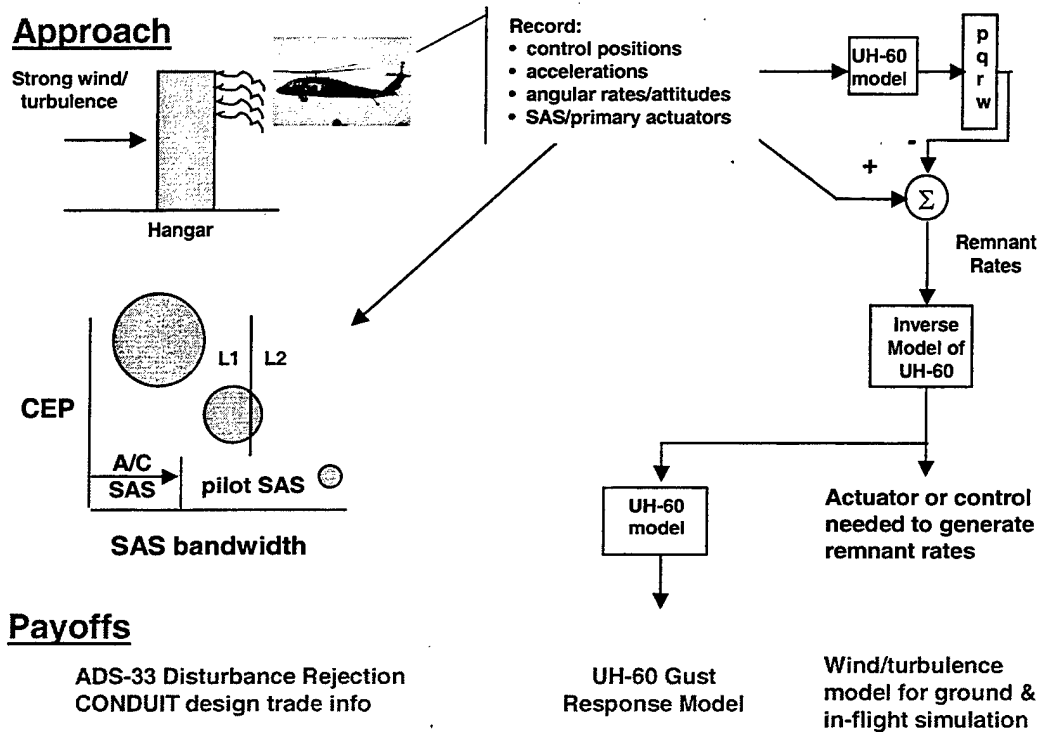


Figure 1. Schematic of overall approach of disturbance rejection study.

In support of the ADS-33 disturbance rejection criteria, the flight test attempted to characterize measured pilot control activity relative to ADS-33 bandwidth requirements. The quantitative requirements of ADS-33 are divided into two flight regimes, Hover/Low Speed and Forward Flight, and further divided by axes of control. These requirements are separated not only by response amplitudes but also for control inputs versus disturbance inputs. The current ADS-33 requirements for short-term pitch, roll, and yaw

responses to disturbance inputs are the same as the control response bandwidths. There are no disturbance rejection requirements in the heave axis. For the qualitative flight test demonstration maneuvers in ADS-33, although there are maneuvers which are only evaluated in calm winds, there are several maneuvers that are evaluated both in calm winds and in moderate winds. There is little or no supporting data for either the disturbance rejection requirements or the moderate wind effects on the flight demonstration maneuvers.

In the area of wind/turbulence modeling, the predominately used fixed-wing Dryden turbulence model, based on a fixed turbulence field, is not valid for hovering rotorcraft. This is because the turbulence model scaling parameters are a function of the vehicle velocity. So, as the helicopter comes to a hover, the model requires a division by zero. The basic modeling assumptions break down and the model is not suitable for rotorcraft in hovering or low speed flight. The recent research on blade-centered and cyclo-stationary random processes for treatment of rotorcraft turbulence modeling, while valid, are complex in terms of modeling, tuning and implementation into ground or in-flight simulation.

In contrast, the present disturbance rejection effort develops a realistic and simple empirical model by extracting the aircraft rates due to atmospheric disturbances from flight test data and modeling a control input spectrum to simulate these rate responses. This first order attempt at characterizing the aircraft response to turbulence makes no attempt to model the exact details of the rotor response that would be contained in a rotating frame turbulence model. It should also be noted that this is an effort to model

aircraft control inputs, which generate the aircraft response to turbulence, and not the components of the gust velocities and gradients themselves. Coupling this model to the UH-60 math model provides the tool needed to develop a UH-60 gust response model. This capability can be used for parametric studies of disturbance rejection with ground or in-flight simulators, as well as for future rotor state flight control trade-studies.

C. FLOW AROUND BLUNT OBJECTS

In the focus on developing a simple wind/turbulence model for handling qualities simulations the goal was to gather low speed flight test data in a turbulent-rich environment. The predominant task for this effort was decided to be hover on the leeward side of a building. Initially, the flight test was designated to occur at Moffett Field, but due to the absence of consistent strong winds during the test period an alternative site was selected at the Coast Guard Air Station, San Francisco. The aircraft hangar at this location was suitable for our testing purposes, being a blunt, cube-like object with an adequate hover location over the flight line and predominant winds across the hangar.

The objective was to find a target hover location where the turbulence intensity would be at its greatest. Research on flow around blunt objects was conducted to give the pilots a feel for the airflow distribution in the environment surrounding the hangar. Fackrell shows in Figure 2 [Ref. 6] an isometric sketch of a flow pattern around a cube, and Figure 3 [Ref. 6] shows the flow on the centerline around a cube. Note in Figure 3, there is a recirculation region, L_r , downstream of the building. For a wind azimuth

aligned perpendicular to the face of the building, the size of this recirculation region is a function of the building aspect ratios, that is, the ratios of width to height and length to height. Figure 4 [Ref. 6], from Fackrell, shows the relationship between these building aspect ratios and the size of the recirculation region. Another important ingredient in estimating optimum locations to obtain high turbulent intensities is not only the distance from the building but also the height relative to the building. Results from Castro et al. (Figure 5 [Ref. 7]) suggest that flying at or just above the height of the top of the building will yield the highest turbulence intensities, i.e., in the neighborhood of 20 to 25% of the free stream velocity. In Figure 5(a), the intensities are shown for $x/h = 1$, i.e., downstream of the building an amount equal to the building height. Note that in Figure 5(b) for $x/h=2$, the overall peak intensity may be less compared to $x/h=1$, but the "thickness" of the intensity area has dramatically increased suggesting that there's a larger area of turbulence at $x/h=2$. The hover work was done on the leeward side of the Coast Guard Hangar (Figure 8 shows the aircraft and hangar). It was assumed that with the wind predominantly from the west (i.e., down the length of hangar), then for this hangar, the size of this recirculation region, L_r , is approximately 100 feet. For safety purposes, the aircraft was restricted to hover 100 feet from the hangar, which was a reasonable starting position for the hover test. This put the aircraft at an x/h value of approximately 1 therefore Figure 5 indicates a target height (y/h) between 1.0 and 1.75. For our purposes, the pilots started their hover task at a height level with the roof of the hangar and varied altitude to find the maximum turbulence area in the hangar wake.

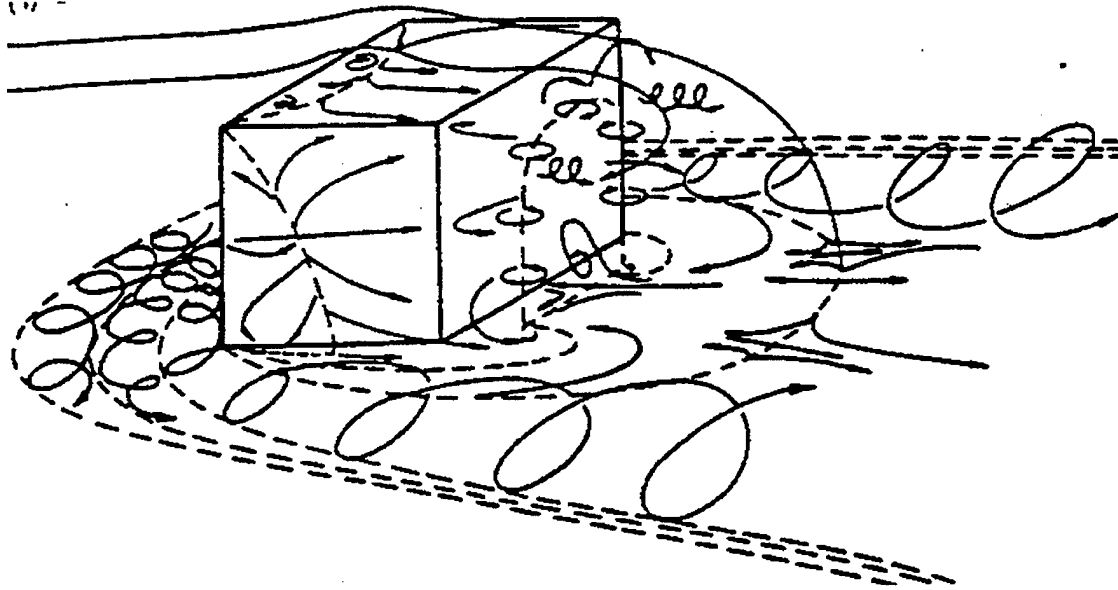


Figure 2. Isometric sketch of flow around a cube. From Ref. [6].

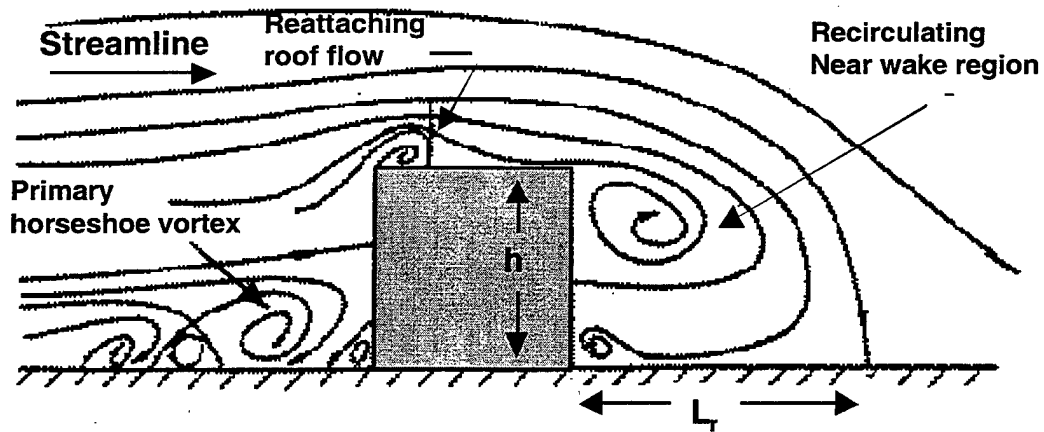


Figure 3. Flow on the centerline of a cube. From Ref. [6].

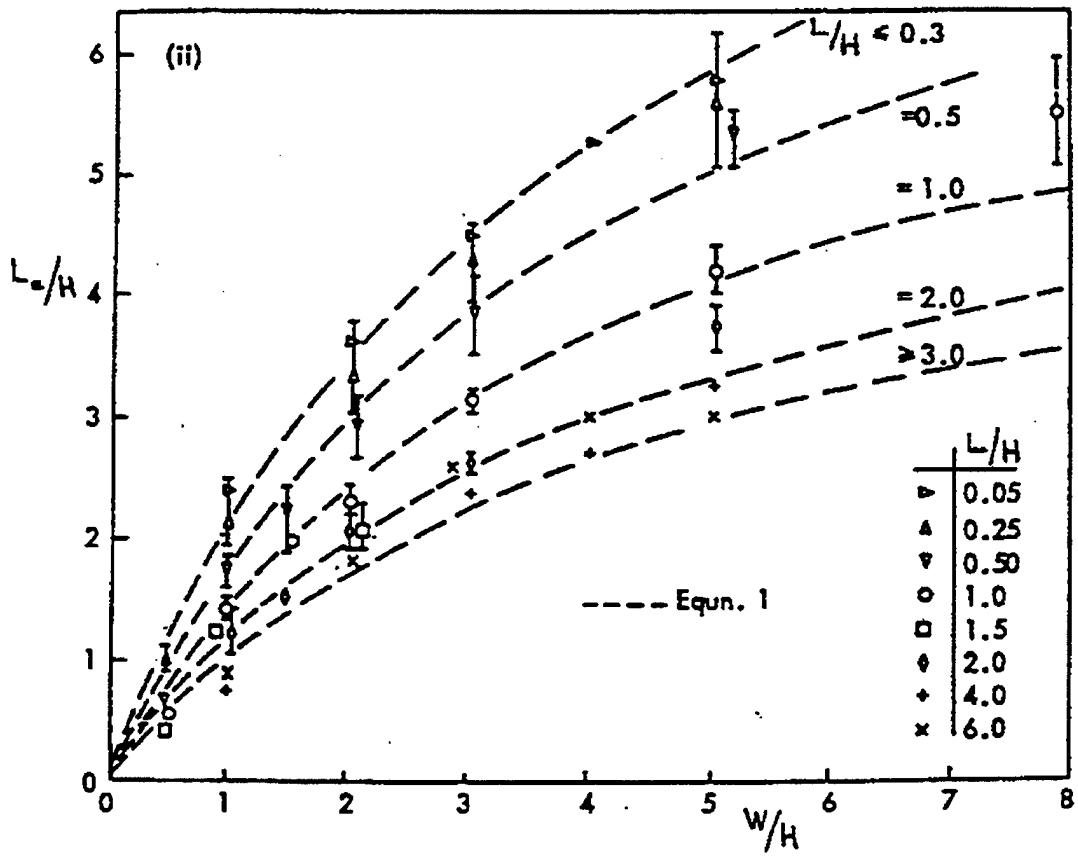


Figure 4. Roof flow behavior. Recirculation region length in deep boundary layer flows with normal wind. From Ref. [6].

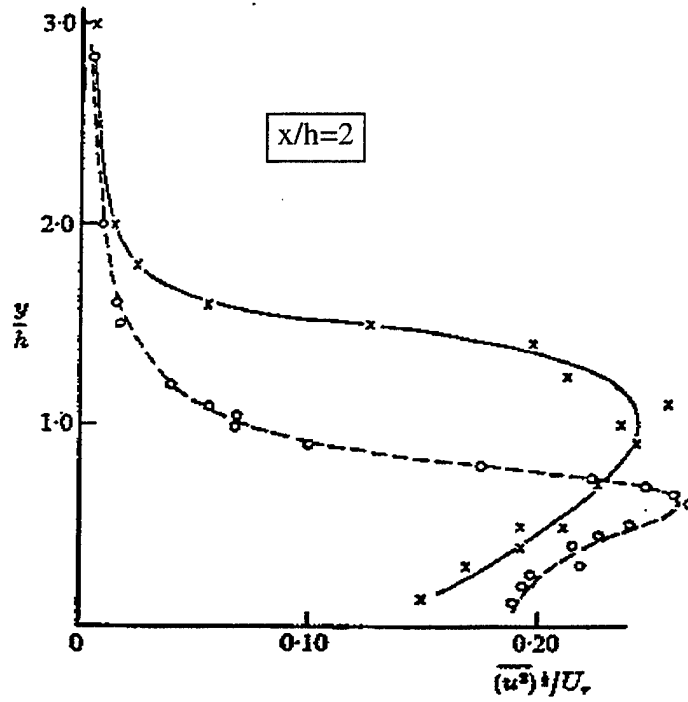
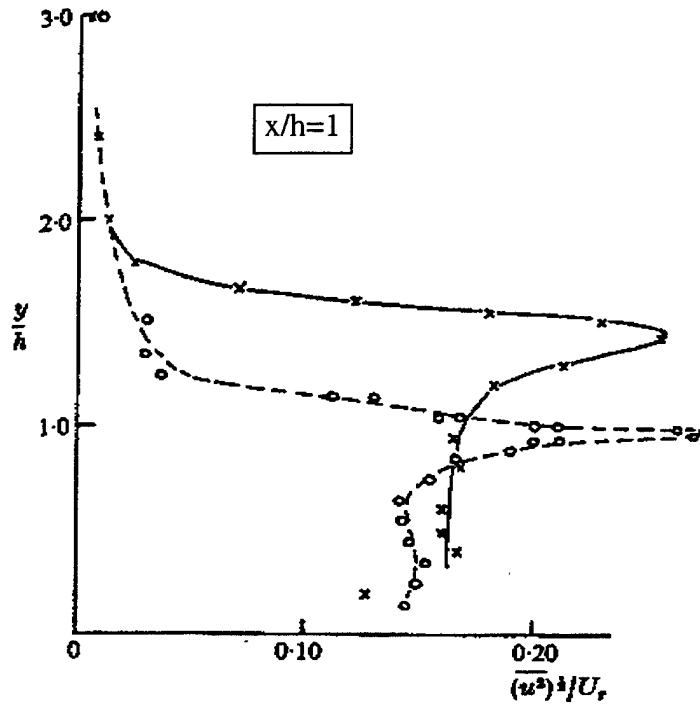


Figure 5. Turbulent intensity variations. From Ref. [7].

III. RESEARCH AIRCRAFT AND FLIGHT TEST

A. UH-60 BLACK HAWK AIRCRAFT DESCRIPTION

The UH-60A Black Hawk helicopter is a twin turbine powered, dual piloted, single main rotor utility helicopter manufactured by Sikorsky Aircraft, a division of United Technologies. The aircraft is designed to carry a crew of three and eleven combat equipped troops at a primary mission gross weight of 16,825 pounds and a maximum gross weight of 20,250 pounds. The propulsion system has two General Electric T700-GE-700 turboshaft engines operating in parallel with a maximum standard day, sea level, installed rating of 1560 SHP each. With both engines operating, the transmission limit for each engine is 1155 ESHP (100% torque). The engines simultaneously drive a fully articulated main rotor and a 20 degree canted tail rotor. The drive train consists of a main transmission, intermediate gear box and tail rotor gear box with interconnecting shafts. The main rotor has four blades with an average chord of 20.8 inches and a diameter of 53.7 feet. The tail rotor has four blades with a chord of 9.7 inches and a diameter of 11 feet. The helicopter flight control system is irreversible and utilizes hydraulic boost which is supplemented with an automatic flight control system (AFCS). The AFCS consists of four subsystems: two stability augmentation systems (SAS), electric trim, flight path stabilization (FPS) and an automatic stabilator. The SAS is designed to provide short term damping in the pitch, roll, and yaw axes. Each SAS provides 5% control authority, for a total of 10%. The trim system provides a gradient force to maintain cyclic and tail rotor pedal position and provides the input forces for the FPS.

The FPS is a basic autopilot that enhances static stability in the pitch, roll, and yaw axis. When coupled with the trim system, the FPS has 100% control authority. A mixing unit installed as part of the flight control system, and located at the output of the pilot-assisted servos is designed to minimize inherent control coupling by providing control mixing. The stabilator is a variable angle of incidence airfoil that is designed to improve flying qualities by positioning in response to collective, airspeed, pitch rate, and lateral acceleration inputs. A more complete description of the test aircraft can be found in the Operator's Manual [Ref. 8] and general specifications can be found in Table 1. The test aircraft, USA S/N 82-23748, is a sixth year production Black Hawk which incorporates the External Stores Support System fixed provisions and fairings, the reoriented production airspeed probes, and the modified production stabilator schedule.

Table 1. UH-60A General Specifications.

Operating Weights and Engine Power		
Empty Weight (lbs)	11,563	
Fuel Weight, Typical (lbs)	2,446	
Takeoff Weight, Typical (lbs)	14,609	
Maximum Takeoff Weight (lbs)	20,250	
Maximum Takeoff Rating (shp)	3,086	
Maximum Useful Power (shp)	2,828	
Rotor Parameters	Main Rotor	Tail Rotor
Radius (ft)	26.83	5.5
Chord (ft)	1.73	0.81
Solidity Ratio	0.082	0.188
Number of Blades	4	4
Rotor Rotational Speed (rad/sec)	27.02	124.54
Tip Speed (ft/sec)	725	685

B. FLIGHT TEST DATA

The research instrumentation and data acquisition system on the helicopter consists of sensors, signal conditioners, pulse-code modulation (PCM) encoder, time code generator, tape recorder, and a transmitting antenna, Figure 6. The helicopter sensor signals include accelerometers, rate and attitude gyros, control position sensors at several points in the control system, and air data sensors. These signals are passed through filters and encoded in a PCM stream, which is then recorded on a tape recorder and also transmitted to the ground telemetry station. Of primary concern in this research effort were the cockpit and mixer flight control positions and the aircraft rate and acceleration responses. A schematic of the flight control system and flight test sensors is presented in Figure 7.

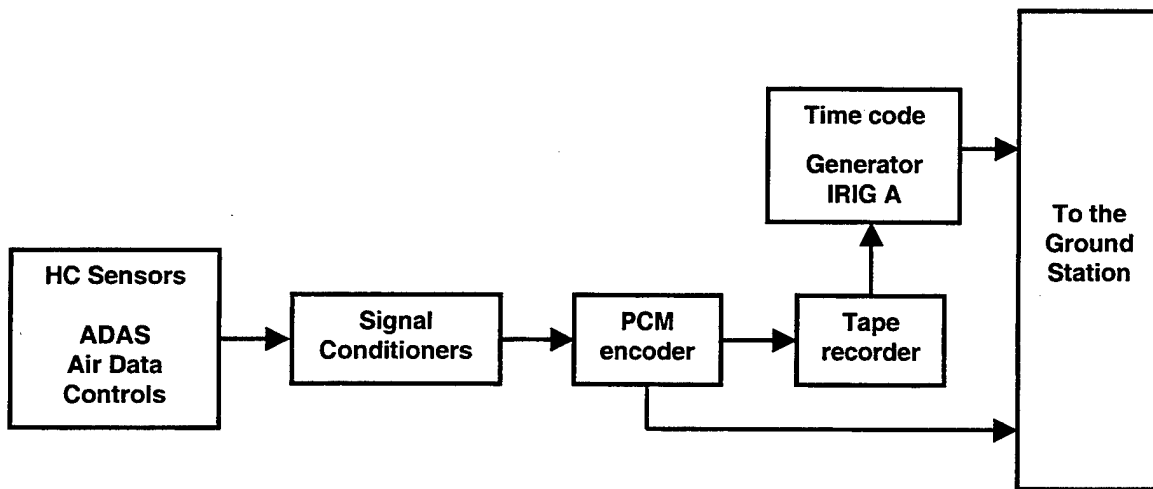


Figure 6. Helicopter data acquisition system.

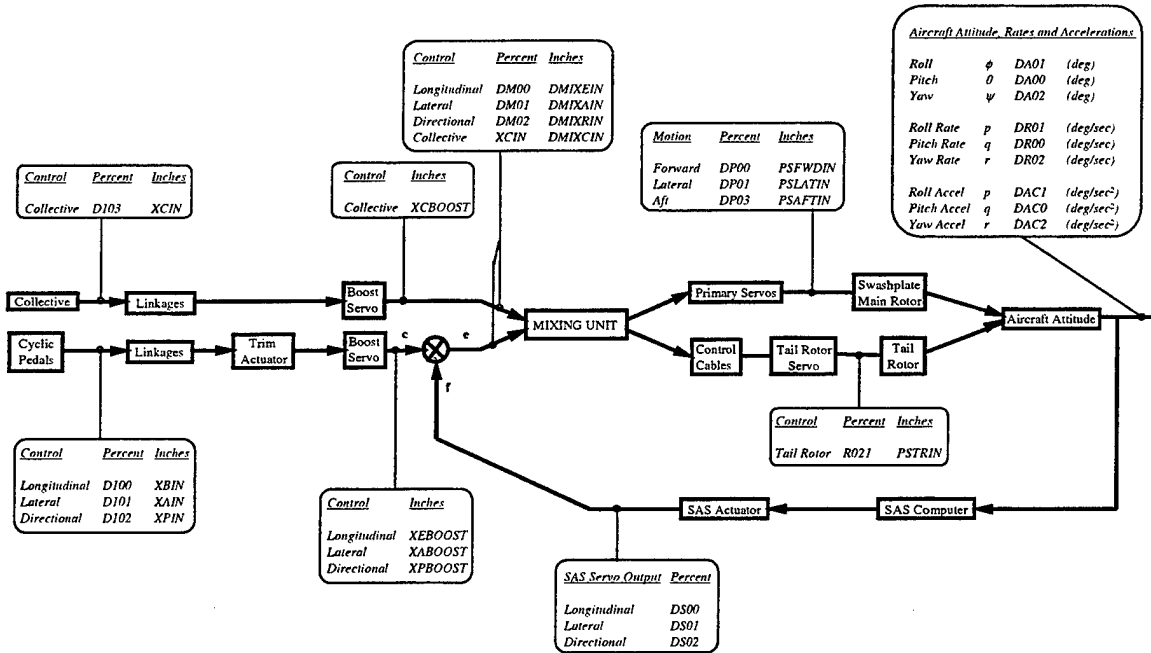


Figure 7. Control system and position sensors.

C. FLIGHT TEST CONDITIONS

The flight test site was the US Coast Guard Air Station at the San Francisco airport. The air station is at the north end of the airport, near the water's edge, and has a large ramp area surrounding the hangar on the leeward side. The hangar was approximately 40 feet in height. A hover task was conducted on the leeward side of the hangar during moderate to strong wind conditions, Figure 8.

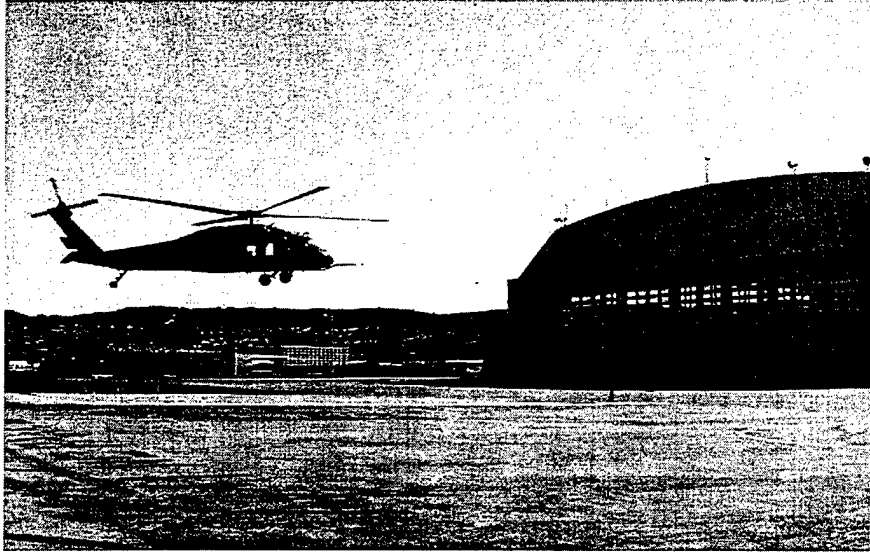


Figure 8. UH-60A hovering on leeward side of CGAS hangar.

Three data flights were conducted over a two month period, targeting windy days to produce maximum turbulence effects. The initial flight was conducted SAS-on and the two subsequent flights were conducted SAS-off. Flight test conditions of interest for the SAS-off flights are provided in Table 2. The average wind deviation at the aircraft hover location was measured immediately after the test at a height just below the aircraft hover location. These values are considered to represent a one sigma deviation in the test data for the turbulence modeling effort.

Table 2. Flight Test Conditions.

Flight Number	OAT (°C)	Average Wind on Roof (kts)	Average Wind Deviation at Hover Location (kts)
1	16	17	+/- 2
2	18	22	+/- 5

D. FLIGHT TEST TECHNIQUE

The test aircraft was operated in accordance with Ames Research Center procedures as set forth in the Aeroflightdynamics Directorate (AFDD) Standard Operating Procedures (SOP) and appropriate Army regulations.

Comprehensive preflight briefings were conducted jointly by the test director and the project pilot for each flight. These briefings covered the overall goal of each flight, including the specifics on each test data point for the applicable data card, the data required, limitations (e.g., a minimum horizontal clearance of 100 ft between the helicopter and the hangar was imposed), test techniques, and expected results. The mission ground rules were also briefed, including the necessary conditions to commit to fly.

Hover altitude was targeted based on flow theory around blunt objects and the pilots found two distinct locations where the turbulent effects appeared maximum. At this point, the evaluation pilots conducted two hover tasks to different performance standards, tight (+/-5 ft) and relaxed (+/-15 ft) in order to capture varying levels of pilot control loop closure. Data were collected with the aircraft pointed into the wind and perpendicular to the wind with each pilot.

An on-board crew member helped to assess aircraft X-Y position and deviations from a desired hover position. This crew member viewed the aircraft's shadow (on a sunny day) relative to graduated markings placed on the ramp. Personnel on the ground equipped with radio communication to the helicopter helped to monitor and maintain the helicopter's distance from the hangar.

A series of wind speed and direction measurements were recorded using a handheld Davis Instruments, Inc. cup and vane anemometer unit. The actual speed and direction sensor was mounted to the end of an adjustable telescopic pole assembly, which could be raised or lowered to obtain speed and direction data at the desired height. The cup and vane sensors were mounted so that their spin axes were horizontal; thus, by orienting the spin plane vertically/longitudinally, measurements of longitudinal and vertical velocity could be obtained, and a simple 90 degree directional pivoting of the pole mount would rotate the sensor so that its spin plane was oriented vertically/laterally, producing vertical and lateral speed measurements. The speed and direction outputs were displayed digitally at the base of the mounting pole and were recorded manually after each reading.

The technique used to obtain data was to place the pole/anemometer unit in the desired location, at the desired height, and aligned so that its spin axis was oriented laterally. The anemometer operator observed the speed and direction outputs for a period of approximately 20 seconds. The speed and direction maxima, minima, and mean attained during this time were acquired from the digital display and recorded as the vertical/longitudinal data. After recording these, the operator pivoted the assembly 90 degrees, repeated the process, and recorded the results as the vertical/lateral data. The series of measurements was then continued for various locations, both on the hangar roof (to quantify test conditions), and on the ground, in the wake of the hangar (to quantify velocities in the region of helicopter operation). Hangar roof measurement locations included right, left, and centerline, at both upstream and downstream ends of the hangar.

For the test wind conditions, the downstream centerline location was chosen as the reference condition. Ground measurement locations were chosen so as to provide information at a variety of locations near the helicopter's hover positions. All hangar roof measurements were conducted while the helicopter was hovering, downwind of the hangar face. All ground measurements were conducted immediately following helicopter departure, as a compromise between personnel safety and measurement expedience.

Uncertainties in anemometer measurements, unless otherwise noted, are approximately +/- 1 foot in position, +/- 5 degrees in direction, and +/- 0.5 knots in speed.

At the conclusion of each test flight, a post-flight debriefing was held to review the UH-60A post-flight debriefing checklist, and any anomalies of the aircraft or the instrumentation system. Additionally, the post-flight debrief identified priorities and requirements for the real-time and post-flight analysis of the data.

IV. DATA ANALYSIS TOOLS

A. CIFER[®]

The U.S. Army has developed a frequency domain data analysis package known as CIFER[®] (Comprehensive Identification from Frequency Responses) which is ideally suited to the task of rotorcraft system identification. This integrated software package takes flight test data and performs batch execution of computationally-intensive procedures to match the frequency response data with a dynamic model of the helicopter. The package includes interactive plot and report utilities which enable the user to provide a convenient form of presenting the processed data. The software and database components of CIFER[®] are presented in Figure 9.

1. Input Data

CIFER[®] requires a database of aircraft state output parameters resulting from pilot-generated frequency sweep inputs. The sweeps are designed to excite the vehicle dynamics of interest, including all of the rigid body and lower-frequency rotor dynamic modes. The data is run through a data compatibility analysis using the Kalman filter/smoothing program SMACK (Smoothing for Aircraft Kinematics). This procedure determines reduced parameter-set models of measurement system errors involving unknown scale factors and biases, and estimates of unknown states and/or noisy measurements are reconstructed [Ref. 9].

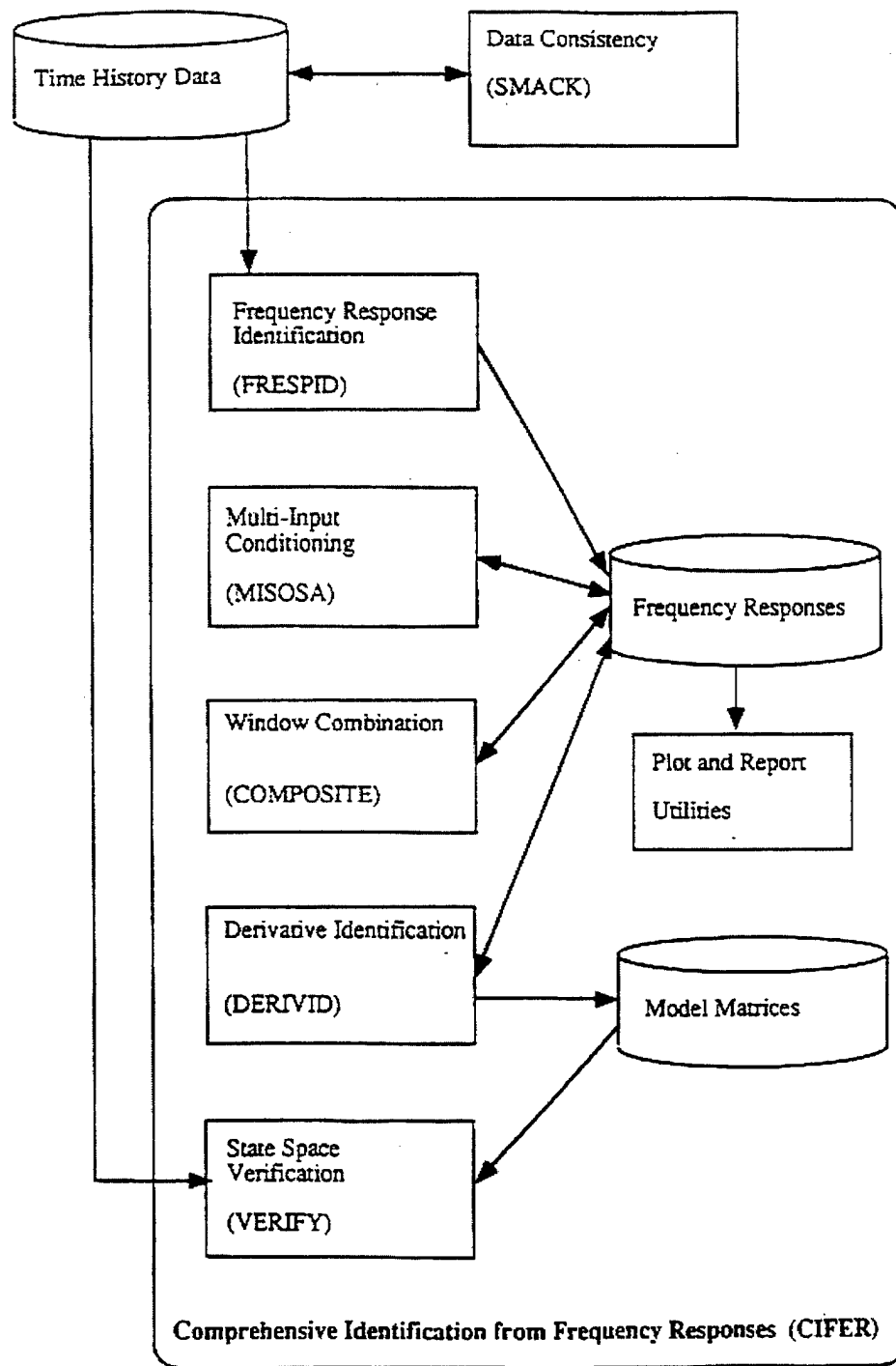


Figure 9. Software and Database Components Used in the System Identification Procedure. From Ref. [10].

2. Frequency Response Calculation

A frequency response is a complex-valued function that relates the Fourier Transform of the system output to the Fourier Transform of the system input.

$$Y(f) = H(f)X(f) \quad (1)$$

where f = frequency in Hz.

A physical interpretation of the frequency response can be seen as:



The frequency response will fully characterize the dynamics of the aircraft.

CIFER[®]'s FRESPID module uses a Chirp-Z transform, which is a very flexible FFT algorithm, to calculate the frequency response. Time history data from several flight records are concatenated to increase the spectral content of the sample. The data is windowed, the transform applied to each window, and the results are spectrally averaged to reduce random error. The frequency response is calculated by normalizing the cross spectrum by the input autospectrum [Ref. 11]:

$$H(\omega) = \frac{G_{xy}(\omega)}{G_{xx}(\omega)} \quad (2)$$

The coherence function, the fraction of the output power that is linearly related to the input power, is calculated as:

$$\gamma_{xy}^2(\omega) = \frac{|G_{xy}(\omega)|^2}{|G_{xx}(\omega)||G_{yy}(\omega)|} \quad (3)$$

This is an indicator of frequency response accuracy and a drop in the coherence function indicates poor accuracy over the specified frequency range [Ref. 9].

The COMPOSITE module of CIFER[®] combines frequency responses calculated from different window sizes into an optimized composite frequency response. This eliminates the requirement to manually optimize window sizes and results in high coherence and low random error in the database over the frequency range of interest.

3. Transfer-Function Modeling

In the case where only the input-to-output dynamic behavior is of concern, the NAVFIT module, first developed at McDonnell Douglas and modified for use at NASA Ames, is applied [Ref. 9]. A transfer-function model contains the least possible number of parameters that will characterize the system. The transfer function model is formulated in coefficient form based on a nonlinear (Rosenbrock) least-squares minimization of the cost function. The single input-single output fit of magnitude and phase characteristics is performed for a user-selected frequency range of fit and number of points, and the coefficients of the transfer-function model and time delay can be individually fixed or freed.

B. MATLAB[®]

The predominant tool used in the data analysis was MATLAB[®]. All scripts and subroutines were coded in M-files and simulations were executed using SIMULINK[®] models. All pertinent MATLAB[®] scripts are presented in Appendix B and are archived

on the NASA Ames AFR server. Scripts were written and executed using version 5.1 of MATLAB[®].

C. TRENDS

All flight test data was archived in the TRENDS interactive Database Operating System by NASA personnel. This database was developed by NASA to support various rotorcraft research studies, beginning with the XV-15 tiltrotor. The TRENDS database facilitated formatting the flight test data for use with various data analysis tools including CIFER[®], MATLAB[®], and SIMULINK[®].

THIS PAGE INTENTIONALLY LEFT BLANK

V. ADS-33 DISTURBANCE REJECTION CRITERIA

A. CUTOFF FREQUENCY ANALYSIS

In flight control modeling and design, a primary concern is the frequency of controller input required to achieve a desired task performance. This is true for both piloted control and automatic control systems. When referring to the closed-loop aircraft response, this is the "bandwidth frequency." This frequency is typically defined for pilot-in-the-loop requirements, such as in ADS-33D, as the -135 degree phase frequency for the attitude response. For automatic stabilization systems, this requirement is typically expressed in terms of the -3 dB magnitude closed-loop bandwidth frequency (or "half-power" frequency), which also corresponds closely to the broken-loop 0 dB "crossover frequency", ω_c . The proper selection of crossover frequency is a key control system design choice, which has implications on actuator and sensor bandwidth, flight computer throughput, and flight control/structural response coupling.

The cut-off frequency ω_{co} [Ref. 12] is determined from a spectral analysis of the control deflection time history data $\delta(t)$, and is a good estimate of the -3 dB control bandwidth (and crossover frequency). The cut-off frequency defines the upper end of the frequency range that encompasses one-half of the total area under the measured autospectrum curve (thus the half-power frequency).

$$\frac{\int_0^{\omega_{co}} G_{\delta\delta}(\omega) d\omega}{\int_0^{\infty} G_{\delta\delta}(\omega) d\omega} = 0.5 \quad (4)$$

where;

ω_{co} = pilot cutoff frequency

$G_{\delta\delta}$ = autospectrum of the controller input

Another interpretation of cut-off frequency is the band limit that encompasses 70.7% (-3dB) of the total control input RMS (since the RMS ratio is just square-root of the power ratio). The RMS utility in CIFER[®] enabled the rapid and accurate extraction of the cutoff frequency characteristics from the control input flight data. These results provide the control system design requirements as a function of turbulence level and accuracy (position tolerance). Flight data were collected with the UH-60 SAS disengaged, thus the pilot was providing all of the required stabilization.

1. Longitudinal

The flight control cutoff frequency was examined to determine the frequency content trends with varying position tolerance. Longitudinal mixer input cutoff frequencies for two hover conditions, tight position tolerance (+/-5 ft) and relaxed tolerance (+/-15 ft), are plotted against approximate hover position tolerances for both flight data sets, Figure 10 and Figure 11. The figures indicate that with a tighter required tolerance, the cutoff frequency increases. At the stronger level of turbulence, Figure 10, the average longitudinal cutoff frequency required to maintain a tight position tolerance was 2.5 rad/sec and for basic stabilization (relaxed tolerance) an average cutoff frequency of 1.8 rad/sec was required. At the lower level of turbulence, Figure 11, the longitudinal

cutoff frequency required to maintain a tight position tolerance was 1.8 rad/sec and for basic stabilization a cutoff frequency of 1.2 rad/sec was required.

It is interesting to note that the same level of longitudinal cutoff frequency was required regardless of aircraft orientation. In other words, whether the aircraft was facing the hangar or perpendicular to the hangar, the longitudinal cutoff frequency was the same for identical tasks. Therefore the exact modeling of the geometry of the aircraft and turbulence does not play a strong role in determining the control system requirements, and the use of first principle math models for control system handling qualities evaluations is acceptable.

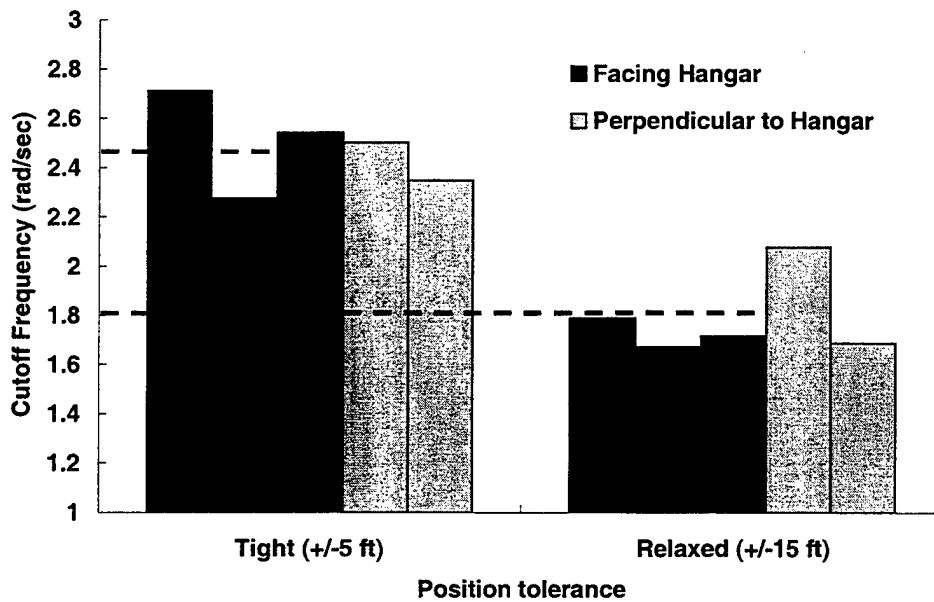


Figure 10. Longitudinal cutoff frequency differences for strong turbulence – Flight 2.

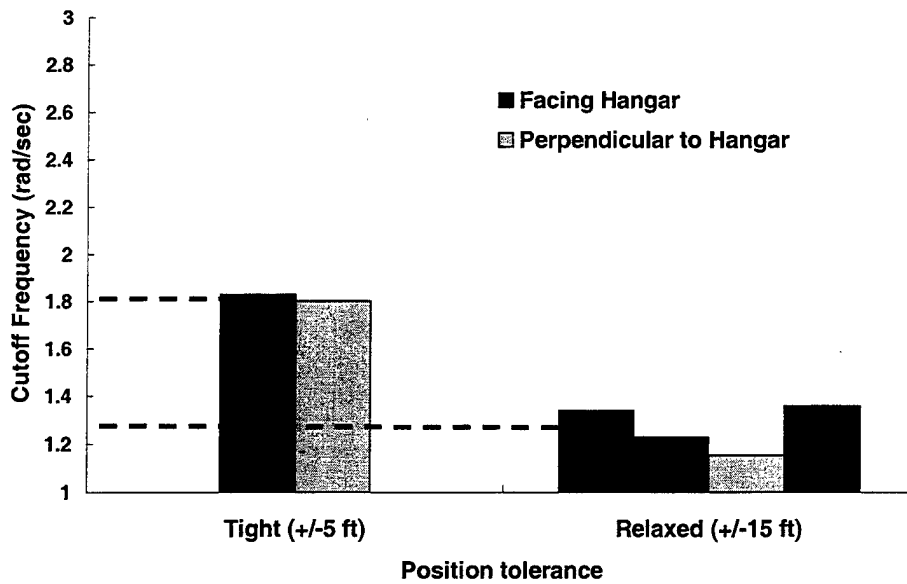


Figure 11. Longitudinal cutoff frequency differences for strong turbulence – Flight 1.

2. Lateral

Lateral mixer input cutoff frequencies for both hover position tolerances are also presented for both flight data sets, Figure 12 and Figure 13. At the stronger level of turbulence, Figure 12, the average lateral cutoff frequency required to maintain a tight position tolerance was 3.6 rad/sec and for basic stabilization (relaxed tolerance) an average cutoff frequency of 2.1 rad/sec was required. At the lower level of turbulence, Figure 13, the lateral cutoff frequency required to maintain a tight position tolerance was 3.1 rad/sec and for basic stabilization a cutoff frequency of 1.4 rad/sec was required. Similar to the longitudinal results, the same level of control cutoff frequency resulted whether the aircraft was facing the hangar or perpendicular to the hangar.

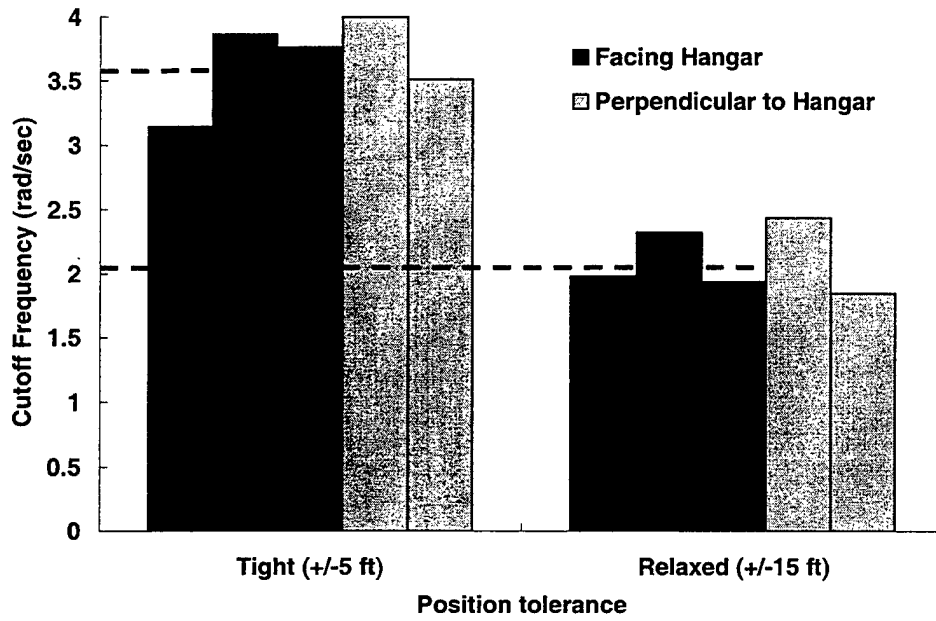


Figure 12. Lateral cutoff frequency differences for strong turbulence – Flight 2.

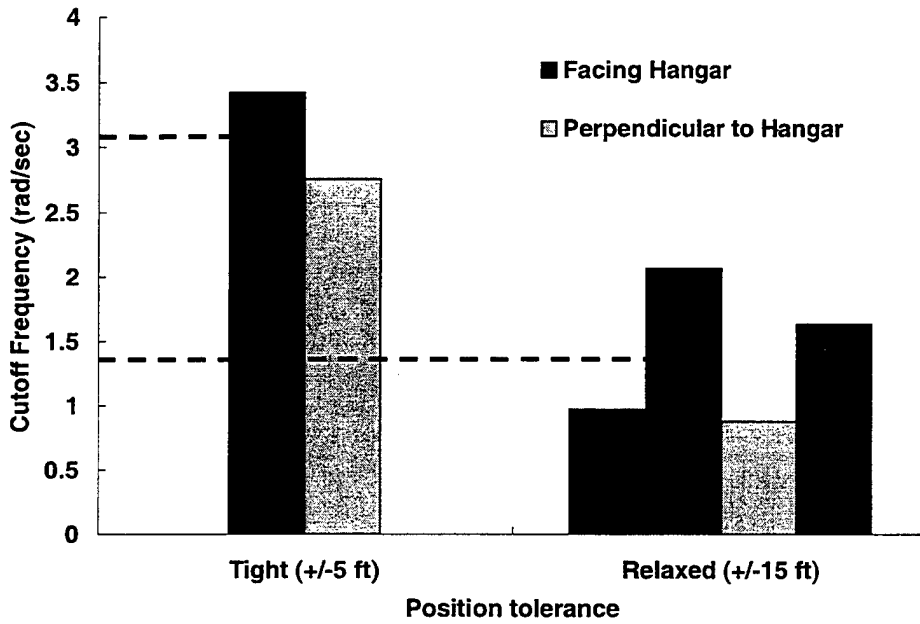


Figure 13. Lateral cutoff frequency differences for strong turbulence – Flight 1.

3. Directional

Directional mixer input cutoff frequencies for both hover position tolerances are presented for both flight data sets, Figure 14 and Figure 15. The effects of turbulence on precision hover are much less pronounced in yaw than in the pitch and roll axes. At the stronger level of turbulence, Figure 14, the average directional cutoff frequency required to maintain a tight position tolerance was 1.6 rad/sec and for basic stabilization (relaxed tolerance) an average cutoff frequency of 1.2 rad/sec was required. At the lower level of turbulence, Figure 15, the directional cutoff frequency required to maintain a tight position tolerance was 0.85 rad/sec and for basic stabilization a cutoff frequency of 0.77 rad/sec was required. It appears that directional control is independent of aircraft orientation and also nearly independent of position tolerance. The level of control activity required for tight tolerance versus relaxed tolerance is nearly the same, which indicates that the pilot is concentrating the workload in the longitudinal and lateral axes. For increasing levels of turbulence we do see an increase in the cutoff frequency required to maintain either tight or relaxed position tolerance. Because the yaw response to disturbances is less than the pitch and roll response, further analysis was concentrated on the pitch and roll axes.

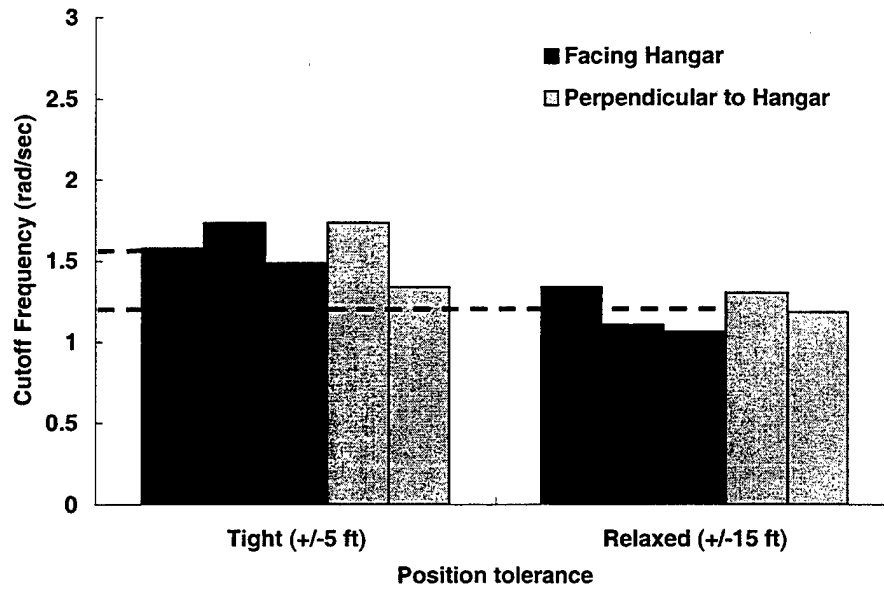


Figure 14. Directional cutoff frequency differences for strong turbulence – Flight 2.

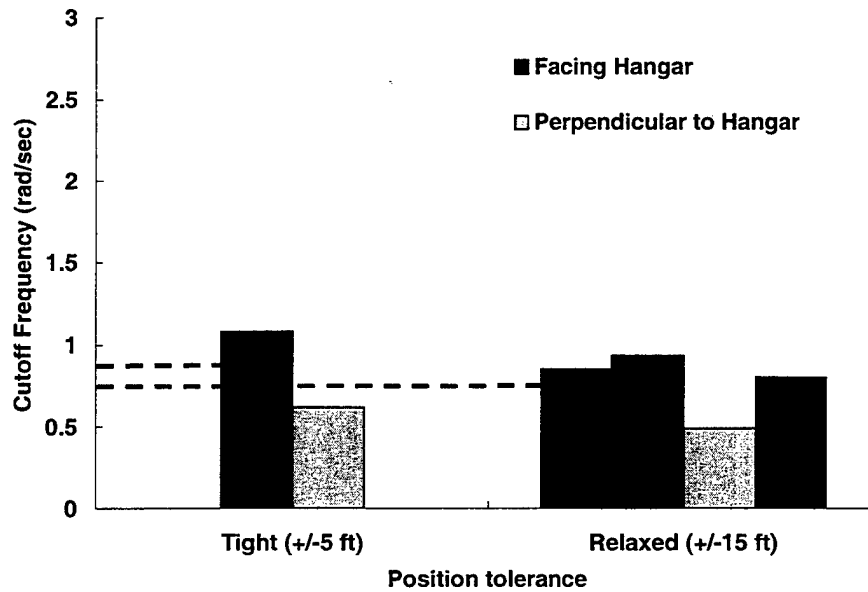


Figure 15. Directional cutoff frequency differences for strong turbulence – Flight 1.

B. FLIGHT CONTROL REQUIREMENTS

The cutoff frequency analysis showed that as the level of turbulence increases, we also see an increase in the flight control cutoff frequency. These results confirm the assumption that with an increasing magnitude of disturbance response, an increasing pilot compensation level is required to achieve desired task performance [Ref. 1]. Assuming the flight control system has satisfactory handling qualities in a disturbance-free environment, these results indicate that to meet desired performance in a turbulent environment additional design criteria must be enforced. Currently, the design criteria for gust response in ADS-33D-PRF [Ref. 2] is tied to the pilot bandwidth thresholds established for control response and are independent of turbulence level. Bandwidth is defined as the lesser of $\omega_{BW_{gain}}$ and $\omega_{BW_{phase}}$ as indicated in the frequency response plot, Figure 16.

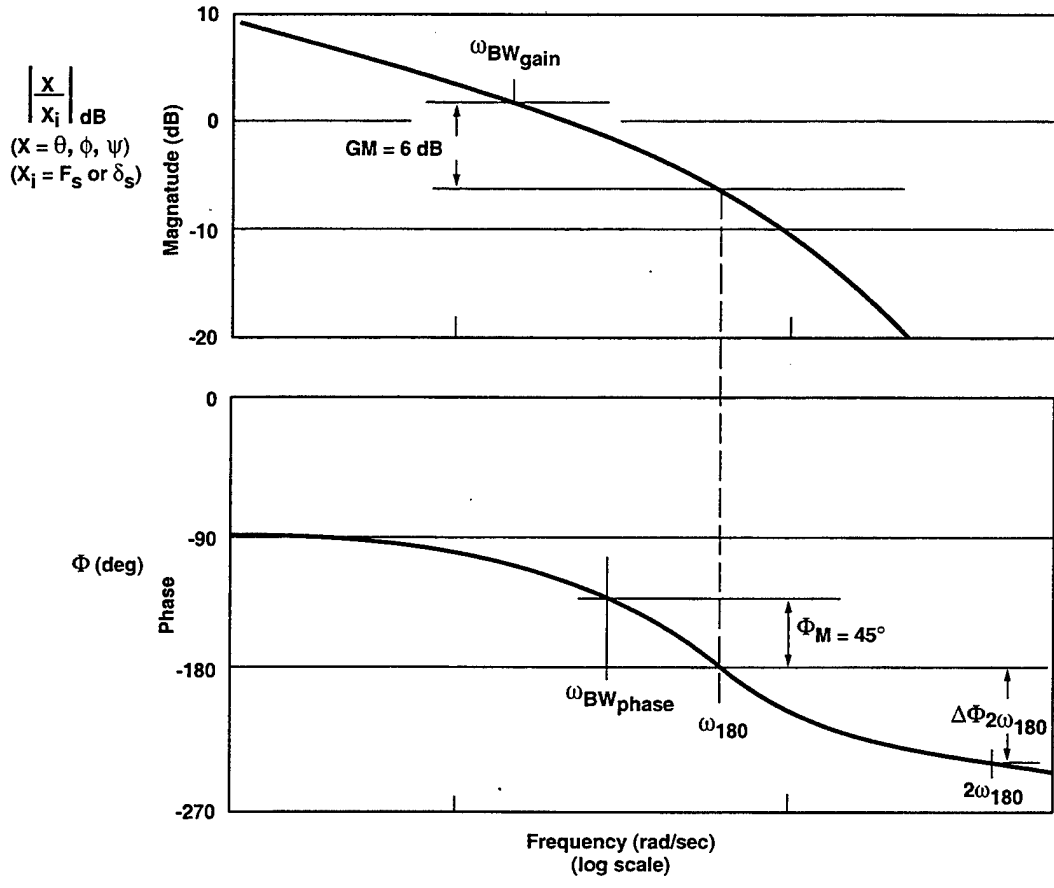


Figure 16. Definition of bandwidth.

Example plots for hover and low speed small amplitude pitch attitude change requirements are included in Figure 17 and Figure 18. It appears from the cutoff frequency results that to adequately establish a design requirement for gust response, the control bandwidth threshold must be set as a function of both task tolerance and disturbance level.

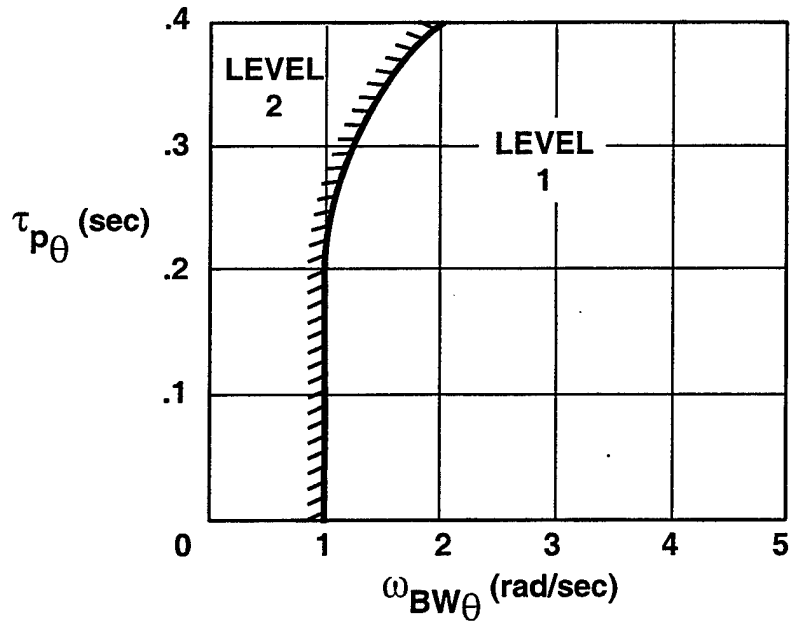


Figure 17. ADS-33D small amplitude pitch attitude requirement - hover and low speed.
From Ref. [2].

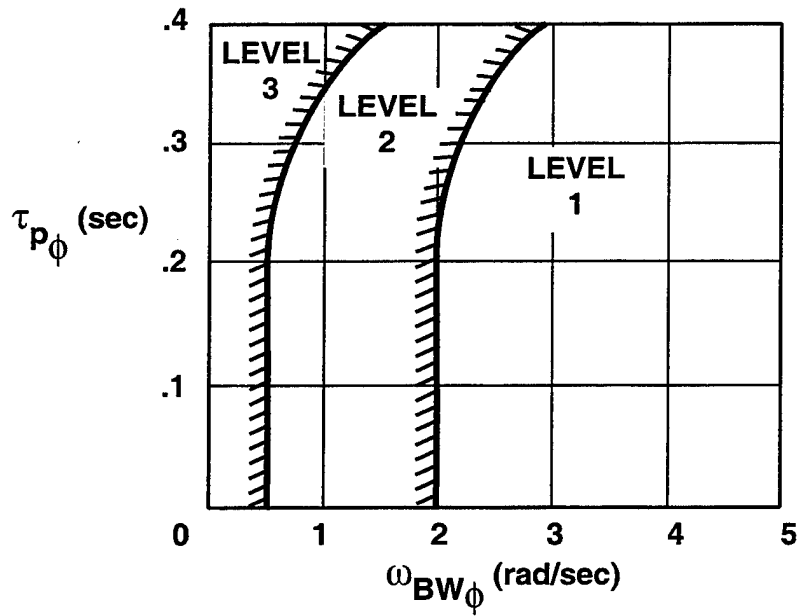


Figure 18. ADS-33D small amplitude roll attitude requirement - hover and low speed.
From Ref. [2].

ADS-33 disturbance rejection requirements state that pitch and roll responses to inputs directly into the control surface actuators shall meet the bandwidth threshold limits based on aircraft response to cockpit control inputs [Ref. 2]. Figure 17 and Figure 18 present the cockpit control input bandwidth thresholds that include the delays and lags in the system between the pilot input and the aircraft response. It appears that the disturbance inputs at the actuator would naturally have higher bandwidth values since the inputs are directly to the actuator and bypass filtering and delays in the processing of pilot stick inputs.

An effort to map the bandwidth to control system crossover frequency was attempted using CONDUIT (Control Designer's Unified Interface) [Ref. 13]. CONDUIT is a state-of-the-art computational tool for aircraft flight control design, evaluation, and integration for modern fixed-wing and rotary-wing aircraft. This correlation was obtained by tuning the flight control system cutoff frequency of the Army/NASA UH-60 Rotorcraft/Aircrew Systems Concepts Airborne Laboratory (RASCAL) model to equivalent cutoff frequency values from the gust response flight test and determining resulting control system bandwidth. RASCAL is a UH-60 Black Hawk helicopter that has been modified by NASA and the US Army for flight systems research. The RASCAL control system is presented in Figure 19, where $M(s)$ contains stick filtering and desired dynamics, $P(s)$ is a model of aircraft dynamics, $\hat{P}^{-1}(s)$ is the approximate inverse of aircraft dynamics, and $H(s)$ contains the feedback dynamics. From this control system schematic, bandwidth for handling-qualities response is obtained from $\theta(s)/\delta_s(s)$, bandwidth for disturbance response is obtained from $\theta(s)/\delta_d(s)$, and crossover frequency

is obtained from $f(s)/e(s)$. A one-to-one mapping of control bandwidth to disturbance crossover frequency was obtained from CONDUIT and is presented in Figure 20.

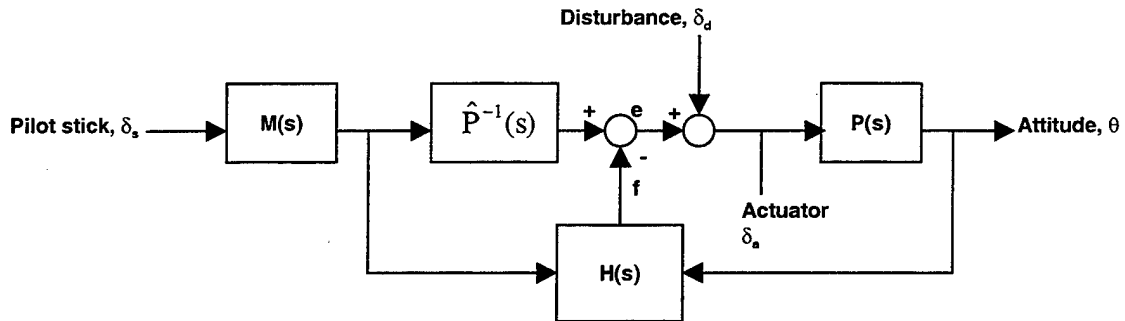


Figure 19. RASCAL model following control system.

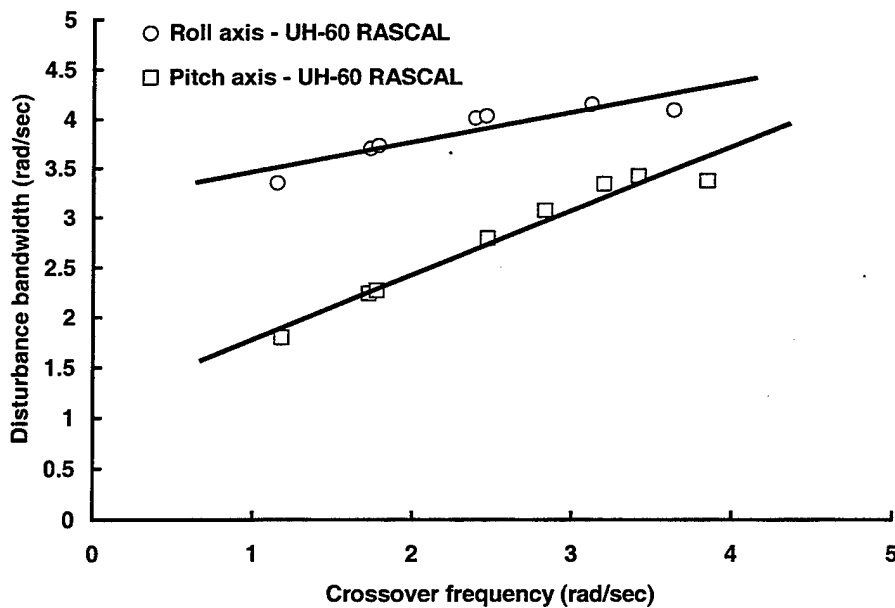


Figure 20. UH-60 disturbance bandwidth vs. control crossover frequency.

To understand the significance of these results we look at where the flight test cutoff frequency values fall on the plot. To maintain desired performance at the tight tolerance in the stronger wind condition, a longitudinal cutoff frequency of 2.5 rad/sec was required and a lateral cutoff frequency of 3.6 rad/sec was required. These cutoff

frequency values correspond to (-135°) gust bandwidth values of approximately 2.7 rad/sec for pitch and 4.5 rad/sec for roll. The significant difference between the cutoff and bandwidth frequencies is simply a result of their definitions. For a typical feedback system equalized to a K/s broken-loop response, the cutoff frequency corresponds to the (-3dB) magnitude or the (-45°) phase point in the closed-loop response. The bandwidth is defined as (-135°) phase point in the same closed-loop response, which will naturally occur at a higher frequency. It should be noted that handling qualities rating scale values were not assigned for the hover task performed and this correlation of our hover flight test results to the level 1 ADS-33 boundary is an approximation.

The pitch response bandwidth value of 2.7 rad/sec and the roll bandwidth value of 4.5 rad/sec appear significantly higher than the current Level 1 boundaries and indicate that the Level 1 gust response bandwidth requirements should be more stringent. As suggested earlier in this section, these results are not surprising considering that the RASCAL control system parameters are tuned so that the end-to-end handling-qualities response (θ/δ_σ) will meet the ADS-33D bandwidth requirements. Since the disturbance response is then obtained for inputs directly into the actuator (θ/δ_δ) we would expect this calculation to yield much higher bandwidths. Additional data on other aircraft and a broad range of conditions are needed to determine a generic boundary, but these results are useful for the Black Hawk class of utility vehicles.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. GUST MODEL IDENTIFICATION METHOD/DATA ANALYSIS

A. OVERVIEW

The hover task performed on the leeward side of the Coast Guard aircraft hangar took advantage of the turbulence generated by the predominant winds across this blunt body (Figure 8). The aircraft response to the disturbed air was captured via the aircraft instrumentation package for development of a gust model. Multiple approaches to processing the flight data were taken over the course of this research project. The main objective throughout remained obtaining an accurate method to extract the aircraft mixer input required to simulate an aircraft response to naturally occurring gusts. The question was how to take the available flight test data, which includes aircraft control positions, accelerations, and rates, and reverse-engineer the process to be left with a control input that can be fed into a ground or in-flight simulation model to replicate the gust response of the aircraft.

The end product of this portion of the research is a gust control transfer function model developed from flight test data that captures the control input spectrum to simulate the aircraft gust response. The utilization of this model consists of inputting a white noise signal into the gust excitation model such that the output signal has the spectrum of the desired control input, Figure 21.

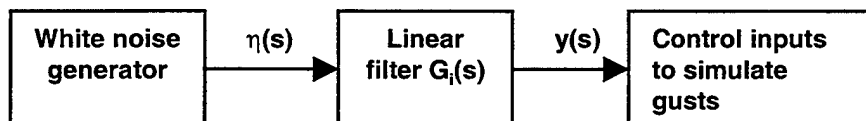


Figure 21. Utilization of gust excitation model.

A similar method to that used by NRC, Canada in their disturbance rejection flight test effort was employed [Ref. 1]. In order to simulate a consistent level of disturbance in the flight test program with their variable stability Bell 205, data from a flight in heavy turbulence (in the lee of a larger building in strong winds) was used to determine a remnant aircraft response due to turbulence. This data was processed into a simple, first order inverse model of the aircraft to create equivalent actuator data traces which could then be processed into the aircraft to create a gust response.

B. AIRCRAFT MODEL DETERMINATION

The first step in the modeling process was to extract the remnant aircraft rates that were caused by the atmospheric disturbances. This required determination of which aircraft model should be used in the data processing. Application of the aircraft model occurs in the forward route when the flight test control inputs are processed through the model to obtain the pure aircraft response that would be expected in the absence of atmospheric disturbances, Figure 22. Once an assumed aircraft response is available from the control inputs this response (aircraft rates) is subtracted from the actual flight test data aircraft rates to obtain a remnant that is caused by the atmospheric disturbances present during the flight test. The remnant rates were then filtered using an elliptical bandpass filter between 1.0-10 rad/sec to reduce the effect of low frequency drift in the open-loop integration and the high frequency measurement noise. The filtering process concentrates the analysis on the frequency range of interest for piloted handling qualities.

An inverse of the aircraft model must be available for processing the remnant rates (gust-induced) to obtain a resultant aircraft control input (mixer) that causes the aircraft response to disturbances.

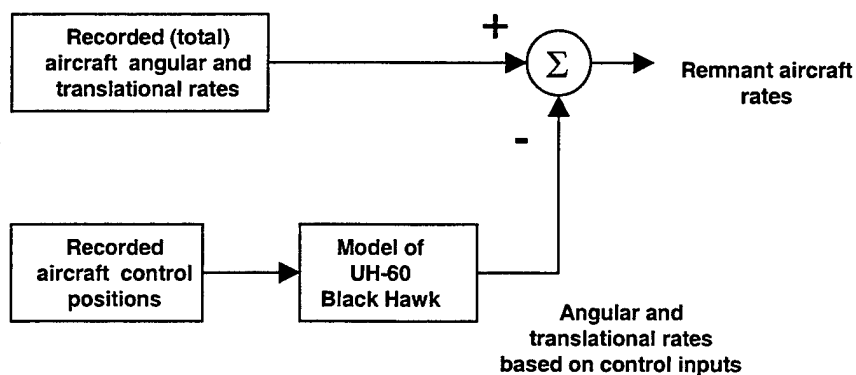


Figure 22. Computation of remnant aircraft rate.

The goal was to look at the pure aircraft response without the influence of the stability augmentation system (SAS). In order to do this the flight test data was collected by flying SAS-off maneuvers and the aircraft models were employed with inputs of aircraft mixer (downstream of the SAS) and output of aircraft rates. Various models were examined in the process of the research and these are described below.

1. Gen Hel Advanced Rotor/Helicopter Model

The Sikorsky-Ames Gen Hel non-linear mathematical model of the UH-60 Black Hawk was the first model considered for use in this research effort. This model was developed under contract for the U.S. Army and NASA by Sikorsky Aircraft. The model was based on the Sikorsky General helicopter Flight Dynamics Simulation, and was

intended to provide an engineering simulation suitable for performance and handling qualities.

The model represents the helicopter by a six degree-of-freedom rigid body, including rotor blade flapping, lagging, air mass, and hub rotational degrees of freedom. The simulation is comprised of program modules representing the major helicopter, with a detailed interface of the physical quantities such as forces, moments, attitudes, and velocities shared between the modules.

The flight control system modeled in Gen Hel matches the physical configuration of the UH-60A Black hawk flight control system, enabling direct utilization with flight test data from the test aircraft. Successful processing of flight test control inputs through the Gen Hel model generated aircraft response rates in the absence of any turbulence. The drawback to utilizing the Gen Hel model in our simulation was that there was no way to invert the model for use in generating control inputs that cause the gust response. In the interest of maintaining consistency of models and reducing modeling error throughout the process, the Gen Hel model was abandoned and other models were pursued.

2. On-Axis Transfer Function Model

In an effort to find a model that could easily be used in both the forward and inverse routes of the data analysis, an on-axis transfer function model was examined. Frequency response data from previous flight test of the aircraft used in this research effort was analyzed via CIPHER[®] in order to obtain the most accurate on-axis transfer function models. The NAVFIT feature of CIPHER[®] was utilized and simple, low-order

transfer functions for the four aircraft control axes were extracted. Accuracy checks of this model were only possible in the on-axis response and these were conducted by hitting the model with flight control doublet inputs and comparing the response to flight test data. Figure 23 presents a comparison of the response of the aircraft and model to a pitch doublet, and Figure 24 presents the response to a roll doublet. The scripts that executed this process are PITCHCHECKER.M and ROLLCHECKER.M, in Appendix B. Satisfactory on-axis results were obtained but the lack of off-axis coupling effects was of concern.

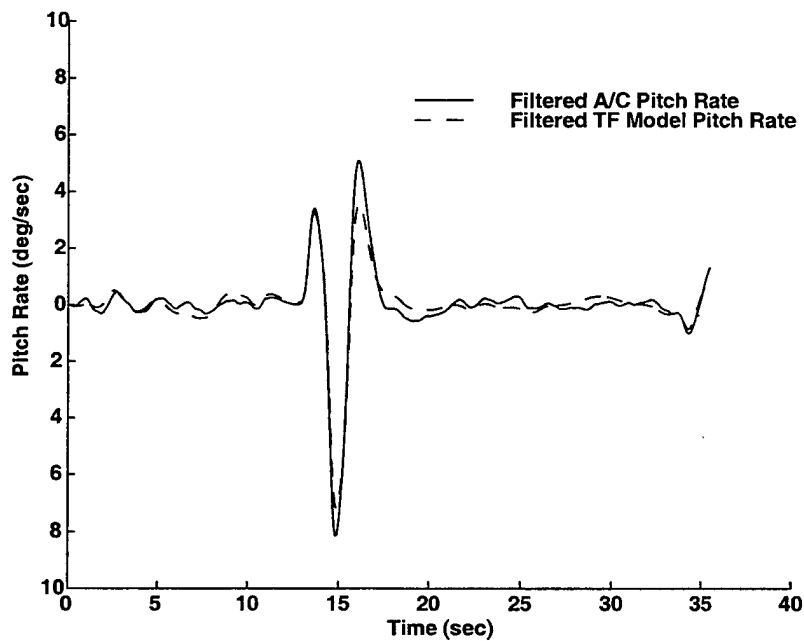


Figure 23. Pitch doublet response of UH-60A and on-axis pitch transfer function model.

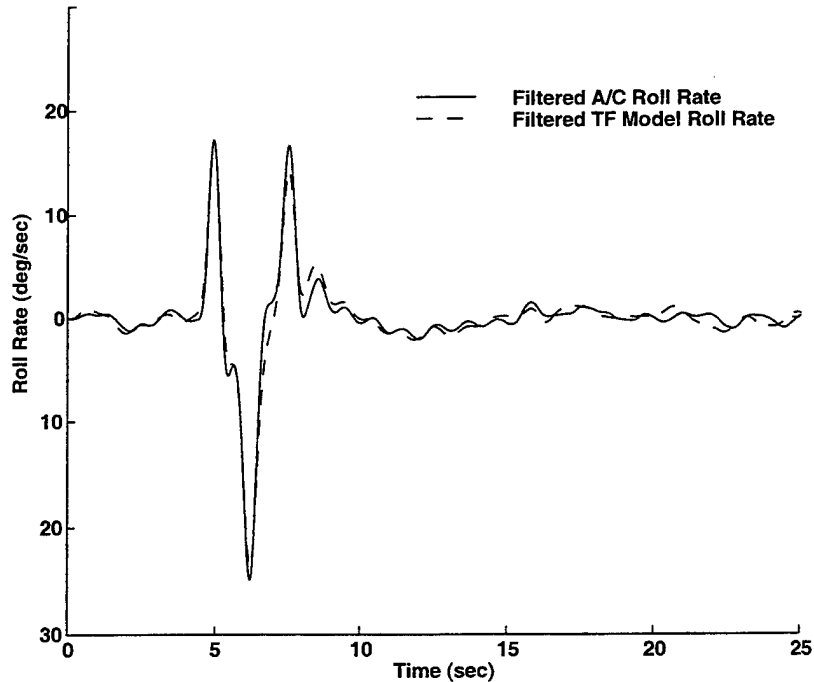


Figure 24. Roll doublet response of UH-60A and on-axis roll transfer function model.

This model enabled ease of inverting the model for use in the process of converting remnant aircraft rates due to gust back into control mixer inputs for application in simulation. MATLAB[®] scripts SSTF.M and INVCHECKTF.M, in Appendix B, were used in assessing the usefulness of the on-axis transfer function models in the turbulence modeling process. The drawback to this model was that any off axis effects were not captured thus reducing the overall accuracy of the aircraft response. The error introduced by this modeling technique was deemed improvable for the final effort, but this initial format enabled researchers to develop the overall process and evaluate the feasibility of the gust modeling project.

3. Identified Coupled Model

A fourteen degree of freedom multiple-input-multiple-output (MIMO) identified model was considered to be the most accurate model available [Ref. 11]. This state space model accurately characterizes the open loop UH-60 flight dynamics in hover and was identified from flight test data gathered on the Army/NASA RASCAL UH-60 and the NASA Blade Loads UH-60 helicopter using CIFER[®]. The model includes fuselage dynamics, vertical inflow, rotor RPM, and engine/governor dynamics. Horizontal and lateral translational degrees of freedom were fixed, and thus eliminated in the model to remove low frequency drift of the aircraft. The model was verified using pitch, roll and yaw control doublet flight test data with satisfactory matching of actual response to simulated response. A comparison of the model response and aircraft response to a roll doublet is presented in Figure 25.

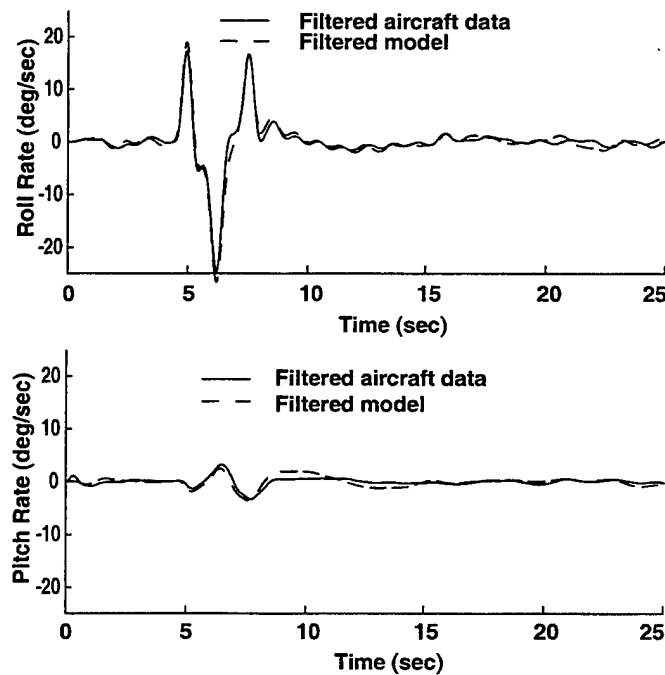


Figure 25. Roll doublet response of UH-60A and 14-DOF model.

4. Comparison of Various UH-60 Models

A check of the on-axis frequency response characteristics of the various helicopter models was conducted. This ensured that the modifications to the models did not compromise the frequency spectrum content of the model response. Bode plots comparing the response of the various models in pitch and roll are presented as Figure 26 and Figure 27 respectively. It can be seen that in the frequency range of interest, 1.0-10 rad/sec, the models show excellent matching to the flight data frequency response.

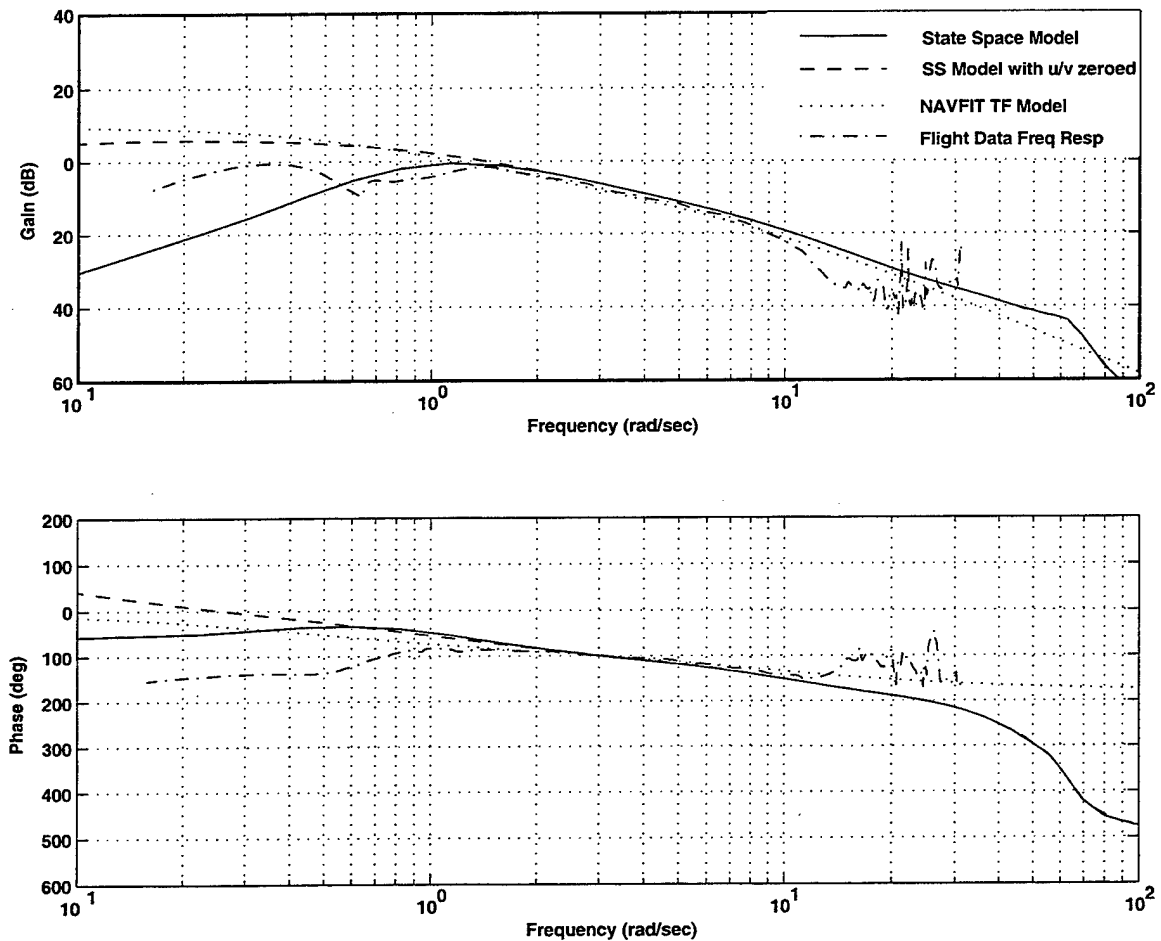


Figure 26. Comparison of pitch rate to longitudinal mixer frequency response characteristics of various models.

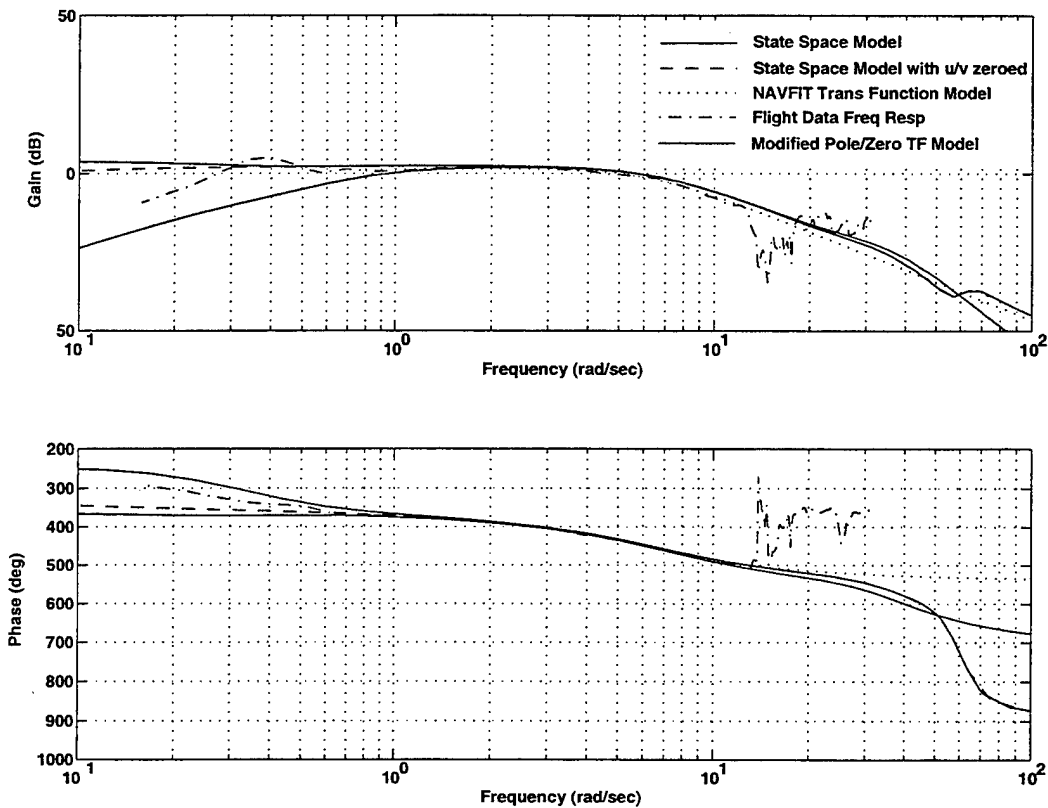


Figure 27. Comparison of roll rate to lateral mixer frequency response characteristics of various models.

C. DETERMINATION OF REMNANT AIRCRAFT RATES

A SIMULINK[®] model was formulated to perform the process of extracting the remnant aircraft rates due to the gusts, Figure 28. The MATLAB[®] code SSSS.M, Appendix B, executed the process. The aircraft mixer inputs are played into the simulation and converted to cockpit control inputs via an inverse mechanical linkage matrix. The cockpit control position time histories are the input to the UH-60 model and the resulting output is aircraft angular rates modeled in the absence of any atmospheric disturbances. These rates are then subtracted from the recorded aircraft angular rates

from the flight test data. The resulting remnant is the aircraft angular rate due to the atmospheric disturbance (gust) on the test day. The remnant rates were then filtered using an elliptical bandpass filter between 1.0-10 rad/sec to reduce the effects of low frequency drift in the open-loop time integration and the effects of high frequency measurement noise. The filtering process concentrates the analysis on the frequency range of interest for piloted handling qualities. An inverse of the aircraft model must be available for processing the remnant rates (gust-induced) to obtain a resultant aircraft control input (mixer) that causes the aircraft response to disturbances.

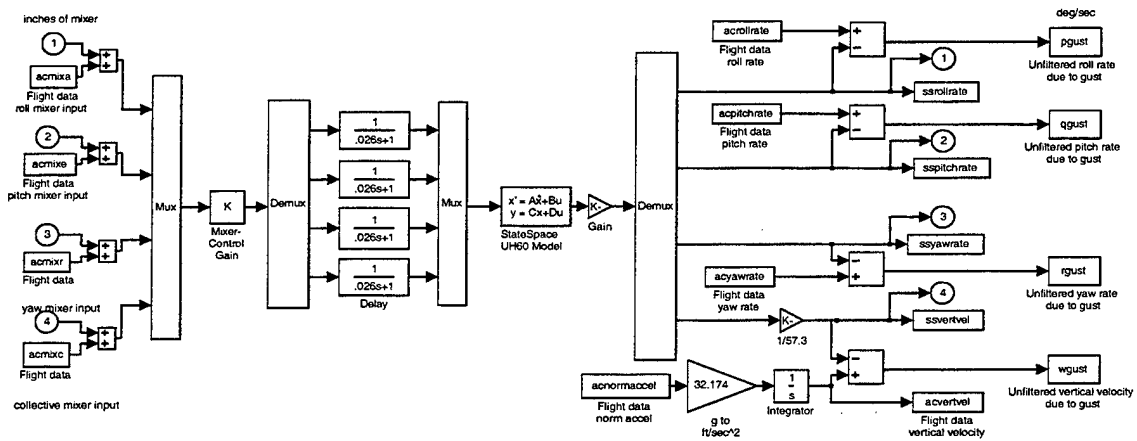


Figure 28. Simulink[®] model of differencing process with state-space model.

D. INVERSE MODEL

To obtain the control positions that cause the remnant aircraft rate (response to atmospheric disturbance), the remnant rates are fed through an inverse of the aircraft model used in the forward process, Figure 29. To avoid divergence of the inverse simulation model when applying the gust response control time histories, the pole-zero

map of the model was modified to include only response between 0.2 and 50 rad/sec, thereby retaining all dynamics which affect the response in the frequency range of interest (1.0-10 rad/sec). MATLAB script PZMOD.M in Appendix B executes this function. Having confirmed the accuracy of the aircraft model being used in the forward process, a similarly accurate inverse model was desired. The complexity of inverting the fully coupled 14-DOF model was deemed beyond the current scope of this project. Therefore, two options were addressed.

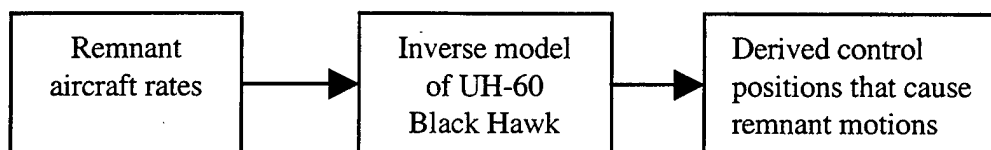


Figure 29. Derivation of disturbance control positions.

The first option was to use on-axis input-to-output transfer functions identified from a linearized model of the entire differencing process, Figure 28. The individual transfer functions for each axis were identified from the inputs to outputs of the entire model; i.e., roll mixer to roll rate transfer function was identified from input 1 to output 1 in Figure 28. The diagonal elements of the MIMO transfer function matrix could be inverted to enable determination of the mixer positions to generate the gust response. This option provided very accurate on-axis inverse dynamics but ignored coupling effects.

In an effort to capture any off-axis coupling effects in the inverse process, the second option investigated a quasi-steady inverse representation of the complete higher-order MIMO model. Starting with the six degree-of-freedom equations of motion, ignoring translational degrees-of-freedom, and assuming that the three angular responses

to control $(p/\delta_{lat}, q/\delta_{long}, r/\delta_{ped})$ and vertical response to control (a_z/δ_{coll}) follow a first order form in the frequency range of interest, then the 6 degree of freedom 8-state equations reduce to:

$$\dot{x} = Fx + Gu(t - \tau) \quad (6)$$

where:

$$x = \begin{bmatrix} p \\ q \\ r \\ w \end{bmatrix} \quad u = \begin{bmatrix} \delta_{lat} \\ \delta_{long} \\ \delta_{ped} \\ \delta_{coll} \end{bmatrix}$$

$$F = \begin{bmatrix} L_p & L_q & L_r & L_w \\ M_p & M_q & M_r & M_w \\ N_p & N_q & N_r & N_w \\ Z_p & Z_q & Z_r & Z_w \end{bmatrix}$$

$$G = \begin{bmatrix} L_{lat} & L_{long} & L_{ped} & L_{coll} \\ M_{lat} & M_{long} & M_{ped} & M_{coll} \\ N_{lat} & N_{long} & N_{ped} & N_{coll} \\ Z_{lat} & Z_{long} & Z_{ped} & Z_{coll} \end{bmatrix}$$

$$\tau = \begin{bmatrix} t_{lat} & t_{long} & t_{ped} & t_{coll} \end{bmatrix}$$

A de-coupled state-space model identification based on first order angular rate responses was run and then all of the off-axis responses were included to obtain a reasonable coupled solution. The elements of F, G, and Tau were identified from the MIMO solution of CIFER[®]. This model provided better coupling effects but less accuracy in the on-axis dynamics. Ideally, an exact numerical inverse of the coupled

MIMO model would be used. This nonlinear inverse solution is recommended for future efforts to improve the accuracy.

Modeling errors due to coupling effects and signal noise will obviously exist. Determination of which of inverse model to use was based on a check solution of the entire modeling process, Figure 30. The remnant aircraft rates were processed through each of the inverse models to obtain a mixer input representing the aircraft response to atmospheric disturbances. These remnant mixer inputs were then summed with the actual aircraft mixer inputs from the flight test data. This total mixer input was run through the complete high-order model to produce an aircraft response that could be compared to the original flight data aircraft response. MATLAB[®] scripts INVCHECKSS.M and INVCHECKSS_COUP.M in Appendix B executed the check solution for the extracted on-axis inverse model and the coupled inverse model respectively. The SIMULINK[®] model in Figure A-4 runs the simulation to generate the check solution aircraft rates.

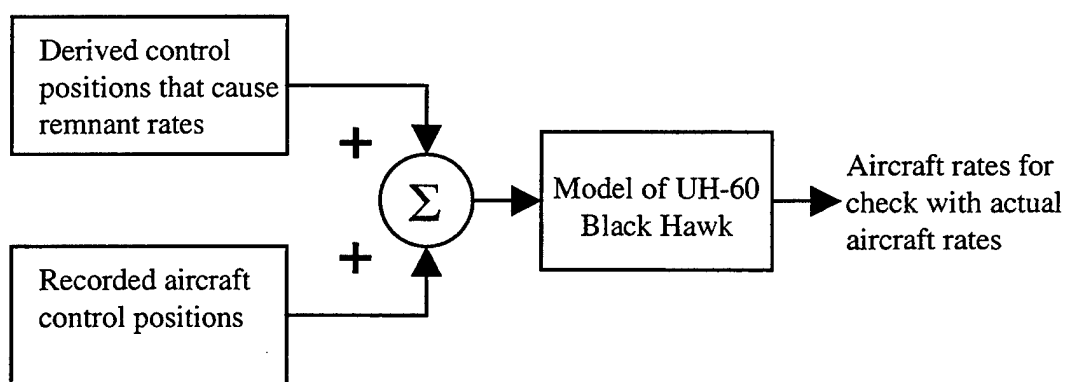


Figure 30. Check solution for inverse aircraft model options.

Figure 31 (a) presents a comparison of the flight test aircraft pitch rate to the pitch rate obtained by summing the actual control inputs with the remnant input calculated based on the on-axis inverse model, Figure A-2. Figure 31 (b) presents a comparison of the flight test aircraft pitch rate to the pitch rate obtained by summing the actual control inputs with the remnant input calculated based on the coupled inverse model, Figure A-3. Comparison of the residual difference between the check solution rates and the aircraft data rates is presented in Figure 31 (c) and indicates that the on-axis inverse model was more accurate overall than the coupled inverse model. This indicates that the accuracy of the on-axis response is of greater importance than the coupling effects in the inverse modeling process.

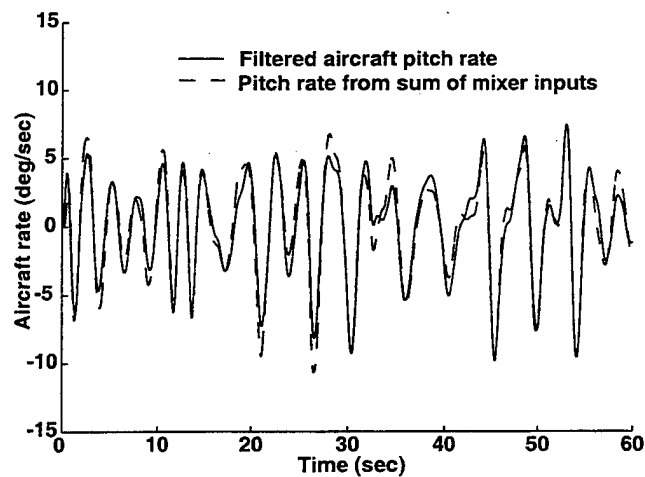


Figure 31. (a) On-axis extracted linear model

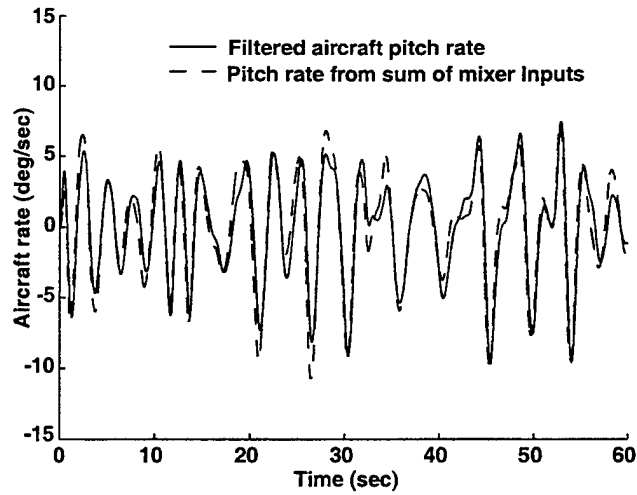


Figure 31. (b) Coupled inverse model

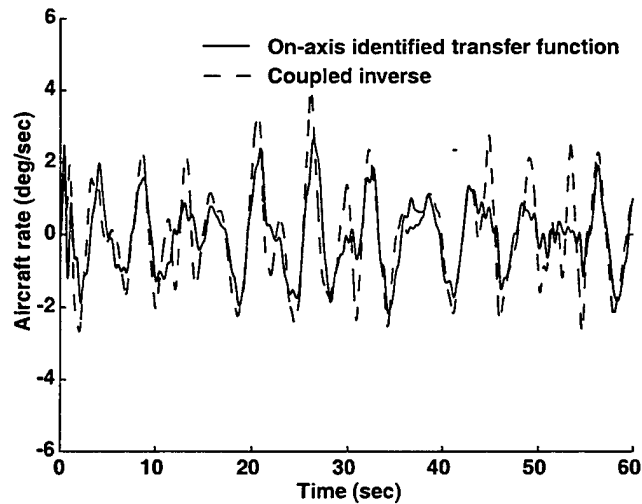


Figure 31. (c) Residual comparison between on-axis and coupled inverse models

Figure 31. Comparison of check solution with aircraft data for on-axis and coupled inverse models.

E. RESULTANT GUST MIXER INPUT

The resultant mixer inputs required to generate the aircraft response to turbulence were analyzed using the spectral analysis tools available in CIFER[®]. The autospectrum of this “gust mixer input” was examined for varying levels of pilot gain on flight events

from the same flight (same level of turbulence). This confirmed that the experimental results were not being contaminated by the pilot inputs or that the extracted gust response was not merely due to modeling errors. The plots of input autospectrum for these various conditions are shown in Figure 32. For PSD plots, CIPHER[®] adopts the convention of power decibels = $10\log G_{\delta\delta}$ which shows the frequency content of δ in dB, rather than the signal-squared [Ref. 9]. (So for example, a 6dB reduction in PSD corresponds to a 50% reduction in the magnitude of δ inputs.) The fact that the two levels of pilot gain produce the same gust PSD in Figure 32 confirms that the extracted gust response is independent of pilot control strategy, as expected.

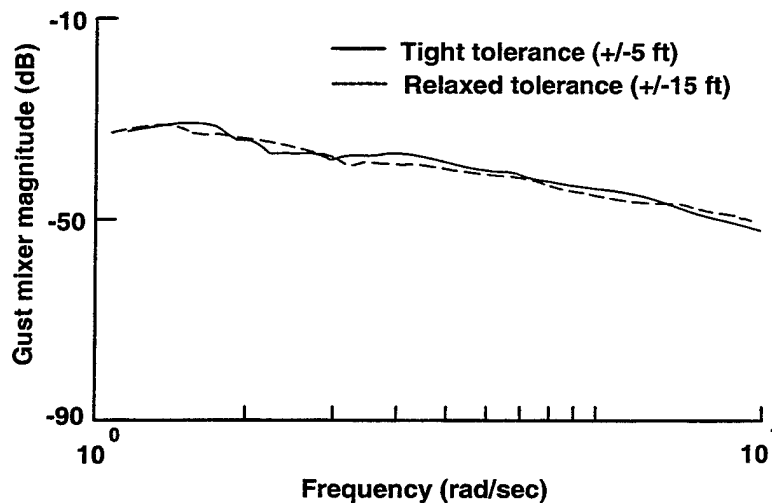


Figure 32. Longitudinal gust mixer autospectrum for varying pilot gain.

The coherence function was examined for the response of the gust remnant to pilot mixer input, as shown in Figure 33. The low coherence values show again that the gust remnant determined by the modeling process is uncorrelated with the input from the pilot. This ensures that the remnant being analyzed is the true gust response and not modeling error.

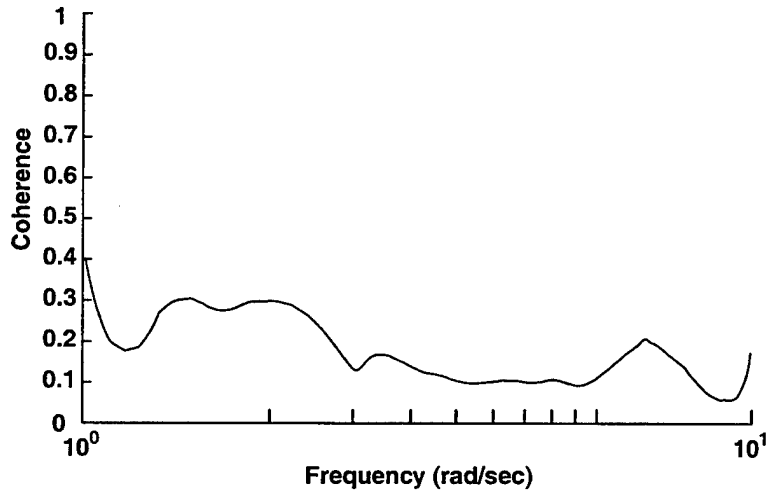


Figure 33. Coherence function for pilot mixer input to identified gust remnant.

When examined for varying levels of turbulence, the input autospectrum of the gust mixer inputs followed the typical behavior of the PSD function of atmospheric turbulence. A parallel shift in the magnitude of the PSD for different turbulence levels is evident, as shown in Figure 34.

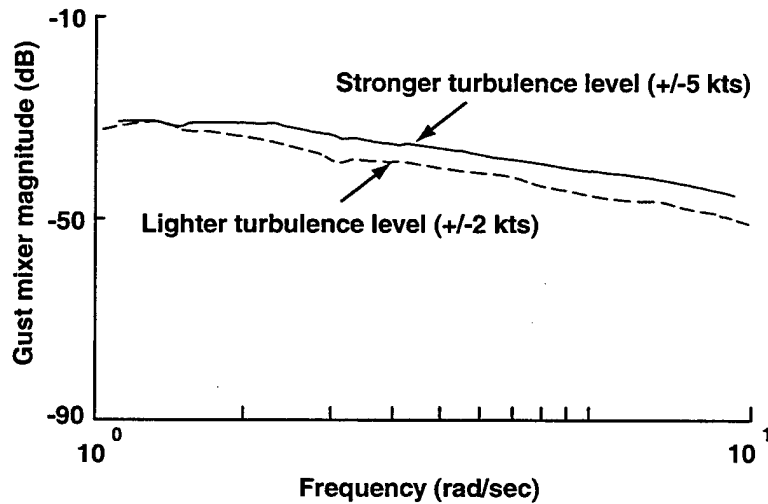


Figure 34. Longitudinal gust mixer autospectrum for varying turbulence levels.

It is interesting to note that regardless of aircraft orientation, the PSD level of longitudinal gust mixer required for each axis remained effectively the same, Figure 35. The same result was found for the lateral gust mixer PSDs. It is expected that the response of the rotor tip path plane to likely produce a rolling moment on the vehicle due to the spatial gradient of an upwash or vertical gust and a pitching moment due to a lateral gust [Ref. 5]. To a first approximation, the PSD results of the gust mixer inputs being independent of aircraft orientation likely indicates turbulence mixing in all directions. This is consistent with the highly turbulent recirculating flows in bluff body wakes, Figure 2. These results are also consistent with the cutoff frequency results that showed the level of pilot workload was independent of aircraft orientation.

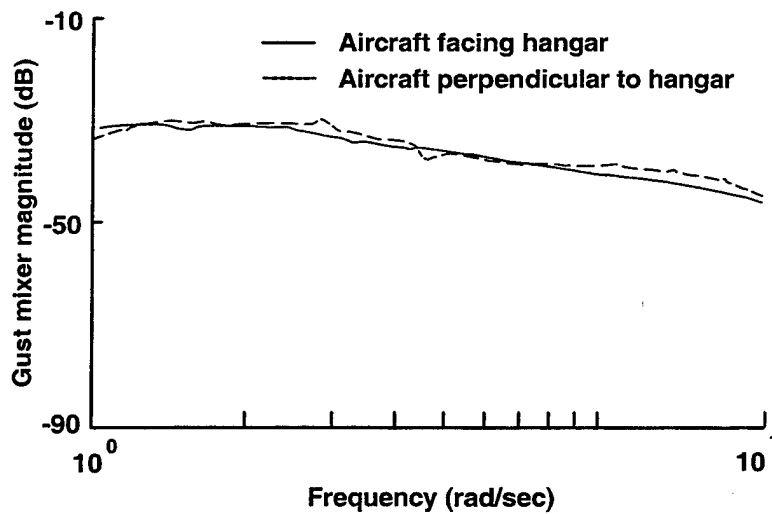


Figure 35. Longitudinal gust mixer autospectrum for varying aircraft heading orientation.

F. FORMATION OF GUST MODEL

The power spectral density for the vertical gust velocity of the von Karman turbulence model can be approximated by [Ref. 14]:

$$S_{w_g w_g}(\Omega) = \left(\frac{4\sigma_{w_g}^2 L_u}{\pi} \right) \frac{1}{4 + (L_u \Omega)^2} \quad (6)$$

for the PSD definition: $\sigma_{w_g}^2 = \int_0^\infty S_{w_g w_g}(\Omega) d\Omega$ as in reference [15], where Ω is the spatial frequency. The key model parameters are the vertical gust velocity rms turbulence intensity (σ_{w_g} , ft/sec) and scale length of longitudinal turbulence (L_u , ft).

Equation (6) can be re-written as:

$$S_{w_g w_g}(\omega) = \left(\frac{L_u \sigma_{w_g}^2}{U_0 \pi} \right) \left[\frac{1}{1 + \left(\frac{\omega}{\alpha_w} \right)^2} \right] \quad (7)$$

where:

ω is the observed angular frequency given by

$$\omega = \Omega U_0 \quad (8)$$

α_w is the PSD temporal break frequency, given from (7) by:

$$\alpha_w = \frac{2U_0}{L_u} \quad (9)$$

It is observed that this approximation is the same form as the classical Dryden spectral model for longitudinal turbulence, but with one-half the scale length, as also noted in reference [14]. These PSD functions characterize the mean squared value of the turbulence distribution with frequency. These specific forms are presented in terms of spatial frequency, Ω , which is the observed angular frequency normalized by a reference airspeed. Whereas the Dryden form normalizes by the equilibrium speed of the aircraft, Hess [Ref. 5] uses the speed of the frozen field convecting past the stationary aircraft, U_0 , in order to compensate for the low speed and hover condition of the helicopter. This research effort utilized a form similar to Hess where the PSD modeled is that of the control input rather than the components of turbulence velocities.

The gust transfer-function model $G_{w_g}(s)$ consistent with (7) that produces the gust velocity (w_g) when driven by a random noise signal is given by:

$$S_{w_g w_g}(\omega) = |G_{w_g}(\omega)|^2 S_{nn}(\omega) \quad (10)$$

where for white noise: $S_{nn}(\omega) = 1$.

The required transfer function determined from equation (7) is:

$$G_{w_g}(s) = 2\sigma_{w_g} \sqrt{\frac{U_0^2}{L_u}} \left(\frac{1}{s + \alpha_w} \right) \quad (11)$$

For our modeling technique, a gust-control transfer function, G_{δ_g} , is required that produces a gust-control PSD for a white noise input. This was obtained from matching

equation (11) with the empirically-extracted results for the gust mixer PSD (G_{δ_g}) to determine the required scale factor K_{δ_g} :

$$G_{\delta_g}(s) = K_{\delta_g} \left(2\sigma_{w_g} \sqrt{\frac{U_0}{L_u}} \right) \left(\frac{1}{s + \alpha_w} \right) \quad (12)$$

This is a first-order transfer function model of the form:

$$G_{\delta_g}(s) = \left(\frac{K}{s + \alpha_w} \right) \quad (13)$$

The modeling parameters (K, α_w) in eq. (13) were obtained from a best fit to the longitudinal and lateral gust mixer autospectrum data (e.g., Figure 34) and are shown in Table 3:

Table 3. Spectral filter fit coefficients for gust mixer model; $K/(s+\alpha_w)$.

Flight	U_0 (ft/sec)	σ_{w_g} (ft/sec)	K (mixer - in.)	α_w (rad/sec)	L_u (ft)
1(lat)	28.66	3.37	0.37	1.30	44.1
1(lon)	28.66	3.37	0.33	1.15	49.9
2(lat)	37.1	8.43	0.60	2.20	33.8
2(lon)	37.1	8.43	0.57	1.80	41.2

The turbulence scale length, L_u , was calculated based on the gust mixer PSD break frequency results, (eq. 9), and is reasonably consistent for all four cases. The average value of $L_u = 42.2$ feet is approximately equal to both the hovering height above the ground during the flight tests and rotor diameter. This is consistent with the modeling results of Costello, et al. [Ref. 4].

It is interesting to note that the transfer-function fits of the empirical gust mixer PSD for the longitudinal and lateral data are nearly the same, reducing the model dependency to wind speed, gust RMS, and a single average scale length. A possible explanation follows. Since the longitudinal crossover frequency is the same for nose into the wind or nose perpendicular to the wind, it is surmised that the character of the aerodynamic moments due to turbulence is uniform for pitch independent of aircraft orientation. This same logic applies to the lateral axis. To a first approximation, the applied gust aerodynamic moments are the same for pitch and roll and are independent of aircraft orientation. Since the extracted pitch and roll gust mixer inputs are essentially simulated gust aerodynamic moments, they should be the same. Since the aircraft response is just the applied aerodynamic moment normalized by the associated inertias, the effects of inertia on the response to turbulence are the same as the effects of the inertia differences on the response to control inputs. Based on these findings we can adopt a single model (eq. 12) for lateral and longitudinal gust inputs.

The required scale factor of eq. (12) was obtained by adopting the average scale length ($L_u = 42.2$ ft) and then fitting the empirical results for K in Table 3:

$$K_{\delta_g} = 0.226\sigma_{w_g}^{-0.581} \quad (14)$$

The final gust-control transfer function model for pitch and roll gust inputs is then:

$$G_{\delta_g}(s) = 0.452\sigma_{w_g}^{0.420} \sqrt{\frac{U_0}{\pi L_u}} \left(\frac{1}{s + \alpha_w} \right) \quad (15)$$

The output of this transfer function model is inches of mixer input. The numerical results from this model (with the average L_u value) match the flight results of Table 3 very closely.

G. CHECK OF MODELING PROCESS

A check of the gust control transfer function model (15) was conducted by executing the white noise simulation in SIMULINK[®] and plotting the gust mixer PSD derived from the processed flight test data with the gust mixer input from simulation, shown in Figure 36 and Figure 37. These models show good fits over the desired frequency range of 1.0-10 rad/sec. The gust model form follows the theoretical low-pass result, which has a constant value at low frequency. The coefficients of the fits for both flight conditions are presented in Table 3. A comparison of the gust control transfer function models for the two levels of turbulence investigated is presented in Figure 38.

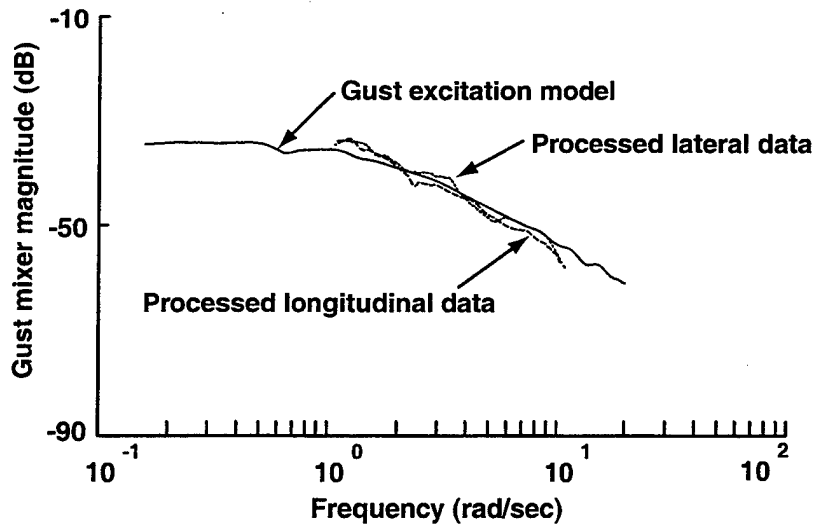


Figure 36. Gust excitation model for flight 1 (light turbulence).

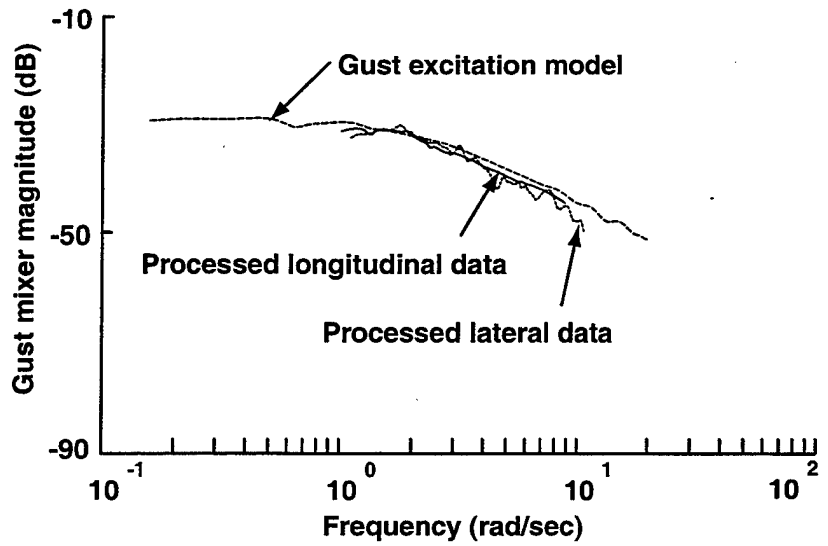


Figure 37. Gust excitation model for flight 2 (strong turbulence).

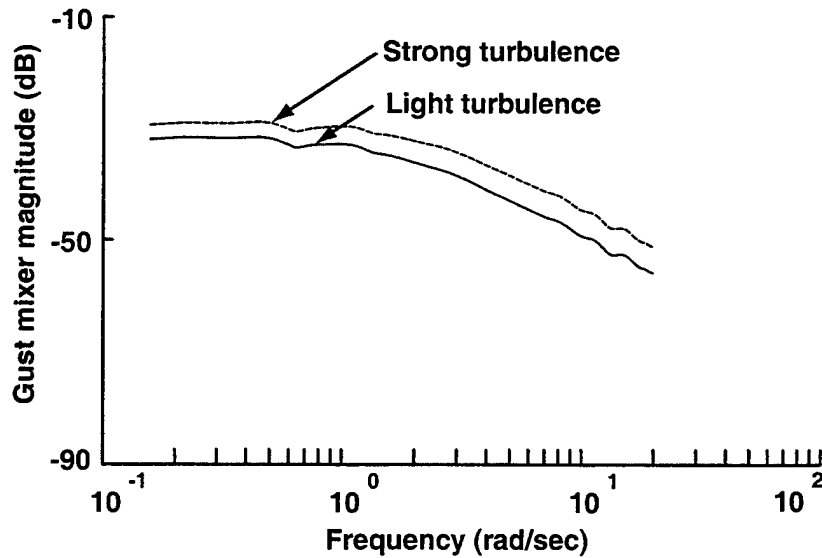


Figure 38. Gust excitation model characteristics for varying levels of turbulence.

As a final check of the overall modeling process, a simulation model was run utilizing white noise through the linear filters for both lateral and longitudinal gust mixer with the resulting mixer input fed through the aircraft model, Figure A-5. The resulting gust remnant PSDs are plotted versus the remnants determined from the initial differencing process in Figure 39 and Figure 40. The resulting PSDs overlay well in the region of interest, reinforcing confidence in the modeling scheme.

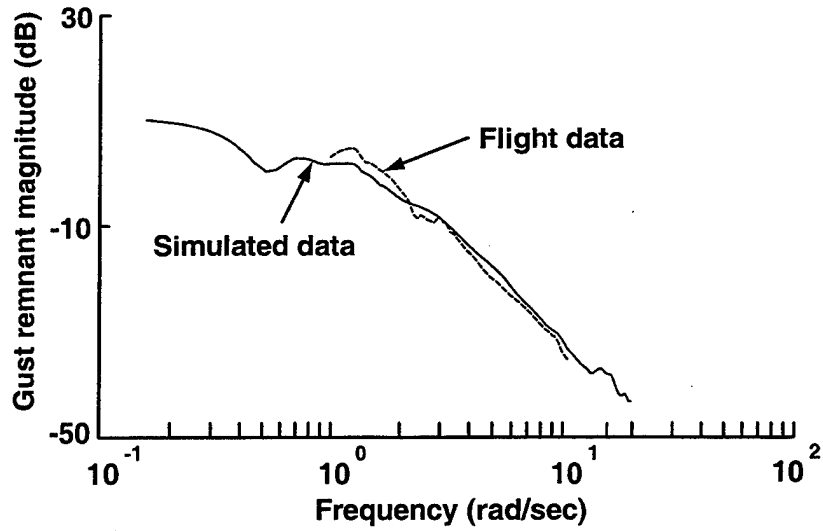


Figure 39. Comparison of longitudinal gust remnants for light turbulence (flight 1).

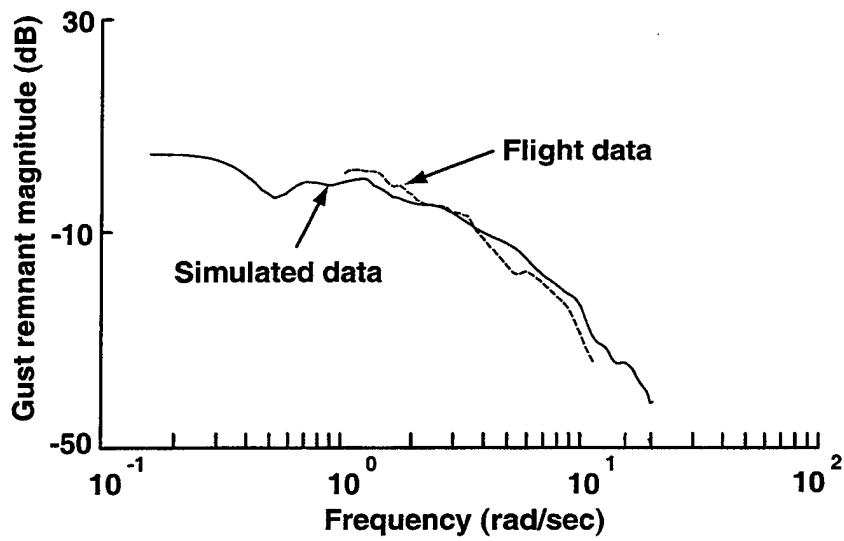


Figure 40. Comparison of lateral gust remnants for strong turbulence (flight 2).

H. UTILIZATION OF GUST MODEL

To utilize the turbulence model representation in simulation, a white noise source is used as an input signal into the gust excitation filter, which produces an output signal

(mixer input inches) that has the PSD of the measured result, for example in Figure 36. (For reference back to equivalent pilot stick input: 1 inch lateral mixer input=0.2425 inch pilot lateral stick input; 1 inch longitudinal mixer input=0.2155 inch pilot longitudinal stick input.) The noise source utilized herein was a random number generator with a mean of zero and a variance of 1. Future working validation of this model via a piloted simulation is recommended.

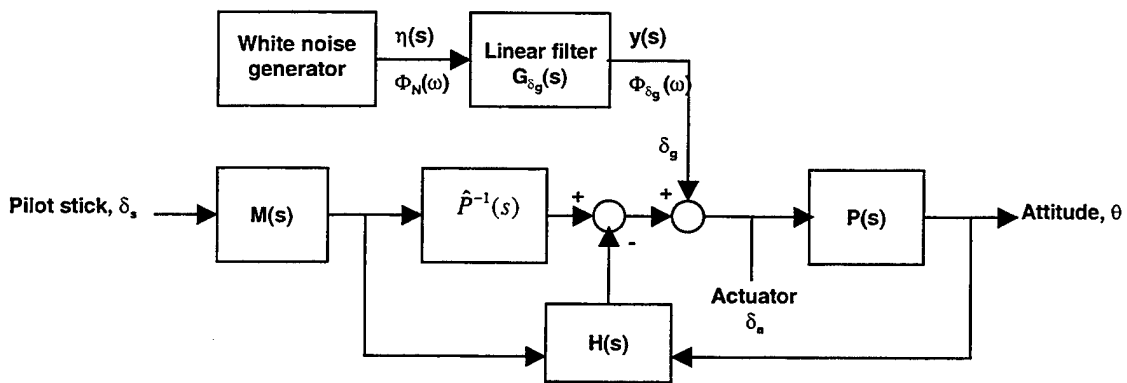


Figure 41. Example gust model implementation in RASCAL analysis.

THIS PAGE INTENTIONALLY LEFT BLANK

VII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUDING REMARKS

A study into the use of a low speed turbulence model in determining the effects of atmospheric disturbances on helicopter low speed handling qualities has been conducted. A low speed turbulence model in the form of a scalable transfer function model of mixer input required to simulate aircraft response to disturbances has been developed. This simple model, obtained from actual flight test data, can be easily employed in a ground or in-flight simulator by summing with the pilot mixer input to give an accurate aircraft response to simulated turbulence. Specific conclusions are:

1. First principle approach to parametrically modeling aircraft gust response via control position can successfully be employed.
2. Gust mixer model is scalable with average wind speed and wind deviation or turbulence strength.
3. Pilot/mixer cutoff frequency changes as a function of task accuracy and turbulence level. Therefore, to establish a design requirement for gust response the control bandwidth must be set as a function of both of these parameters.
4. The same level of mixer cutoff frequency was required regardless of aircraft orientation.
5. The current ADS-33D disturbance rejection requirements appear too lenient in the pitch and roll axes when applied to the UH-60 utility class of helicopter.

6. Accuracy of on-axis dynamics appears to be of more importance than coupling effects when modeling the inverse of the gust mixer input.
7. The PSD of the gust mixer input model follows the typical PSD of atmospheric turbulence, with a parallel shift with varying levels of turbulence.
8. The PSD of the gust mixer input for both pitch and roll are nearly identical.

B. RECOMMENDATIONS FOR FOLLOW-ON RESEARCH

It is recommended that further testing be conducted with a more accurate real-time record of flow field characteristics at the hover location and a larger spread of gust conditions. Further validation should be conducted through employment of the model in a piloted simulation.

Possible applications of this research include not only handling qualities evaluation, but also training and safety applications. Accurate turbulence models for training simulators for all helicopter operators, including shipboard operators, military, law enforcement, firefighters, and tour operators, can improve the safety and readiness of pilots familiar with handling the nature of turbulent flow fields.

LIST OF REFERENCES

1. Baillie, S.W. and Morgan, J.M., "An In-flight Investigation into the Relationships Among Control Sensitivity, Control Bandwidth and Disturbance Rejection Bandwidth Using a Variable Stability Helicopter," 15th European Rotorcraft Forum, Amsterdam, The Netherlands, September 1989.
2. Anon., "Handling Qualities Requirements for Military Rotorcraft," Aeronautical Design Standard - 33 (ADS-33D-PRF), US Army Aviation and Missile Command, May 1996.
3. Dahl, H.J. and Faulkner, A.J., "Helicopter Simulation in Atmospheric Turbulence," 4th European Rotorcraft Forum, Stresa, Italy, September 1978.
4. Costello, M., Gaonkar, G.H., Prasad, J.V.R., Schrage, D.P., "Some Issues on Modeling Atmospheric Turbulence Experienced by Helicopter Rotor Blades," *Journal of the American Helicopter Society*, April 1992, pp. 71-75.
5. Hess, R.A., "Rotorcraft Handling Qualities in Turbulence," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 1, January-February 1995, pp. 39-45.
6. Woo, H., Peterka, J., Cermak, J., "Wind Tunnel Measurements in the Wakes of Structures", NASA CR-2806, March 1977.
7. Maull, D.J., "The Wake Characteristics of a Bluff Body in a Shear Flow," *Aerodynamics of Atmospheric Shear Flows*, AGARD CP-48, brief 16, 1970.
8. Technical Manual, TM 55-1520-237-10, with change 3, Operator's Manual for Army UH-60A Helicopter, Headquarters, Department of the Army. 30 October 1998.
9. Tischler, M.B., Cauffman, M.G., "Frequency-Response Method for Rotorcraft System Identification with Applications to the BO-105 Helicopter", 46th Annual Meeting of the American Helicopter Society, May 1990, Washington, D.C.
10. Tischler, M.B., Cauffman, M.G., "CIFER Version 2.1, Comprehensive Identification from FrEQuency Responses, An interactive facility for system identification and verification," Sterling Software, Palo Alto, CA.
11. Fletcher, J.W., "Identification of UH-60 Stability Derivative Models in hover from Flight Test Data," American Helicopter Society 49th Annual Forum, St. Louis, Missouri, May, 1993.
12. Atencio, A. Jr., "Fidelity Assessment of a UH-60A Simulation on the NASA Ames Vertical Motion Simulator," NASA TM 104016, September 1993.

13. Tischler, M.B., Colbourne, J.D., Morel, M.R., Biezd, D.J., Levine, W.S., and Moldoveanu, V., "Conduit-A New Multidisciplinary Integration Environment For Flight Control Development," AIAA Guidance, Navigation, and Control Conference, New Orleans, Louisiana, August 11-13, 1997.
14. George, V.V., Gaonkar, G.H., Prasad, J.V.R., and Schrage, D.P., "Adequacy of Modeling Turbulence and Related Effects of Helicopter Response," *AIAA Journal*, Vol. 30, No. 6, 1992, pp. 1468-1469.
15. McLean, D., *Automatic Flight Control Systems*, Prentice Hall International (UK) Ltd, Hertfordshire, 1990, pp. 127-135.

APPENDIX A. SIMULINK® MODELS

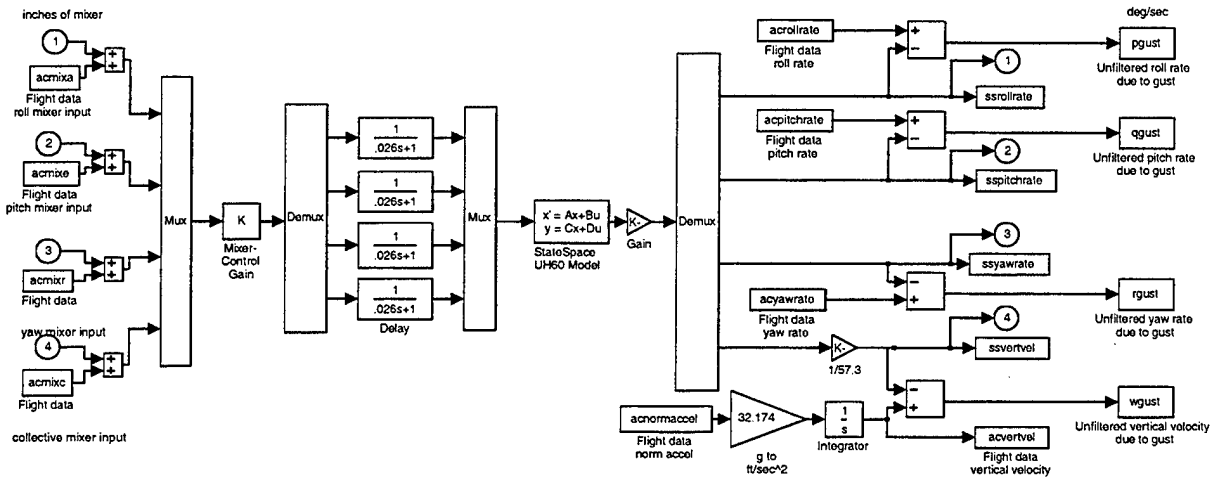
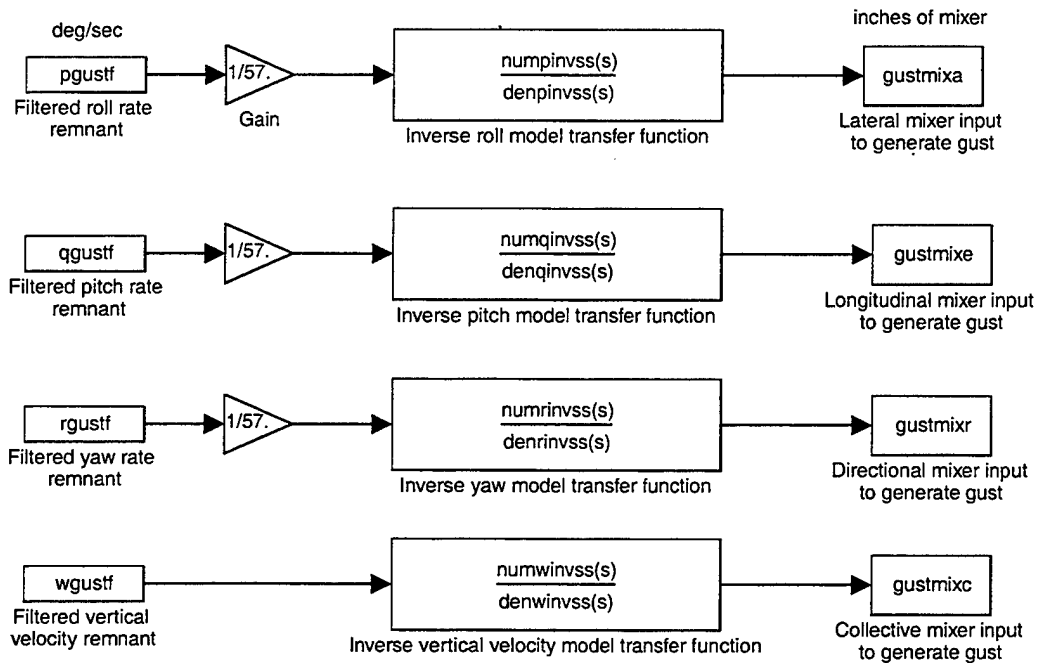
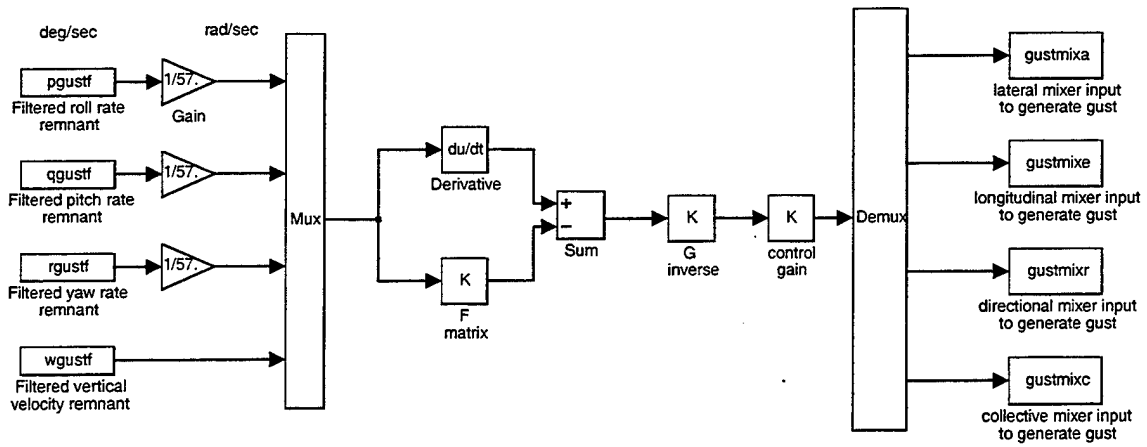


Figure A-1. Model of differencing process (extraction of gust remnant) with state-space model.



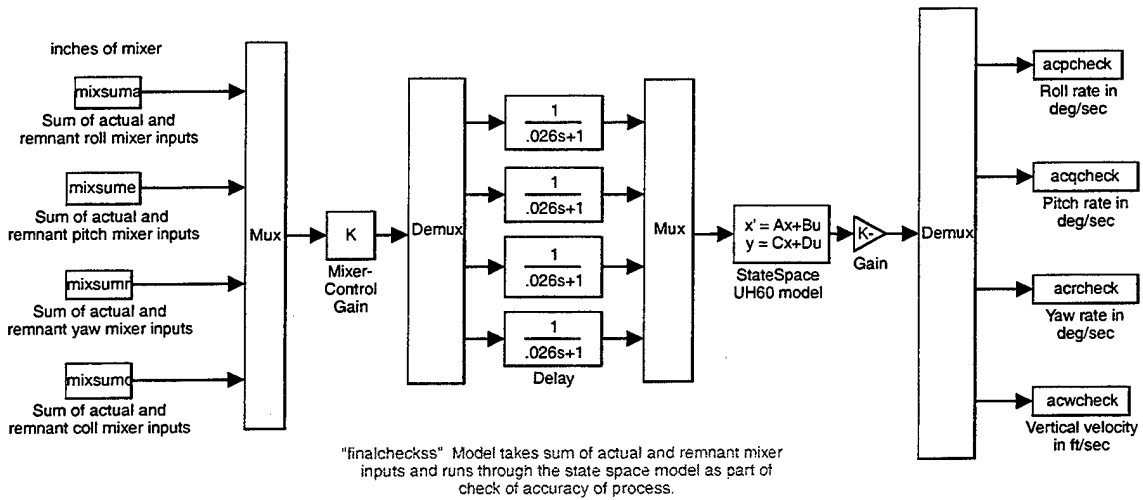
Model running filtered aircraft rate and acceleration remnants through onaxis inverse transfer function models to produce mixer inputs required to produce the gust response. Transfer function models developed from modified state space model. Remnant in deg/sec is converted to rad/sec via gain block and mixer output is in inches.

Figure A-2. On-axis inverse model for obtaining mixer input to simulate gust.



'coupinv' Model running filtered aircraft rate and vertical velocity remnant through coupled inverse model to produce mixer inputs required to produce the gust response. Remnant in deg/sec is converted to rad/sec via gain block and mixer output is in inches.

Figure A-3. Coupled inverse model for obtaining mixer input to simulate gust.



"finalcheckss" Model takes sum of actual and remnant mixer inputs and runs through the state space model as part of check of accuracy of process.

Figure A-4. Check of inverse process by summing

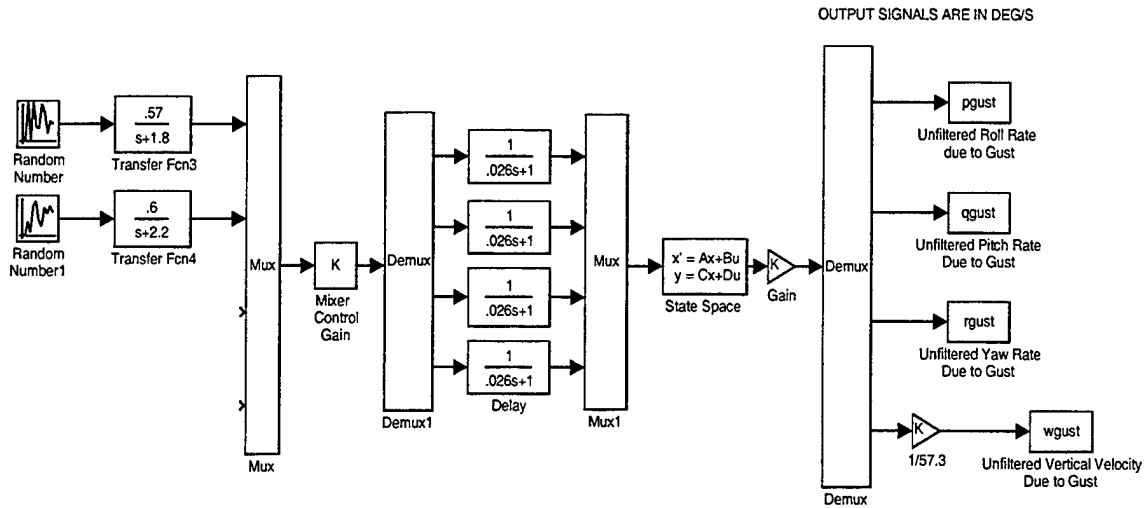


Figure A-5. Check of gust excitation model using random number generator.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MATLAB® SCRIPTS

The following MATLAB® scripts reside in the NASA Ames afrserver under the directory /u9/sjlabows/turbinv. The codes are presented below and the flight test data they utilize resides in mat files under the same directory with the following naming convention:

NNNhov.mat – where NNN is the flight number.

The mat files have been formatted from flight test data in the TRENDS database using the OUTDATA feature of TRENDS.

Script Name	Function
SSSS.M	Determine remnant aircraft rates due to atmospheric disturbance and invert the remnant to generate control inputs that will simulate disturbance response. The coupled state space model is used in the forward path and the on-axis transfer functions extracted from the overall model are used in the inverse process.
FORMDATA.M	Load Trends data to workspace and format for use in SIMULINK®.
PLOTDIFFSS.M	Plot time histories of aircraft rates, model rates from control time histories, and remnant rates from differencing process.
PZMOD.M	Modify the poles and zeros of the on-axis transfer functions extracted from the state space model to only include 0.2-50 rad/sec response.
INVCHECKSS.M	Filter and compare the actual mixer input and the remnant required mixer input. As a final check of the model the two mixer inputs are summed and run through the transfer function model and the aircraft rate output compared to the actual aircraft output.
SSSS_COUP.M	Determine remnant aircraft rates due to atmospheric disturbance and invert the remnant to generate control inputs that will simulate disturbance response. The coupled state space model is used in the forward path and the quasi-

steady simplified inverse model is used in the inverse process.

INVCHECKSS_COUP.M	Filter and compare the actual mixer input and the remnant required mixer input. As a final check of the quasi-steady simplified inverse model the two mixer inputs are summed and run through the transfer function model and the aircraft rate output compared to the actual aircraft output.
SSTF.M	Determine remnant aircraft rates due to atmospheric disturbance and invert the remnant to generate control inputs that will simulate disturbance response. The coupled state space model is used in the forward path and the on-axis transfer functions from NAVFIT solutions of flight test data are used in the inverse process.
INVCHECKTF.M	Filter and compare the actual mixer input and the remnant required mixer input. As a final check of the on-axis transfer functions from NAVFIT the two mixer inputs are summed and run through the transfer function model and the aircraft rate output compared to the actual aircraft output.
BODECHECK.M	Bode plot comparison of frequency response of various models in order to reconcile any differences in the spectrum of the models.
PITCHCHECKER.M	Check the accuracy of the on-axis pitch transfer function models with flight test data from a pitch doublet.
ROLLCHECKER.M	Check the accuracy of the on-axis roll transfer function models with flight test data from a roll doublet.


```

%% Bandpass filter (1.0-10 rad/sec) remnant rate due to gust

[b,a]=ellip(3,0.2,40,[.15915*1.3 1.5915*.87]*2/samp);
pgustf=[t filter(b,a,pgust)];
qgustf=[t filter(b,a,qgust)];
rgustf=[t filter(b,a,rgust)];
wgustf=[t filter(b,a,wgust)];

%% Run script to plot the a/c rates and differences (remnants)

plotdiffss

% Numerator and denominators for on-axis transfer functions identified
from
% state space identified model.

[A2,B2,C2,D2]=linmod('ssdiff'); % Using block diagram with output of
rad/s

[nump,denp]=ss2tf(A2,B2,C2,D2,1);
nump=nump(1,:);
[numq,denq]=ss2tf(A2,B2,C2,D2,2);
numq=numq(2,:);
[numr,denr]=ss2tf(A2,B2,C2,D2,3);
numr=numr(3,:);
[numw,denw]=ss2tf(A2,B2,C2,D2,4);
numw=numw(4,:);

% Modify state space extracted transfer functions to enable inverse
pzmod

% Invert on-axis transfer functions

numpinvss=conv(denpnew,[10000*10000]);
denpinvss=conv(numpnew,conv([1 140 10000],[1 140 10000]));

numqinvss=conv(denqnew,[10000*10000]);
denqinvss=conv(numqnew,conv([1 140 10000],[1 140 10000]));

numrinvss=conv(denrnew,[10000*10000]);
denrinvss=conv(numrnew,conv([1 140 10000],[1 140 10000]));

numwinvss=conv(denwnew,[10000*10000]);
denwinvss=conv(numwnew,conv([1 140 10000],[1 140 10000]));

% Run simulation inverting filtered rates to give control inputs

options=simset('FixedStep',1/samp);
[t,x,y]=sim('ssinv',time,options);

% At this point we have outputs from the ssinv simulation of mixer
% input that will cause the gust response. Now format these for input
% to a simulation model, via INVCHECKSS.m.

```

2. SCRIPT FORMDATA.M

```
% formdata.m
% LCDR S.J. Labows, 24 Aug 99
% Load Trends data to workspace and format for use in simulink.

eval(['load ',filename])

acrollrate=[TIME DR01S-mean(DR01S)]; % Subtract trim value to null
bias
acpitchrate=[TIME DR00S-mean(DR00S)];
acyawrate=[TIME DR02S-mean(DR02S)];
acnormaccel=[TIME DL02S-mean(DL02S)];

acmixa=[TIME DMIXA-mean(DMIXA)]; % Subtract trim value to null bias
acmixe=[TIME DMIXE-mean(DMIXE)];
acmixr=[TIME DMIXR-mean(DMIXR)];
acmixc=[TIME DMIXC-mean(DMIXC)];
```

3. SCRIPT PLOTDIFFS.M

```
% plotdiffss.m
% LCDR S.J. Labows, 7 Oct 99
% Plot time histories of a/c output, ss model output and difference

n=input('What is desired length of plots in seconds? (less than or
equal to record length)')

% Pitch
figure(1)
plot(TIME(1:n*samp),acpitchrate(1:n*samp,2),'-
',TIME(1:n*samp),sspitchrate(1:n*samp),'-.',...
TIME(1:n*samp),qgust(1:n*samp),':',TIME(1:n*samp),qgustf(1:n*samp,2),'-
-')
legend('Actual A/C Pitch Rate','SS Model Pitch Rate','Unfiltered Pitch
Rate Remnant','Filtered Pitch Rate Remnant')
xlabel('Time (sec)');ylabel('A/C Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])

% Roll
figure(2)
plot(TIME(1:n*samp),acrollrate(1:n*samp,2),'-
',TIME(1:n*samp),ssrollrate(1:n*samp),'-.',...
TIME(1:n*samp),pgust(1:n*samp),':',TIME(1:n*samp),pgustf(1:n*samp,2),'-
-')
legend('Actual A/C Roll Rate','SS Model Roll Rate','Unfiltered Roll
Rate Remnant','Filtered Roll Rate Remnant')
xlabel('Time (sec)');ylabel('A/C Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])

% Yaw - current effort focuses on pitch and roll results
%figure(3)
%plot(TIME(1:n*samp),acyawrate(1:n*samp,2),'-
',TIME(1:n*samp),ssyawrate(1:n*samp),'-.',...
%TIME(1:n*samp),rgust(1:n*samp),':',TIME(1:n*samp),rgustf(1:n*samp,2),'
--')
%legend('Actual A/C Yaw Rate','SS Model Yaw Rate','Unfiltered Yaw Rate
Remnant','Filtered Yaw Rate Remnant')
%xlabel('Time (sec)');ylabel('A/C Rate (deg/sec)');grid on
%title(['Data from flight ',flight,', event ',event])
```


4. SCRIPT PZMOD.M

```
% pzmod.m
% LCDR Labows
% 30 Nov 99 (modified 4 Jan 00)
% This script is designed to modify the poles and zeros of the on-axis
tf
% extracted from the state space model to only include 0.2-50 rad/sec
response
kpp=1;kpq=1;kpr=1;kpw=1;
kzp=1;kzq=1;kzr=1;kzw=1;
pp1=[];pq1=[];pr1=[];pw1=[];
zp1=[];zq1=[];zr1=[];zw1=[];

%%%%% Roll
sysp=tf(numpp,denpp); % Set extracted transfer function model as LTI
system
[zp,pp,kpold,ts]=zpkdata(sysp,'v'); % Pull out zero, pole, gain
information

% Eliminate poles and zeros less than 0.2 and greater than 50 rad/sec
for n=1:length(pp)
    if abs(pp(n))>50
        kpp=kpp*(abs(pp(n)));
    elseif abs(pp(n))<0.2
        pp1=[pp1;0];
    else
        pp1=[pp1;pp(n)];
    end
end

for n=1:length(zp)
    if abs(zp(n))>50
        kzp=kzp*(abs(zp(n)));
    elseif abs(zp(n))<0.2
        zp1=[zp1;0];
    else
        zp1=[zp1;zp(n)];
    end
end

% Set new system parameters and required parameters for bode plot
kpnew=kpold*kzp/kpp;
syspnew=zpk(zp1,pp1,kpnew);
[numppnew,denppnew]=tfdata(syspnew,'v');
[magppnew,phaseppnew,wpnew]=bode(numppnew,denppnew);
magppnewdb=20*log10(magppnew);

%%%% Pitch
sysq=tf(numq,denq);
[zq,pq,kqold,ts]=zpkdata(sysq,'v');
```

```

for n=1:length(pq)
    if abs(pq(n))>50
        kpq=kpq*(abs(pq(n)));
    elseif abs(pq(n))<0.2
        pq1=[pq1;0];
    else
        pq1=[pq1;pq(n)];
    end
end

for n=1:length(zq)
    if abs(zq(n))>50
        kzq=kzq*(abs(zq(n)));
    elseif abs(zq(n))<0.2
        zq1=[zq1;0];
    else
        zq1=[zq1;zq(n)];
    end
end

kqnew=kqold*kzq/kpq;
sysqnew=zpk(zq1,pq1,kqnew);
[numqnew,denqnew]=tfdata(sysqnew,'v');
[magqnew,phaseqnew,wqnew]=bode(numqnew,denqnew);
magqnewdb=20*log10(magqnew);

%%% Yaw

sysr=tf(numr,denr);
[zr,pr,krold,ts]=zpkdata(sysr,'v');

for n=1:length(pr)
    if abs(pr(n))>50
        kpr=kpr*(abs(pr(n)));
    elseif abs(pr(n))<0.2
        pr1=[pr1;0];
    else
        pr1=[pr1;pr(n)];
    end
end

for n=1:length(zr)
    if abs(zr(n))>50
        kzr=kzr*(abs(zr(n)));
    elseif abs(zr(n))<0.2
        zr1=[zr1;0];
    else
        zr1=[zr1;zr(n)];
    end
end

krnew=krold*kzr/kpr;
sysrnew=zpk(zr1,pr1,krnew);

```

```

[numrnew,denrnew]=tfdata(sysrnew,'v');
%bode(numrnew,denrnew);
[magrnew,phasernew,wrnew]=bode(numrnew,denrnew);
magrnewdb=20*log10(magrnew);

%%% Vertical Velocity

sysw=tf(numw,denw);
[zw,pw,kwold,ts]=zpkdata(sysw,'v');

for n=1:length(pw)
    if abs(pw(n))>20
        kpw=kpw*(abs(pw(n)));
    elseif abs(pw(n))<0.2
        pw1=[pw1;0];
    else
        pw1=[pw1;pw(n)];
    end
end

for n=1:length(zw)
    if abs(zw(n))>20
        kzw=kzw*(abs(zw(n)));
    elseif abs(zw(n))<0.2
        zw1=[zw1;0];
    else
        zw1=[zw1;zw(n)];
    end
end

kwnew=kwold*kzw/kpw;
syswnew=zpk(zw1,pw1,kwnew);
[numwnew,denwnew]=tfdata(syswnew,'v');
[magwnew,phaseswnew,wwnew]=bode(numwnew,denwnew);
magwnewdb=20*log10(magwnew);

```

5. SCRIPT INVCHECKSS.M

```
% invcheckss.m
% UH-60 Disturbance Rejection Study
% LCDR S.J. Labows
% 30 Nov 99 (modified 4 Jan 00)

% This program takes the actual mixer input and the remnant required
% mixer input
% and filters and compares the two. As a final check of the model the
% two mixer
% inputs are summed and run through the transfer function model and
% the aircraft
% rate output compared to the actual aircraft output.

% Filter and sum mixer inputs from aircraft and gust remnant

[b,a]=ellip(3,0.2,40,[.15915*1.3 1.5915*.87]*2/samp);

acmixeraf=filter(b,a,acmixa(:,2)); % roll mixer
gustmixaf=filter(b,a,gustmixa); % roll mixer remnant
mixsumaf=acmixeraf+gustmixaf; % sum of both mixer inputs
mixsuma=[t mixsumaf]; % format sum for simulink input

acmixeref=filter(b,a,acmixe(:,2)); % pitch
gustmixef=filter(b,a,gustmixe);
mixsumef=acmixeref+gustmixef;
mixsume=[t mixsumef];

acmixerrf=filter(b,a,acmixr(:,2)); % yaw
gustmixrf=filter(b,a,gustmixr);
mixsumrf=acmixerrf+gustmixrf;
mixsumr=[t mixsumrf];

acmixercf=filter(b,a,acmixc(:,2)); % heave
gustmixcf=filter(b,a,gustmixc);
mixsumcf=acmixercf+gustmixcf;
mixsumc=[t mixsumcf];

% Set x-axis for plots

m=input('What is desired length of plots in seconds? (less than or
equal to record length)')

% Plot aircraft mixer, remnant mixer and sum mixer inputs

figure(6)
plot(TIME(1:m*samp),acmixeraf(1:m*samp),'--
',TIME(1:m*samp),gustmixaf(1:m*samp),'.',TIME(1:m*samp),mixsumaf(1:m*samp))
legend('Filtered A/C Lateral Mixer Input','Filtered Remnant Required
Lateral Mixer Input','Sum of Mixer Inputs')
```

```

xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event])

figure(7)
plot(TIME(1:m*samp),acmixeref(1:m*samp),'--
',TIME(1:m*samp),gustmixef(1:m*samp),'.',TIME(1:m*samp),mixsumef(1:m*sa
mp))
legend('Filtered A/C Longitudinal Mixer Input','Filtered Remnant
Required Longitudinal Mixer Input','Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event])

%figure(8)
%plot(TIME(1:m*samp),acmixerrf(1:m*samp),'--
',TIME(1:m*samp),gustmixrf(1:m*samp),'.',TIME(1:m*samp),mixsumrf(1:m*sa
mp))
legend('Filtered A/C Directional Mixer Input','Filtered Remnant
Required Directional Mixer Input','Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event])

%figure(9)
%plot(TIME(1:m*samp),acmixercf(1:m*samp),'--
',TIME(1:m*samp),gustmixcf(1:m*samp),'.',TIME(1:m*samp),mixsumcf(1:m*sa
mp))
%legend('Filtered A/C Collective Mixer Input','Filtered Remnant
Required Collective Mixer Input','Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
%title(['Data from flight ',flight,', event ',event])

% Run simulation to check A/C rate resulting from sum of actual mixer
and remnant required mixer
% against the actual A/C rate. Litmus check of the entire process.

options=simset('FixedStep',1/samp);
[t,x,y]=sim('finalcheckss',time,options);

% Plot actual aircraft rates and rates generated by sum of mixer
inputs. Also generate
% plot of residual aircraft rates from difference of

figure(10)
plot(TIME(1:m*samp),filter(b,a,acrollrate(1:m*samp,2)),TIME(1:m*samp),a
cpcheck(1:m*samp),'--')
legend('Filtered A/C Roll Rate','Roll Rate from Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])
axis([0 60 -15 15]);

figure(11)
plot(TIME(1:m*samp),filter(b,a,acrollrate(1:m*samp,2))-
acpcheck(1:m*samp))
resrollratess=filter(b,a,acrollrate(1:m*samp,2))-acpcheck(1:m*samp);
legend('Residual Roll Rate')

```

```

xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])
axis([0 60 -6 6]);

figure(12)
plot(TIME(1:m*samp),filter(b,a,acpitchrate(1:m*samp,2)),TIME(1:m*samp),
acqcheck(1:m*samp),'--')
legend('Filtered A/C Pitch Rate','Pitch Rate from Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])
axis([0 60 -15 15]);

figure(13)
plot(TIME(1:m*samp),filter(b,a,acpitchrate(1:m*samp,2))-
acqcheck(1:m*samp))
respitchrateress=filter(b,a,acpitchrate(1:m*samp,2))-acqcheck(1:m*samp);
legend('Residual Pitch Rate')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])
axis([0 60 -6 6]);

%figure(14)
%plot(TIME(1:m*samp),filter(b,a,acyawrate(1:m*samp,2)),TIME(1:m*samp),a
crcheck(1:m*samp),'--')
%legend('Filtered A/C Yaw Rate','Yaw Rate from Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
%title(['Data from flight ',flight,', event ',event])

%figure(16)
%plot(TIME(1:m*samp),filter(b,a,acnormaccel(1:m*samp,2)),TIME(1:m*samp)
,acnzcheck(1:m*samp),'--')
%legend('Filtered A/C Normal Acceleration','Normal Acceleration from
Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Aircraft Accel (deg/sec^2)');grid on
%title(['Data from flight ',flight,', event ',event])

% Format .mat file to be read into CIFER in order to determine cutoff
freq of mixer inputs and
% evaluate PSD

MIXACHK=acmixa(:,2);
MIXACHKF=acmixeraf;
GMXACHK=gustmixa;
GMXACHKF=gustmixaf;
ACROLL=acrollrate(:,2);
ACROLLF=filter(b,a,acrollrate(:,2));
GROLL=acpcheck;
PGUST=pgustf(:,2);

MIXECHK=acmixe(:,2);
MIXECHKF=acmixeref;
GMXECHK=gustmixe;
GMXECHKF=gustmixef;
ACPTCH=acpitchrate(:,2);

```

```

ACPTCHCF=filter(b,a,acpitchrate(:,2));
GPTCH=acqcheck;
QGUST=qgustf(:,2);

MIXRCHK=acmixr(:,2);
MIXRCHKF=acmixerrf;
GMXRCHK=gustmixr;
GMXRCHKF=gustmixrf;
ACYAW=acyawrate(:,2);
ACYAWF=filter(b,a,acyawrate(:,2));
GYAW=acrcheck;
RGUST=rgustf(:,2);

MIXCCHK=acmixc(:,2);
MIXCCHKF=acmixercf;
GMXCCHK=gustmixc;
GMXCCHKF=gustmixcf;
ACVERT=acvertvel;
ACVERTF=filter(b,a,acvertvel);
GVERT=acwcheck;
WGUST=wgustf(:,2);

save 307out.mat TIME MIXACHK MIXACHKF GMXACHK GMXACHKF ACROLL ACROLLF
GROLL PGUST ...
MIXECHK MIXECHKF GMXECHK GMXECHKF ACPTCH ACPTCHCF GPTCH QGUST ...
MIXRCHK MIXRCHKF GMXRCHK GMXRCHKF ACYAW ACYAWF GYAW RGUST ...
MIXCCHK MIXCCHKF GMXCCHK GMXCCHKF ACVERT ACVERTF GVERT WGUST -v4

```

6. SCRIPT SSSS_COUP.M

```
% ssss_coup.m
% UH-60 Disturbance Rejection Study
% LCDR S.J. Labows
% 18 Dec 99 (modified 4 Jan 00)

% This program takes flight data from the TRENDS database and runs
% these flight test control inputs through the UH-60 State Space
% model. The outputs are then summed to produce a remnant
% time history of aircraft rates which are caused by turbulence.

clear

flight=input('What is desired flight number?','s')
event=input('What is desired event number?','s')
filename=input('What is desired input file name?','s')
time=input('What is the duration of the input file in seconds?')
samp=input('What is the sample rate of the input file?')

%% Run script to load data to workspace and null bias.

formdata

% Mechanical linkage matrix of cockpit control -> mixer input

L=[0.2425  0  0  0;
    0  0.2155  0  0;
    0  0  0.3783  0;
    0  0  0  0.1960];

% Generate F,G,C,D matrices for state space model

load fhov.mat % Loads F matrix
F(1:2,:)=0; % Eliminates u/v translational degrees of freedom
F(:,1:2)=0;
load ghov.mat % Loads G matrix
G(1:2,:)=0; % Eliminates u/v translational degrees of freedom
% C matrix set up to extract p,q,r,w from flight data
C=[0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0;
   0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0;
   0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0;
   0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0];
D=zeros(4);

%% Run simulation to perform differencing of a/c rates

options=simset('FixedStep',1/samp);
```



```

[t,x,y]=sim('ssdiff',time,options);

%% Bandpass filter (1.0-10 rad/sec) remnant rate due to gust

[b,a]=ellip(3,0.2,40,[.15915*1.3 1.5915*.87]*2/samp);
pgustf=[t filter(b,a,pgust)];
qgustf=[t filter(b,a,qgust)];
rgustf=[t filter(b,a,rgust)];
wgustf=[t filter(b,a,wgust)];

%% Run script to plot the a/c rates and differences (remnants)

plotdiffss

% Numerator and denominators for on-axis identified transfer function
models
% identified via CIPHER from flight test freq sweeps.

numptf=[47.5722];
denptf=[1 9.0304 40.1855];

numqtf=[12.113];
denqtf=[1 9.9125 3.988];

numrtf=[63.831];
denrtf=[1 49.962 16.264];

numwtf=[129.294]; % These numbers are not accurate for vert velocity
denwtf=[1 .150E-05 166.725];

% Numerator and denominators for on-axis transfer functions identified
from
% state space identified model.

[A2,B2,C2,D2]=linmod('ssdiff'); % Using block diagram with output of
rad/s

[nump,denp]=ss2tf(A2,B2,C2,D2,1);
nump=nump(1,:);
[numq,denq]=ss2tf(A2,B2,C2,D2,2);
numq=numq(2,:);
[numr,denr]=ss2tf(A2,B2,C2,D2,3);
numr=numr(3,:);
[numw,denw]=ss2tf(A2,B2,C2,D2,4);
numw=numw(4,:);

%%% Run simulation inverting filtered rates

% Modify state space model extracted transfer functions to enable
inverse
pzmod

numpinvss=conv(denpnew,[10000*10000]);
denpinvss=conv(numpinvss,conv([1 140 10000],[1 140 10000]));

```

```

numqinvss=conv(denqnew,[10000*10000]);
denqinvss=conv(numqnew,conv([1 140 10000],[1 140 10000]));

numrinvs=conv(denrnew,[10000*10000]);
denrinvs=conv(numrnew,conv([1 140 10000],[1 140 10000]));

numwinv=conv(denw,[2500]);
denwinv=conv(numw,[1 70 2500]);

% Load coupled inverse F and G matrices. Then run inverse process to
% obtain mixer inputs for gust response. Time delay has not been
% applied.

load fcoup.mat
load gcoup.mat

options=simset('FixedStep',1/samp);
[t,x,y]=sim('coupinv',time,options);

% At this point we have outputs from the coupinv simulation of mixer
% input
% that will cause the gust response. Now format these for input to
% a simulation model.

```

7. SCRIPT INVCHECKSS_COUP.M

```
% invcheckss_coup.m
% UH-60 Disturbance Rejection Study
% LCDR S.J. Labows
% 18 Dec 99 (modified 4 Jan 00)

% This program takes the actual mixer input and the remnant required
% mixer input
% and filters and compares the two. As a final check of the model the
% two mixer
% inputs are summed and run through the transfer function model and
% the aircraft
% rate output compared to the actual aircraft output.

% Filter and sum mixer inputs from aircraft and gust remnant

[b,a]=ellip(3,0.2,40,[.15915*1.3 1.5915*.87]*2/samp); % bandpass
filter (1.0-10 rad/sec)

acmixeraf=filter(b,a,acmixa(:,2)); % roll mixer
gustmixaf=filter(b,a,gustmixa); % roll mixer remnant
mixsumaf=acmixeraf+gustmixaf; % sum of both mixer inputs
mixsuma=[t-.092 mixsumaf]; % format sum for simulink input

acmixeref=filter(b,a,acmixe(:,2)); % pitch
gustmixef=filter(b,a,gustmixe);
mixsumef=acmixeref+gustmixef;
mixsume=[t-.092 mixsumef];

acmixerrf=filter(b,a,acmixr(:,2)); % yaw
gustmixrf=filter(b,a,gustmixr);
mixsumrf=acmixerrf+gustmixrf;
mixsumr=[t-.026 mixsumrf];

acmixercf=filter(b,a,acmixc(:,2)); % heave
gustmixcf=filter(b,a,gustmixc);
mixsumcf=acmixercf+gustmixcf;
mixsumc=[t-.026 mixsumcf];

% Set x-axis for plots

m=input('What is desired length of plots in seconds? (less than or
equal to record length)')

% Plot aircraft mixer, remnant mixer and sum mixer inputs

figure(6)
plot(TIME(1:m*samp),acmixeraf(1:m*samp),'--
',TIME(1:m*samp),gustmixaf(1:m*samp),'.',TIME(1:m*samp),mixsumaf(1:m*sa
mp))
```

```

legend('Filtered A/C Lateral Mixer Input','Filtered Remnant Required
Lateral Mixer Input','Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event,', coupled
inverse'])

figure(7)
plot(TIME(1:m*samp),acmixeref(1:m*samp),'--
',TIME(1:m*samp),gustmixef(1:m*samp),'.',TIME(1:m*samp),mixsumef(1:m*sa
mp))
legend('Filtered A/C Longitudinal Mixer Input','Filtered Remnant
Required Longitudinal Mixer Input','Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event,', coupled
inverse'])

%figure(8)
%plot(TIME(1:m*samp),acmixerrf(1:m*samp),'--
',TIME(1:m*samp),gustmixrf(1:m*samp),'.',TIME(1:m*samp),mixsumrf(1:m*sa
mp))
%legend('Filtered A/C Directional Mixer Input','Filtered Remnant
Required Directional Mixer Input','Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
%title(['Data from flight ',flight,', event ',event,', coupled
inverse'])

%figure(9)
%plot(TIME(1:m*samp),acmixercf(1:m*samp),'--
',TIME(1:m*samp),gustmixcf(1:m*samp),'.',TIME(1:m*samp),mixsumcf(1:m*sa
mp))
%legend('Filtered A/C Collective Mixer Input','Filtered Remnant
Required Collective Mixer Input','Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
%title(['Data from flight ',flight,', event ',event,', coupled
inverse'])

% Run simulation to check A/C rate resulting from sum of actual mixer
and remnant required mixer
% against the actual A/C rate. Litmus check of the entire process.

options=simset('FixedStep',1/samp);
[t,x,y]=sim('finalcheckss',time,options);

% Plot actual aircraft rates and rates generated by sum of mixer
inputs. Also generate
% plot of residual aircraft rates from difference of

figure(10)
plot(TIME(1:m*samp),filter(b,a,acrollrate(1:m*samp,2)),TIME(1:m*samp),a
cpcheck(1:m*samp),'--')
legend('Filtered A/C Roll Rate','Roll Rate from Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event,', coupled
inverse'])

```

```

axis([0 60 -15 15]);

figure(11)
plot(TIME(1:m*samp),filter(b,a,acrollrate(1:m*samp,2))-
acpcheck(1:m*samp))
resrollratess=filter(b,a,acrollrate(1:m*samp,2))-acpcheck(1:m*samp);
legend('Residual Roll Rate')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event,', coupled
inverse'])
axis([0 60 -6 6]);

figure(12)
plot(TIME(1:m*samp),filter(b,a,acpitchrate(1:m*samp,2)),TIME(1:m*samp),
acqcheck(1:m*samp),'--')
legend('Filtered A/C Pitch Rate','Pitch Rate from Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event,', coupled
inverse'])
axis([0 60 -15 15]);

figure(13)
plot(TIME(1:m*samp),filter(b,a,acpitchrate(1:m*samp,2))-
acqcheck(1:m*samp))
respitchratess=filter(b,a,acpitchrate(1:m*samp,2))-acqcheck(1:m*samp);
legend('Residual Pitch Rate')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event,', coupled
inverse'])
axis([0 60 -6 6]);

%figure(14)
%plot(TIME(1:m*samp),filter(b,a,acyawrate(1:m*samp,2)),TIME(1:m*samp),a
crcheck(1:m*samp),'--')
%legend('Filtered A/C Yaw Rate','Yaw Rate from Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
%title(['Data from flight ',flight,', event ',event,', coupled
inverse'])

%figure(16)
%plot(TIME(1:m*samp),filter(b,a,acnormaccel(1:m*samp,2)),TIME(1:m*samp)
,acnzcheck(1:m*samp),'--')
%legend('Filtered A/C Normal Acceleration','Normal Acceleration from
Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Aircraft Accel (deg/sec^2)');grid on
%title(['Data from flight ',flight,', event ',event,', coupled
inverse'])

% Format .mat file to be read into CIFER in order to determine cutoff
freq of mixer inputs and
% evaluate PSD

MIXACHK=acmixa(:,2);
MIXACHKF=acmixerf;

```

```

GMXACHK=gustmixa;
GMXACHKF=gustmixaf;
ACROLL=acrollrate(:,2);
ACROLLF=filter(b,a,acrollrate(:,2));
GROLL=acpcheck;
PGUST=pgustf(:,2);

MIXECHK=acmixe(:,2);
MIXECHKF=acmixeref;
GMXECHK=gustmixe;
GMXECHKF=gustmixef;
ACPTCH=acpitchrate(:,2);
ACPTCHCF=filter(b,a,acpitchrate(:,2));
GPTCH=acqcheck;
QGUST=qgustf(:,2);

MIXRCHK=acmixr(:,2);
MIXRCHKF=acmixerrf;
GMXRCHK=gustmixr;
GMXRCHKF=gustmixrf;
ACYAW=acyawrate(:,2);
ACYAWF=filter(b,a,acyawrate(:,2));
GYAW=acrcheck;
RGUST=rgustf(:,2);

MIXCCHK=acmixc(:,2);
MIXCCHKF=acmixercf;
GMXCCHK=gustmixc;
GMXCCHKF=gustmixcf;
ACVERT=acvertvel;
ACVERTF=filter(b,a,acvertvel);
GVERT=acwcheck;
WGUST=wgustf(:,2);

save 106outcoup.mat TIME MIXACHK MIXACHKF GMXACHK GMXACHKF ACROLL
ACROLLF GROLL PGUST ...
MIXECHK MIXECHKF GMXECHK GMXECHKF ACPTCH ACPTCHCF GPTCH QGUST ...
MIXRCHK MIXRCHKF GMXRCHK GMXRCHKF ACYAW ACYAWF GYAW RGUST ...
MIXCCHK MIXCCHKF GMXCCHK GMXCCHKF ACVERT ACVERTF GVERT WGUST -v4

```

8. SCRIPT SSTF.M

```
% sstf.m
% UH-60 Disturbance Rejection Study
% LCDR S.J. Labows
% 9 Nov 99

% This program takes flight data from the TRENDS database and runs
% these flight test control inputs through the UH-60 State Space
% model. The outputs are then summed to produce a remnant
% time history of aircraft rates which are caused by turbulence.
% Inverse script is called using on-axis transfer functions generated
% from frequency sweep flight test data.

% clear

flight=input('What is desired flight number?','s')
event=input('What is desired event number?','s')
filename=input('What is desired input file name?','s')
time=input('What is the duration of the input file in seconds?')
samp=input('What is the sample rate of the input file?')

%% Run script to load data to workspace.

formdata

% Mechanical linkage matrix of cockpit control -> mixer input

L=[0.2425  0  0  0;
   0  0.2155  0  0;
   0  0  0.3783  0;
   0  0  0  0.1960];

% Generate F,G,C,D matrices for state space model

load fhov.mat      % Loads F matrix
F(1:2,:)=0;        % Eliminates u/v translational degrees of freedom
F(:,1:2)=0;
load ghov.mat      % Loads G matrix
G(1:2,:)=0; % Eliminates u/v translational degrees of freedom
C=[0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0;
   0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0;
   0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0;
   0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0];
D=zeros(4);

%% Run simulation to perform differencing of a/c rates
```

```

options=simset('FixedStep',1/samp);
[t,x,y]=sim('ssdiff',time,options);

%% Bandpass filter (0.8-10 rad/sec) remnant rate due to gust

[b,a]=ellip(3,0.2,40,[.1273 1.5915]*2/samp);
pgustf=[t filter(b,a,pgust)];
qgustf=[t filter(b,a,qgust)];
rgustf=[t filter(b,a,rgust)];
azgustf=[t filter(b,a,azgust)];

%% Run script to plot the a/c rates and differences (remnants)

plotdiffss

% Numerator and denominators for on-axis identified transfer function
models
% identified via CIPHER from flight test freq sweeps.

numptf=[47.5722];
denptf=[1 9.0304 40.1855];

numqtf=[12.113];
denqtf=[1 9.9125 3.988];

numrtf=[63.831];
denrtf=[1 49.962 16.264];

numaztf=[129.294];
denaztf=[1 .150E-05 166.725];

% Run simulation inverting filtered rates

numpinvtf=conv(denptf,[2500]);
denpinvtf=conv(numptf,[1 70 2500]);

numqinvtf=conv(denqtf,[2500]);
denqinvtf=conv(numqtf,[1 70 2500]);

numrinvtf=conv(denrtf,[2500]);
denrinvtf=conv(numrtf,[1 70 2500]);

numazinvtf=conv(denaztf,[2500]);
denazinvtf=conv(numaztf,[1 70 2500]);

options=simset('FixedStep',1/samp);
[t,x,y]=sim('tfinv',time,options);

% At this point we have outputs from the tfinv simulation of mixer
input
% that will cause the gust response. Now format these for input to
% a simulation model.

```


9. SCRIPT INVCHECKTF.M

```
% invchecktf.m
% UH-60 Disturbance Rejection Study
% LCDR S.J. Labows
% 8 Sep 99

% This program takes the actual mixer input and the remnant required
% mixer
% input and filters and compares the two. As a final check of the
% model the
% two mixer inputs are summed and run through the transfer function
% model and
% the aircraft rate output compared to the actual aircraft output..

% Filter and sum mixer inputs from aircraft and gust remnant

[b,a]=ellip(3,0.2,40,[.1273 1.5915]*2/samp);

acmixeraf=filter(b,a,acmixa(:,2)); % roll mixer
gustmixaf=filter(b,a,gustmixa); % roll mixer remnant
mixsumaf=acmixeraf+gustmixaf; % sum of both mixer inputs
mixsuma=[t mixsumaf]; % format sum for simulink input

acmixeref=filter(b,a,acmixe(:,2)); % pitch
gustmixef=filter(b,a,gustmixe);
mixsumef=acmixeref+gustmixef;
mixsume=[t mixsumef];

acmixerrf=filter(b,a,acmixr(:,2)); % yaw
gustmixrf=filter(b,a,gustmixr);
mixsumrf=acmixerrf+gustmixrf;
mixsumr=[t mixsumrf];

acmixercf=filter(b,a,acmixc(:,2)); % heave
gustmixcf=filter(b,a,gustmixc);
mixsumcf=acmixercf+gustmixcf;
mixsumc=[t mixsumcf];

% Set x-axis for plots

m=input('What is desired length of plots in seconds? (less than or
equal to record length)')

% Plot aircraft mixer, remnant mixer and sum mixer inputs

figure(6)
plot(TIME(1:m*samp),acmixeraf(1:m*samp),'--
',TIME(1:m*samp),gustmixaf(1:m*samp),'.',TIME(1:m*samp),mixsumaf(1:m*sa
mp))
legend('Filtered A/C Lateral Mixer Input','Filtered Remnant Required
Lateral Mixer Input','Sum of Mixer Inputs')
```

```

xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event])

figure(7)
plot(TIME(1:m*samp),acmixeref(1:m*samp),'--
',TIME(1:m*samp),gustmixef(1:m*samp),'.',TIME(1:m*samp),mixsumef(1:m*samp))
legend('Filtered A/C Longitudinal Mixer Input','Filtered Remnant
Required Longitudinal Mixer Input','Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event])

figure(8)
plot(TIME(1:m*samp),acmixerrf(1:m*samp),'--
',TIME(1:m*samp),gustmixrf(1:m*samp),'.',TIME(1:m*samp),mixsumrf(1:m*samp))
legend('Filtered A/C Directional Mixer Input','Filtered Remnant
Required Directional Mixer Input','Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
title(['Data from flight ',flight,', event ',event])

%figure(9)
%plot(TIME(1:m*samp),acmixercf(1:m*samp),'--
',TIME(1:m*samp),gustmixcf(1:m*samp),'.',TIME(1:m*samp),mixsumcf(1:m*samp))
%legend('Filtered A/C Collective Mixer Input','Filtered Remnant
Required Collective Mixer Input','Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Mixer Input (inches)');grid on
%title(['Data from flight ',flight,', event ',event])

% Run simulation to check A/C rate resulting from sum of actual mixer
and remnant required mixer
% against the actual A/C rate. Litmus check of the entire process.

options=simset('FixedStep',1/samp);
[t,x,y]=sim('finalchecktf',time,options);

% Plot actual aircraft rates and rates generated by sum of mixer
inputs.

figure(10)
plot(TIME(1:m*samp),filter(b,a,acrollrate(1:m*samp,2)),TIME(1:m*samp),a
cpcheck(1:m*samp),'--')
legend('Filtered A/C Roll Rate','Roll Rate from Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])

figure(11)
plot(TIME(1:m*samp),filter(b,a,acpitchrate(1:m*samp,2)),TIME(1:m*samp),
acqcheck(1:m*samp),'--')
legend('Filtered A/C Pitch Rate','Pitch Rate from Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])

```

```

figure(12)
plot(TIME(1:m*samp),filter(b,a,acyawrate(1:m*samp,2)),TIME(1:m*samp),ac
rcheck(1:m*samp),'--')
legend('Filtered A/C Yaw Rate','Yaw Rate from Sum of Mixer Inputs')
xlabel('Time (sec)');ylabel('Aircraft Rate (deg/sec)');grid on
title(['Data from flight ',flight,', event ',event])

%figure(13)
%plot(TIME(1:m*samp),filter(b,a,acnormaccel(1:m*samp,2)),TIME(1:m*samp)
,acnzcheck(1:m*samp),'--')
%legend('Filtered A/C Normal Acceleration','Normal Acceleration from
Sum of Mixer Inputs')
%xlabel('Time (sec)');ylabel('Aircraft Accel (deg/sec^2)');grid on
%title(['Data from flight ',flight,', event ',event])

% Format .mat file to be read into CIFER in order to determine cutoff
freq of mixer inputs and
% evaluate PSD

MIXACHK=acmixa(:,2);
MIXACHKF=acmixeraf;
GMXACHK=gustmixa;
GMXACHKF=gustmixaf;
ACROLL=acrollrate(:,2);
ACROLLF=filter(b,a,acrollrate(:,2));
GROLL=acpcheck;

MIXECHK=acmixe(:,2);
MIXECHKF=acmixeref;
GMXECHK=gustmixe;
GMXECHKF=gustmixef;
ACPTCH=acpitchrate(:,2);
ACPTCHCF=filter(b,a,acpitchrate(:,2));
GPTCH=acqcheck;

MIXRCHK=acmixr(:,2);
MIXRCHKF=acmixerrf;
GMXRCHK=gustmixr;
GMXRCHKF=gustmixrf;
ACYAW=acyawrate(:,2);
ACYAWF=filter(b,a,acyawrate(:,2));
GYAW=acrcheck;

save 21001out.mat TIME MIXACHK MIXACHKF GMXACHK GMXACHKF ACROLL ACROLLF
GROLL ...
MIXECHK MIXECHKF GMXECHK GMXECHKF ACPTCH ACPTCHCF GPTCH ...
MIXRCHK MIXRCHKF GMXRCHK GMXRCHKF ACYAW ACYAWF GYAW -v4

```

10. SCRIPT BODECHECK.M

```
% bodecheck.m
% LCDR Steven Labows
% 14 Oct 99
% Bode plot comparison of tf for ss model, cifer model and flight
data.
% This is an effort to reconcile the differences between the different
models.
clear

% Set up state space model matrices (without zeroing u/v)
load fhov.mat      % Loads F matrix
load ghov.mat      % Loads G matrix
C=[0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0;
    0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0;
    0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0;
    0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0];
D=zeros(4);

% Run simulation to set all required variables within model
flight=input('What is desired flight number?','s')
event=input('What is desired event number?','s')
filename=input('What is desired input file name?','s')
time=input('What is the duration of the input file in seconds?')
samp=input('What is the sample rate of the input file?')

% Run script to load data to workspace and filter/format for use in
simulink.
% Loads data from flight specified above under 'filename'
diffdatas

% Mechanical linkage matrix of cockpit control -> mixer input. Use
inverse in
% mixer-control gain block of ssdifftf

L=[ 0.2425    0    0    0;
    0    0.2155    0    0;
    0    0    0.3783    0;
    0    0    0    0.1960];

options=simset('FixedStep',1/samp);
[t,x,y]=sim('ssdiff',time,options);

% Extract SS representation of model with loaded F and G matrices
[A1,B1,C1,D1]=linmod('ssdiff');
```

```

sys1=ss(A1,B1,C1,D1);           % Identify system as an LTI object

% Generate mag, phase and freq values for bode plot of this system for
roll and
% pitch responses. Form vectors of mag and phase values.

[mag1p,phase1p,w1p]=bode(sys1(1,1));
mlpdb=20*log10(mag1p);
mp(1:length(w1p))=(mlpdb(1,1,1:length(w1p)));
mag1pdb=mp';
php(1:length(w1p))=(phase1p(1,1,1:length(w1p)));
phase1p=php';

[mag1q,phase1q,w1q]=bode(sys1(2,2));
mlqdb=20*log10(mag1q);
mq(1:length(w1q))=(mlqdb(1,1,1:length(w1q)));
mag1qdb=mq';
phq(1:length(w1q))=(phase1q(1,1,1:length(w1q)));
phase1q=phq';

% Extract SS representation of model with F and G matrices modified
% to eliminate u and v translational DOF

F(1:2,:)=0;           % Eliminates F matrix u/v translational degrees of
freedom
F(:,1:2)=0;
G(1:2,:)=0; % Eliminates G matrix u/v translational degrees of freedom
[A2,B2,C2,D2]=linmod('ssdiff');
sys2=ss(A2,B2,C2,D2);

% Numerator and denominators for on-axis transfer functions identified
from
% state space identified model.
[nump,denp]=ss2tf(A2,B2,C2,D2,1);
nump=nump(1,:);
[numq,denq]=ss2tf(A2,B2,C2,D2,2);
numq=numq(2,:);
[numr,denr]=ss2tf(A2,B2,C2,D2,3);
numr=numr(3,:);
[numaz,denaz]=ss2tf(A2,B2,C2,D2,4);
numaz=numaz(4,:);

% Modify poles and zeros of SS model to enable inversion of extracted
transfer functions
pzmod

% Generate mag, phase and freq values for bode plot of this system for
roll and
% pitch responses.

[mag2p,phase2p,w2p]=bode(sys2(1,1));
m2pdb=20*log10(mag2p);
np(1:length(w2p))=(m2pdb(1,1,1:length(w2p)));
mag2pdb=np';

```

```

phap(1:length(w2p))=(phase2p(1,1,1:length(w2p)));
phase2p=phap';

[mag2q,phase2q,w2q]=bode(sys2(2,2));
m2qdb=20*log10(mag2q);
nq(1:length(w2q))=(m2qdb(1,1,1:length(w2q)));
mag2qdb=nq';
phaq(1:length(w2q))=(phase2q(1,1,1:length(w2q)));
phase2q=phaq';

% Values from on-axis TF model extracted using CIFER
nump=[47.5722];
denp=[1 9.0304 40.1855];
[mag3p,phase3p,w3p]=bode(nump,denp);
mag3pdb=20*log10(mag3p);

numq=[12.113];
denq=[1 9.9125 3.988];
[mag3q,phase3q,w3q]=bode(numq,denq);
mag3qdb=20*log10(mag3q);

% Load data from CIFER frequency response of flight test data for
specific flight.
% (loads w,mag,phase)
% In this case we are using flight 189 frequency responses
load fr189p.mat % Data for 189 P/MIXA
load fr189q.mat % Data for 189 Q/MIXE

% Plot overlay bode plots with all four models represented

figure(1); % P/MIXA bode plot for all 4 models
subplot(2,1,1)

semilogx(w1p,mag1pdb,'-',w2p,mag2pdb,'--',w3p,mag3pdb,':',w,mag,'-
.',wpnew,magpnewdb)
grid
title(['Data from flight ',flight,', event ',event,' for P/MIXA'])
xlabel('Frequency (rad/sec)');
ylabel('Gain (dB)');
axis([0.1 100 -50 50]);
legend('SS Model','SS Model with u/v zeroed','NAVFIT TF Model','Flight
Data FR','Modified PZ TF Model');

subplot(2,1,2)
semilogx(w1p,phase1p,'-',w2p,phase2p,'--',w3p,phase3p-
360,':',w,phase,'-.',wpnew,phasepnew-360)
grid
xlabel('Frequency (rad/sec)');
ylabel('Phase (deg)');
axis([0.1 100 -1000 -200]);

figure(2); % Q/MIXE bode plot for all 4 models
subplot(2,1,1)
semilogx(w1q,mag1qdb,'-',w2q,mag2qdb,'--',w3q,mag3qdb,':',w2,mag2,'-')

```

```

grid
title(['Data from flight ',flight,', event ',event,', for Q/MIXE'])
xlabel('Frequency (rad/sec)');
ylabel('Gain (dB)');
axis([0.1 100 -60 40]);
legend('SS Model', 'SS Model with u/v zeroed', 'NAVFIT TF Model', 'Flight
Data FR');

subplot(2,1,2)
semilogx(w1q,phase1q,'-',w2q,phase2q+360,'--
',w3q,phase3q,':',w2,phase2,'-.')
grid
xlabel('Frequency (rad/sec)');
ylabel('Phase (deg)');
axis([0.1 100 -600 200]);

```

11. SCRIPT PITCHCHECKER.M

```
% pitchchecker.m
% UH-60 Disturbance Rejection Study
% LCDR S.J. Labows
% 14 Sep 99

% This program takes longitudinal doublet info from flight test and
runs through
% the extracted transfer function model to verify accuracy of model

clear

load pitchdoub.mat

time=input('What is the duration of the input file in seconds?')
samp=input('What is the sample rate of the input file?')

% subtract mean of aircraft inputs and rates
acqrate=[TIME DR00S-mean(DR00S)];
acqmix=[TIME DM00*.02108-mean(DM00*.02108)];

% Set transfer function parameters from NAVFIT
numq=[12.113];
denq=[1 9.9125 3.988];

% Run simulation to verify model and aircraft rate outputs
options=simset('FixedStep',1/samp);
[t,x,y]=sim('pitchcheck',time,options);

% Filter rate outputs
[b,a]=ellip(3,0.2,40,[.1273 1.5915]*2/samp);
acqratef=filter(b,a,acqrate);
pitchratecheckf=filter(b,a,pitchratecheck);

% Plot both aircraft and model rates
plot(TIME(1:time*samp),acqratef(1:time*samp,2),TIME(1:time*samp),pitchr
atecheckf(1:time*samp),'--')
legend('Filtered A/C Pitch Rate','Filtered TF Model Pitch Rate')
xlabel('Time (sec)');ylabel('Pitch Rate (deg/sec)');grid on
title('Data from flight 189.36 - Pitch Doublet')
```


12. SCRIPT ROLLCHECKER.M

```
% rollchecker.m
% UH-60 Disturbance Rejection Study
% LCDR S.J. Labows
% 7 Sep 99

% This program takes lateral doublet info from flight test and runs
% through
% the extracted transfer function model to verify accuracy of model

clear

load rolldoub.mat

time=input('What is the duration of the input file in seconds?')
samp=input('What is the sample rate of the input file?')

% subtract mean of aircraft inputs and rates
acprate=[TIME DR01S-mean(DR01S)];
acpmix=[TIME DM01*.02065-mean(DM01*.02065)];

% Set transfer function parameters from NAVFIT
nump=[47.5722];
denp=[1 9.0304 40.1855];

% Run simulation to verify model and aircraft rate outputs
options=simset('FixedStep',1/samp);
[t,x,y]=sim('rollcheck',time,options);

% Filter rate outputs
[b,a]=ellip(3,0.2,40,[.1273 1.5915]*2/samp);
acpratef=filter(b,a,acprate);
rollratecheckf=filter(b,a,rollratecheck);

% Plot both aircraft and model rates
plot(TIME(1:time*samp),acpratef(1:time*samp,2),TIME(1:time*samp),rollra
techeckf(1:time*samp),'--')
legend('Filtered A/C Roll Rate','Filtered TF Model Roll Rate')
xlabel('Time (sec)');ylabel('Roll Rate (deg/sec)');grid on
title('Data from flight 189.05 - Roll Doublet')
```

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
8725 John J. Kingman Rd., Suite 0944
Ft. Belvoir, VA 22060-6218
2. Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Rd.
Monterey, CA 93943-5101
3. Dr. Mark Tischler 2
U.S. Army/NASA Rotorcraft Division
Ames Research Center
Mail Stop 210-5
Moffett Field, CA 94035-1000
4. Mr. Chris L. Blanken 1
U.S. Army/NASA Rotorcraft Division
Ames Research Center
Mail Stop 210-5
Moffett Field, CA 94035-1000
5. Prof. Max F. Platzer, Code AA/Pl 1
Chairman, Department of Aeronautics and Astronautics
Naval Postgraduate School
Monterey, CA 93943-5000
6. Prof. E. R. Wood, Code AA/Wd 2
Department of Aeronautics and Astronautics
Naval Postgraduate School
Monterey, CA 93943-5000
7. Commander, Naval Air Forces Atlantic 1
Attn: LCDR Steven J. Labows, Code N421C1
1279 Franklin St.
Norfolk, VA 23511-2494
8. Mr. Kurt Long 1
Ames Research Center
Mail Stop 260-1
Moffett Field, CA 94035-1000

- 9. Mr. Thomas H. Lawrence 1
 Sikorsky Aircraft Corporation
 6900 Main Street
 Stratford, CT 06601-1381

- 10. Mr. Ronald Duval..... 1
 President, Advanced Rotorcraft Technology, Inc.
 1685 Plymouth Street #250
 Mountain View, CA 94043

- 11. Dr. Dev Banerjee..... 1
 Director of Engineering
 Bldg. 530, MS B111
 The Boeing Company
 Mesa, AZ 85215

- 12. Mr. Andrew W. Kerr 1
 U.S. Army
 Director, Aeroflightdynamics Directorate
 Ames Research Center
 Moffett Field, CA 95120

- 13. Prof. Louis Schmidt, Code AA/Sc 1
 Department of Aeronautics and Astronautics
 Naval Postgraduate School
 Monterey, CA 93943-5000