NASA-CR-178058
19860015674

NASA Contractor Report 178058

AN EMPIRICAL STUDY OF FLIGHT CONTROL SOFTWARE RELIABILITY

Janet R. Dunham
John L. Pierce

Software Research and Development
Center for Digital Systems Research
Research Triangle Institute
Research Triangle Park, North Carolina 27709

Contract NAS1-16489
Task Assignment No. 22
March 1986

NASA
National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

NASA Contractor Report 178058

AN EMPIRICAL STUDY OF FLIGHT CONTROL SOFTWARE RELIABILITY

Janet R. Dunham
John L. Pierce

Software Research and Development
Center for Digital Systems Research
Research Triangle Institute
Research Triangle Park, North Carolina 27709

Contract NAS1-16489
Task Assignment No. 22

March 1986

N 86-25145#

TABLE OF CONTENTS

Page No.

LIST OF TABLES

Page No.

LIST OF PLOTS

ACKNOWLEDGEMENT

The authors wish to acknowledge the technical direction given by G. Earle Migneault of NASA - Langley Research Center. We also thank Don Rich, Sam McBride, and Jay Dunn for their perserverance in the development of the N-VERSION CONTROLLER. Finally, we thank Brian Lupton of the Flight Control Systems Division of NASA-LaRC who was the contract monitor for this research project and our manager Jim Clary for his continued support.

1. INTRODUCTION

1.1. Background

1.1.1. Reliability Goals for Flight-Critical Software

Examples of digital computers being used in increasingly critical functions in flight management systems are the navigation, guidance, and energy management system for the Boeing 757/767[1,2] commercial jet transport series, the slat and flap control system for the Airbus Industries A310,[3] and the flight control system for the Grumman X29A.[4] Since software failures can be equally as hazardous as hardware failures, the use of software to perform critical functions of these flight management systems necessitates demonstrating that the flight management software complies with Part 25 of the Federal Aviation Regulations. Toward this end, DO-178, published by the Radio and Technical Commission for Aeronautics (RTCA),[5] provides guidance on techniques and methods that may be used for building and certifying reliable flight control software. The FAA Advisory Circular 25.1309-1 [6] which references DO-178 defines the software reliability requirements by specifying three overlapping quantitative ranges which are to be interpreted as the allowable risk for an hour of flight time based on a flight of mean duration for the type of airplane being certified. These ranges are associated with the frequency with which failure events may be expected to occur several times during the operational life of each airplane (i.e., probable events that occur with a probability on the order of 1×10^{-5}or greater). The failure events may not be expected to occur during the total operational life of a random single airplane of a particular type but may be expected to occur during the total operational life of all airplanes of a particular type (i.e., improbable events that occur with a probability on the order of 1×10^{-5} or less). These failure events are so unlikely that they need not be considered to occur ever unless engineering judgement would require their consideration (i.e., extremely improbable events that occur with a probability on the order of 1×10^{-9}or less).

1.1.2. NASA-LaRC Software Reliability Research Goals

The software reliability research sponsored by NASA-LaRC focuses on the development of a credible method for predicting operational reliability — that is, predicting the improbability that the system will fail due to residual faults remaining in the software. [7] It is these residual faults, which surface infrequently, that cause the rare event or extremely improbable failures. As evidenced by the first well-publicized Space Shuttle software bug, the failure of the initialization logic in J. Garman's words resulted from a "very small, very improbable, very intricate, and a very old mistake."[8] This bug typifies the rare and convoluted combination of events which cause carefully developed software to fail.

Although considering all faults is important in reliability prediction, the most probable faults are often eliminated using the software quality assurance methods described in RTCA DO-178. The research thus concentrates on the development of a method which yields quantitative assurance that the aggregate probability of the remaining, less probable faults, constitutes an acceptable risk. Accordingly, the System Validation

Branch of NASA - Langley Research Center has used a probability value of 10^{-9} for a ten hour flight as an informal standard in the construction of a credible reliability prediction method for validating critical software.[9] To date no known software has been validated to that extent.

1.2. Study Objectives

The construction of a credible reliability prediction method for critical software is hindered by our lack of knowledge about the underlying nature of the software failure process. This lack of knowledge contributes to our inability to eradicate or tolerate faults and to our lack of confidence in the extent to which we have approximated the goal of minimal defects, i.e., achieving a reliability of less than 1×10^{-9} for a ten hour flight. The study was undertaken to provide data for use in developing a model for predicting the reliability of software in systems with feedback, namely flight control. In particular, we were interested in the software error burst phenomenon[10] and in gauging the effect of these bursts on the behavior of the system. A software error burst is a sequence of observations of erroneous outputs which are repeated manifestations of latent software faults that due to the memory of the system and the correlated nature of the inputs can eventually lead to a system failure. The frequency and duration of these bursts may severely effect the operational reliability of process control software, and consequently may be a critical reliability prediction parameter.

1.3. Related Work

The flight control software tested as a part of this study was developed in a manner which emulated a realistic software development effort. It was developed in a software research and development laboratory at RTI by three moderate to advanced skill level programmers using a remote link to the computational facilities of NASA's AIRLAB at Langley Research Center. Each of the programmers was given two programming tasks which involved implementing a launch interceptor condition module for a radar tracking system and a pitch axis module for a PA28 flight control system. The specifications for both modules had been written in English by a senior analyst who had also written and extensively tested a fourth or comparison version of each application. The programming activity was managed in a conventional fashion with the exception that the programmers were not permitted to discuss their code with anyone other than their manager or the senior analyst who was responsible for answering all specification questions. The programmers were instructed to optimize the reliability of their code. Additional information about the software development activity can be found in NASA CR-172553, *An Experiment in Software Reliability*.[11]

1.4. Summary of Study and Conclusions

The development and testing of the pitch axis control module for a flight control system was undertaken to provide data for use in constructing a model for predicting the reliability of software in systems with feedback. In particular, the study was undertaken to investigate the software error burst phenomenon[10] and to gauge the effect of these bursts on system behavior. A software error burst is a sequence of erroneous outputs due to the memory of the system and a correlated nature of the inputs. The frequency and duration of these bursts may severely effect the operational reliability of process control software and, consequently, may be a significant reliability prediction parameter.

To investigate the software error burst phenomenon, three pitch axis control modules were implemented for a PA28 aircraft. Input to the control law modules are the pitch command and the current pitch of the aircraft. Output from the control law is the elevator deflection command. The specification, written in English and provided to the programmers for coding, contained three operating modes of increasing complexity. A one axis, single input, single output simulation of the aircraft comprised the PA28 aircraft response model.

The three implementations were tested in parallel with over 14 million pitch commands (about 78,000 flight hours) having varying levels of additive input and feedback noise. This testing surfaced four software faults in an implementation of the pitch axis control law. This small number of detected faults may be due to the overly simple example chosen for the control law software and the programmer's ability to exploit this simplicity in validating it. As software components of the flight control systems increase in functionality and complexity, the full implication of software errors remains to be seen.

2. THE SYSTEM UNDER STUDY

2.1. The PA28 Pitch Axis Control Module

A survey of flight control software systems [12,13,14,1,4,15,16,17,2,3] which could be used for this study culminated with the decision to select a system for which we had both the control laws and a description of the aircraft response model. As a result, the PA28 aircraft was chosen because its aircraft response model was known and relatively simple to code. The pitch axis control module was chosen as reliable control of the pitch axis is safety-critical during landing. Loss of control of the elevator during final approach will almost certainly result in a catastrophic accident. The pitch axis control law is typically computed first and used as input to the control laws for the longitudinal, axes, although, in a few flight control systems it is computed in parallel with the roll axis.

Appendix A contains the specification of the pitch axis control law provided to the three programmers for independent development. The specification requires that the pitch axis control law operate in nine modes. The modes correspond to the use of Proportional (P), Proportional/Derivative (PD), and Proportional/Integral/Derivative (PID) control strategies and to different computations of the error term. A fourth module was developed and extensively tested for use as a comparator. The pitch axis module specified constitutes a simplified pitch axis control system as it ignores voting to handle sensor failures and coupling with an analog backup system.

2.2. The PA28 Aircraft Pitch Axis Response Model

A single input-single output aircraft response model defines the PA28 aircraft's response about the pitch axis. It was defined in the frequency domain and implemented in the time domain using Tustin's approximation. In the frequency domain, the aircraft response in pitch is given by the following transfer function

$$\frac{\theta_A}{\delta_E^N}(s) = \frac{\alpha_1 s^2 + \alpha_2 s + \alpha_3}{\beta_1 s^4 + \beta_2 s^3 + \beta_3 s^2 + \beta_4 s + \beta_5}$$

and the aircraft response in pitch rate is given by

$$\frac{\dot{\theta}_A}{\delta_E^N}(s) = s\,\frac{\theta_A}{\delta_E^N}(s)$$

An elevator transfer function of the form $\frac{\delta_E^N}{\delta^N}(s) = \frac{50}{s\,(s+50)}$, is used.

where

θ_A is the aircraft response in pitch.

$\dot{\theta}_A$ is the aircraft response in pitch.

δ^N is the elevator deflection for the N-version system after voting and prior to the application of the elevator transfer function.

δ_E^N is the elevator deflection for the N-version system after the application of the elevator transfer function.

α_1 , α_2 , α_3

and

β_1 , β_2 , β_3 , β_4 , β_5. are model coefficients

and

s is the Laplace domain variable.

3. TEST PROCEDURE AND RESULTS

3.1. Configuration of the N-VERSION CONTROLLER

The three independently developed software implementations of the pitch axis control law are executed in parallel using the N-VERSION CONTROLLER[11] shown in Figure 1. This CONTROLLER relies on the technique of n-version programming to detect program errors. (N-version programming, first widely publicized by Avizienis[18] and recently examined by Knight, Leveson, and St. Jean[19] involves n programmers independently coding the problem from the same specification). All program outputs are compared pairwise during this testing. Whenever an output inequality occurs which is outside the range specified, the testing is halted, and the faulty implementation is identified, analyzed, and corrected. Using n-version programming for error detection avoids reliance on a comparison version to determine output correctness.

The N-VERSION CONTROLLER simulates the execution of the pitch axis control system and provides both dynamic and static viewing of the system behavior through an interface program. The CONTROLLER can be used to control the pitch axis simulation for combinations of the following configurations:

- Extensively Tested System (ETS) flying
- N-Version System (NVS) flying
- ETS monitoring NVS flying.

These configurations can have noise added to the pitch commands generated and to the aircraft response in pitch.

3.2. Test Initialization

3.2.1. Pitch Command Generation

The pitch command waveform selected to test the control law software can be propogated as either a sine wave or a generalized square wave. The sine wave is of the form

$$\beta(t) = A\sin(Bt+\gamma) + k$$

where β, A, B, γ, and k are parameters specified by the experimenter. The generalized square wave has six specifiable parameters which characterize the waveform; 1: lead time, 2: rise time, 3: duration, 4: fall time, 5: height, and 6: vertical displacement. The sine waveform is denoted by sine (A,B,γ,k) and the square waveform is denoted by square (1,2,3,4,5,6).

For the purpose of this study, the coefficients of the input square wave and the sine wave are chosen to have a maximum rate of change in pitch of 30 degrees over 40 timesteps or 2 millisecs. These requirements yield a square waveform of square (20,40,20,40,60,-30) and a sine waveform of sine (30,.16,0,0) as show in Plots 1 and 2 respectively.

The noise which can be added to the pitch command and the pitch feedback is sampled from a truncated skewed Normal (μ_1,σ_1^2) distribution with $\frac{\mu_1}{\sigma_1^3} \neq 0$ and a truncated skewed Normal (μ_2,σ_2^2) distribution with $\frac{\mu_2}{\sigma_2^3} \neq 0$ respectively.

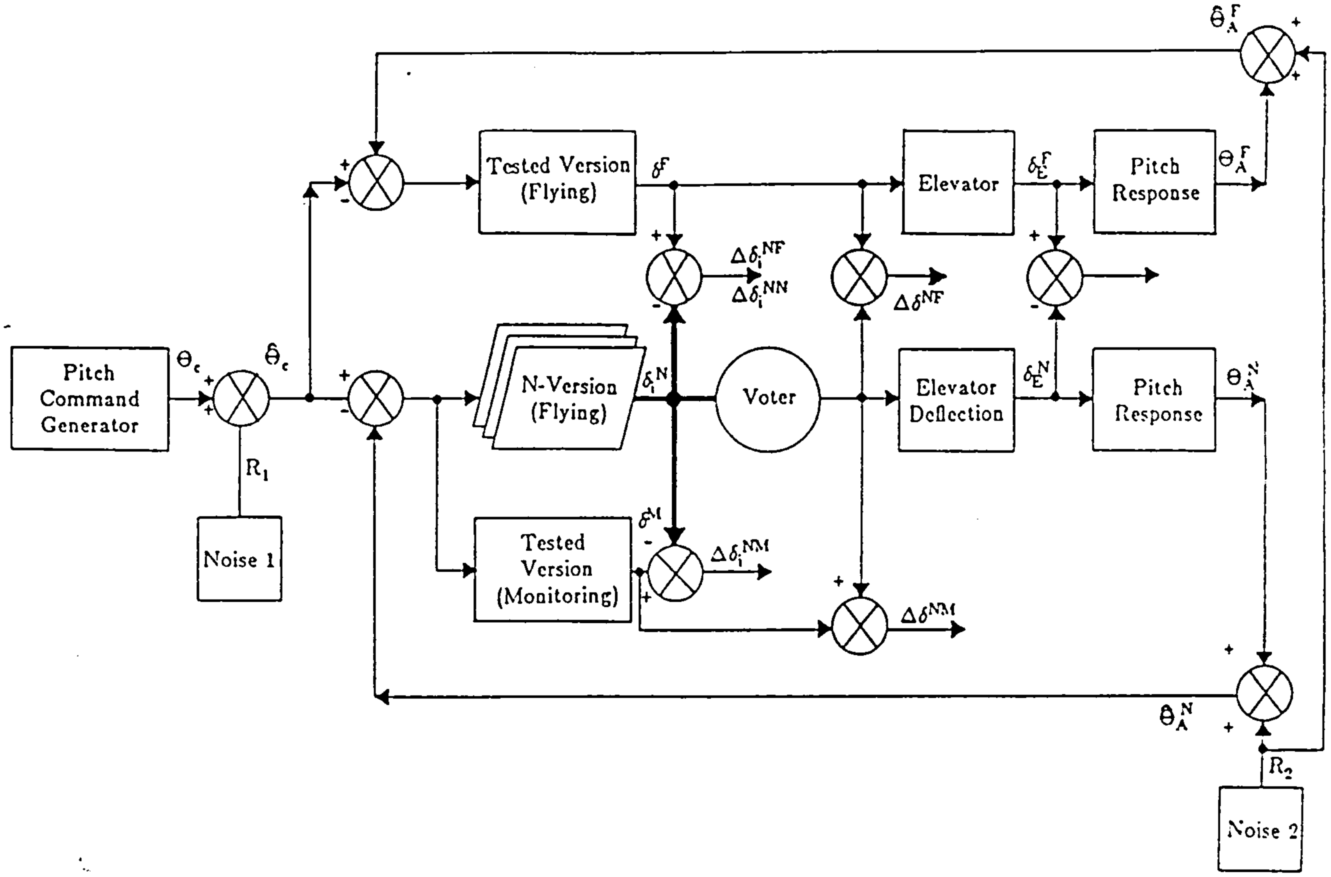


FIGURE 1. CONFIGURATION OF THE N-VERSION CONTROLLER

where

$\theta_C(n)$ - Pitch command at time step n for the ETS and NVS SYSTEMs prior to the addition of random noise.

$R_1(n)$ - Random noise added to $\theta_{C(n)}$.

$\hat{\theta}_C(n)$ - Pitch command at time step n for the ETS and NVS SYSTEMS after the addition of random noise.

$\delta^F(n)$ - Deflection of elevator at time step n for ETS flying the aircraft prior to application of the elevator transfer function.

$\delta_i^N(n)$ - Deflection of elevator at time step n for version i of NVS prior to vote and application of elevator transfer function.

$\delta^M(n)$ - Deflection of elevator at time step n for the ETS monitoring the NVS prior to application of the elevator transfer function.

$\Delta\delta_{ij}^{NN}(n)$ - Difference between $\delta^N(n)$ for all unordered NVS pairs (i.e. $(\delta_i^N(n),\delta_j^N(n))$ for all i, j ϵ [0,2]).

$\Delta\delta_i^{NF}(n)$ - Difference between $\delta_i^N(n)$ for all NVS pairs (i.e. $(\delta_i^N(n),\delta^F(n))$ for all i ϵ [0,2]).

$\Delta\delta_i^{NM}(n)$ for all NVS pairs (i.e. $(\delta_i^N(n),\delta^M(n))$ for all i ϵ [0,2]). - Difference between $\delta_i^N(n)$

$\delta^N(n)$ - Deflection of elevator at time step n for NVS flying the aircraft after voting and prior to application of the elevator transfer function.

$\Delta\delta^{NM}(n)$ - Difference between $\delta^N(n)$ and $\delta^M(n)$

$\Delta\delta^{NF}(n)$ - Difference between $\delta^N(n)$ and $\delta^F(n)$

$\delta_E^F(n)$ - Deflection of elevator at time step n for ETS flying the aircraft after application of the elevator transfer function.

$\delta_E^N(n)$ - Deflection of elevator at time step n for NVS after application of elevator transfer function.

$\Delta\delta_E^{NF}(n)$ - Difference between $\delta_E^N(n)$ and $\delta_E^F(n)$

$\theta_A^F(n)$ - Pitch of aircraft at time step n for the ETS flying the aircraft prior to adding random noise.

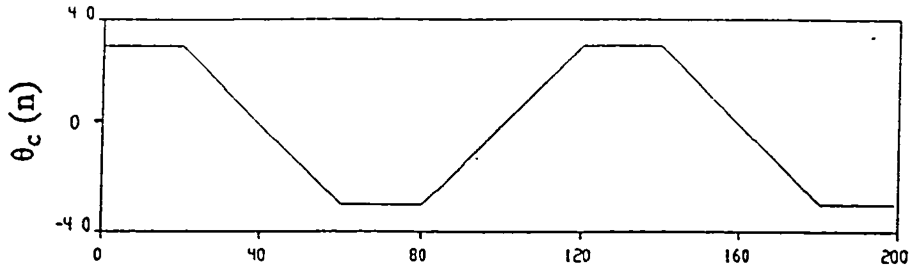
$\theta_A^N(n)$ - Pitch of aircraft at time step n for the NVS flying the aircraft prior to adding random noise.

$R_2(n)$ - Random noise added to θ_A^F and θ_A^N at time step n.

$\hat{\theta}_A^F(n)$ - Pitch of aircraft at time step n for the ETS flying the aircraft after adding random noise.

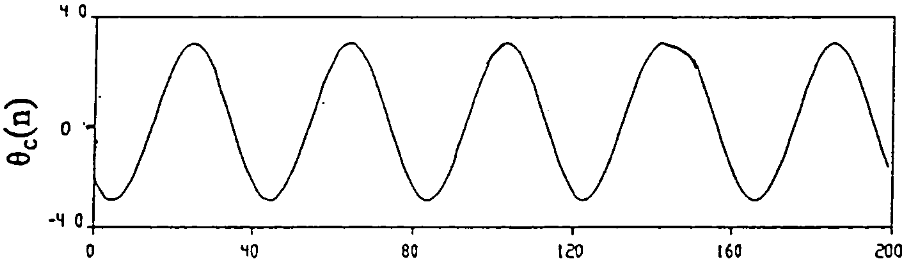
$\hat{\theta}_A^N(n)$ - Pitch of aircraft at time step n for the NVS flying the aircraft after adding random noise.

PLOT 1. SQUARE (20,40,20,40,60,-30)



Time Step t

PLOT 2. SINE (30,.16,0,0)



Time Step t

3.2.2. Tuning the N-VERSION CONTROLLER

Tuning the N-VERSION CONTROLLER involved (i) testing of the ETS module in conjunction with the aircraft response model, (ii) testing the control law software written by programmer 1 in conjunction with the aircraft response model, and (iii) testing of the NVS in conjunction with the aircraft response model. The first two test efforts were conducted as the ETS and the independent implementations of the control law are implementations of different forms of the same control law algorithm in that the ETS control law does not have the mode switching capability and it operates in Proportional-Integral-Derivative mode only. The above testing resulted in near optimal performance of the software to control the pitch axis by setting of the aircraft response model coefficients as follows:

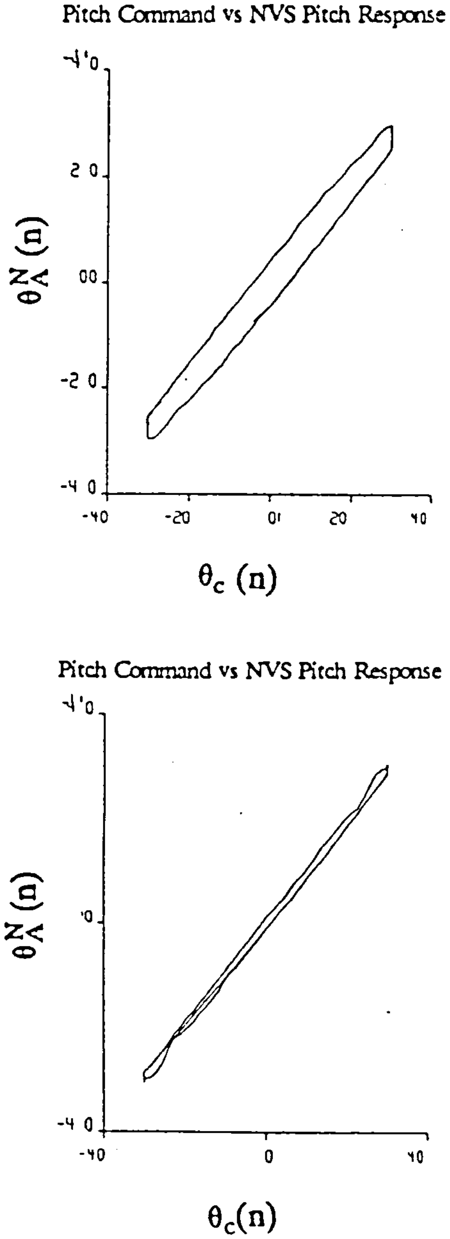
$$\alpha_1 = 17.5; \ \alpha_2 = 39.4; \ \alpha_3 = 17.0;$$

$$\beta_1 = 1.02; \ \beta_2 = 7.01; \ \beta_3 = 18.7; \ \beta_4 = 8.6; \ \beta_5 = 2.84;$$

3.2.3. Tuning the NVS Control Law Coefficients

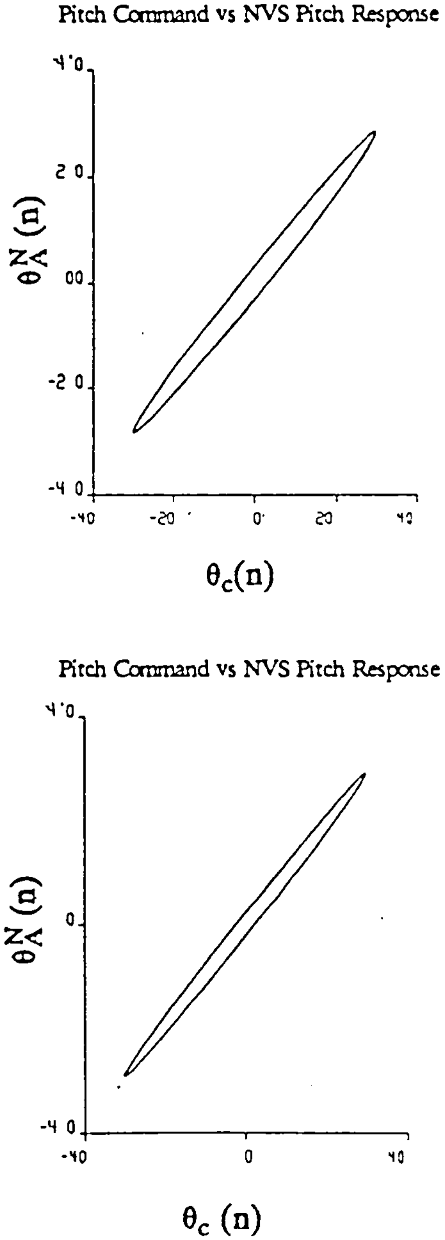
The control law coefficients were tuned to minimize the amount of overshooting and undershooting of the pitch response in the absence of input and feedback noise when the NVS is flying the aircraft. Table 1 shows the results of executing the NVS system with different values of the control law parameters for the square (20,40,20,40,60,-30) and sine (30,.16,0,0) waveforms. The values shown are estimates derived from visual inspection of the corresponding graphs. Plots 3 and 4 show the system performance for the tuning runs using the square (20,40,20,40,60,-30) and sine (30,.16,0,0) waveforms respectively. The ETS control law was tuned during the system tuning runs.

PLOT 3. TSQ4 SYSTEM PERFORMANCE


Pitch Command vs NVS Pitch Response

Pitch Command vs NVS Pitch Response

PLOT 4. TSN3 SYSTEM PERFORMANCE


Pitch Command vs NVS Pitch Response

Pitch Command vs NVS Pitch Response

TABLE 1. NVS CONTROL LAW RUNS OF LENGTH 1,000 INPUT CASES

Run	a_0	a_1	a_2	a_3	$\epsilon_\theta^N(n)$	$E_\theta^N(n)$	$\theta_A^N(n)$	$\delta^N(n)$	Comments
tsq1	.000	.167	.084	.042	(-180,180)	(-300,300)	(-200,200)	(-30,30)	high freq. oscillation in θ_A^N, overshoots θ_A^N changes sign every time step
tsq2	.000	.167	.042	.021	(-30,30)	(-200,200)	(-50,50)	(-20,20)	high freq. oscillation in θ_A^N changes sign alter. time steps
tsq3	.000	.167	.021	.042	(-200,200)	(-500,500)	(-200,200)	(-110,110)	low freq. oscillation in θ_A^N changes sign every time step
tsq4•	.000	.167	.021	.042	(-7,7)	(-125,125)	(-32,35)	(-7,10)	low freq. oscillation in θ_A^N at peak and low amplitude
tsq5	.000	.167	.010	.021	(-10,10)	(-200,200)	(-29,29)	(-6,8)	low freq. oscillation in θ_A^N at peak and low amplitude
tsq6	.000	.167	.005	.021	(-10,10)	(-200,200)	(-29,29)	(-6,8)	low freq. oscillation in θ_A^N at peak and low amplitude
tsq7	.000	.200	.013	.025	(-10,10)	(-200,200)	(-35,33)	(-10,10)	low freq. oscillation in θ_A^N at peak and low amplitude
tsq8	.000	.200	.006	.025	(-10,10)	(-200,200)	(-32,33)	(-8,10)	low freq. oscillation in θ_A^N at peak and low amplitude
tsn1	.000	.167	.025	.042	(-12,12)	(-75,75)	(-27,30)	(-6,7)	overshoots θ_A^N low freq. oscillation
tsn2	.000	.200	.013	.025	(-12,12)	(-75,75)	(-25,28)	(-6,7)	undershoots θ_A^N no oscillation
tsn3•	.000	.167	.042	.084	(-10,10)	(-50,50)	(-32,34)	(-7,9)	overshoots θ_A^N low freq. oscillation

where

tsq1 through tsq8	are the square wave tuning runs
tsn1 through tsn3	are the sine wave tuning runs
a_0, a_1, a_2, a_3	are the control law coefficients
ϵ_θ^N	is the current error in $\hat{\theta}$ at time step n for NVS flying the aircraft.
E_θ^N	is the cummulative error in pitch of NVS at time step n.
θ_A^N	is the pitch of aircraft at time step n for the NVS flying the aircraft prior to adding random noise.
δ^F	is the deflection of elevator at time step n for ETS flying the aircraft prior to application of the elevator transfor.

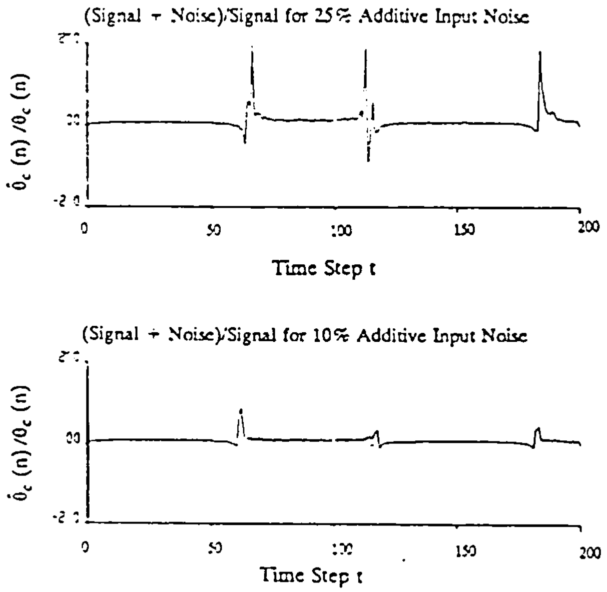
Based on the results of the tuning runs, the control law coefficients for the square wave stress test runs were set at $a_0=0.000$; $a_1=0.167$; $a_2=0.032$; $a_4=0.084$; (based on tuning run tsq4) and for the sine wave stress test runs $a_0=0.000$; $a_1=0.167$; $a_2=0.021$; $a_3=0.042$; (based on tuning run tsn3).

3.3. Stress Test Results

Stress testing entailed executing the system in Proportional-Integral-Derivative (PID) mode 7 (see Appendix A) with the goal of detecting software failures under worst case input conditions. The noise distributions used during this testing were selected to simulate both continuous turbulence and discrete incremental gust loads. The stress test runs were made with the square waveform and the sine waveform and with and without input noise on the pitch command and the pitch response as depicted in Table 2. Plots 5 and 6 show the signal and noise to signal ratio for the square (20,40,20,40,60,-30) and sine (30,.16,0,0) stress test runs respectively.

During the stress tests if the absolute range of the pairwise comparisons of all δ_i^N exceeded the limit specified, the N-VERSION CONTROLLER logs the system state for the next two hundred test cases inclusive of the failure causing case. The control law implementations are then scrutinized for the presence of a software fault and the impact of that fault on the system performance is analyzed.

PLOT 5. SQUARE WAVE RESULTS



PLOT 6. SINE WAVE RESULTS

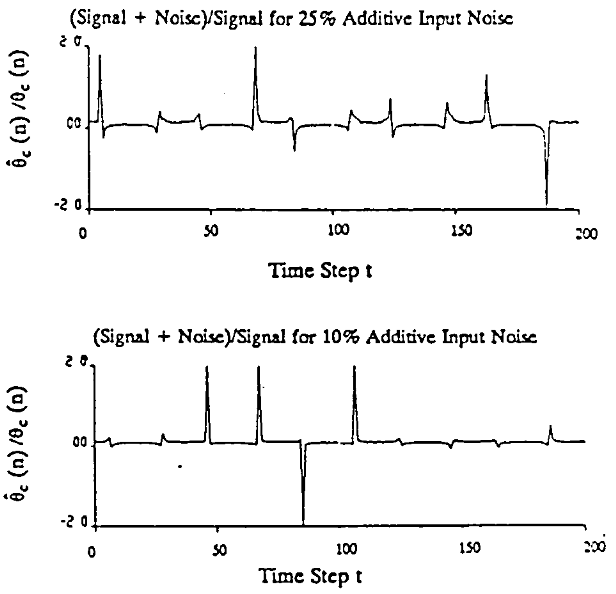


TABLE 2. STRESS TEST RUNS OF LENGTH 1,000,000 INPUT CASES

Run	$R_1(n)$	$r_{1L}(n)$	$r_{1U}(n)$	$R_2(n)$	$r_{2L}(n)$	$r_{2U}(n)$	Comments
ssq 1	N(0.0,0.0)	0.0	0.0	N(0.0,0.0)	0.0	0.0	0% input & output noise
ssq 2	N(3.0,1.0)	2.0	4.0	N(0.0,0.0)	0.0	0.0	10% input noise
ssq 3	N(0.0,0.0)	0.0	0.0	N(3.0,1.0)	2.0	4.0	10% output noise
ssq 4	N(3.0,1.0)	2.0	4.0	N(3.0,1.0)	2.0	4.0	10% input & output noise
ssq 5	N(7.5,2.5)	5.0	10.0	N(0.0,0.0)	0.0	0.0	25% input noise
ssq 6	N(0.0,0.0)	0.0	0.0	N(7.5,2.5)	5.0	10.0	25% output noise
ssq 7	N(7.5,2.5)	5.0	10.0	N(7.5,2.5)	5.0	10.0	25% input & output noise
ssn 1	N(0.0,0.0)	0.0	0.0	N(0.0,0.0)	0.0	0.0	0% input noise output noise
ssn 2	N(3.0,1.0)	2.0	4.0	N(0.0,0.0)	0.0	0.0	10% input noise
ssn 3	N(0.0,0.0)	0.0	0.0	N(3.0,1.0)	2.0	4.0	10% output noise
ssn 4	N(3.0,1.0)	2.0	4.0	N(3.0,1.0)	2.0	4.0	10% input & output noise
ssn 5	N(7.5,2.5)	5.0	10.0	N(0.0,0.0)	0.0	0.0	25% input noise
ssn 6	N(0.0,0.0)	0.0	0.0	N(7.5,2.5)	5.0	10.0	25% output noise
ssn 7	N(7.5,2.5)	5.0	10.0	N(7.5,2.5)	5.0	10.0	25% input & output noise

Table 3 shows the maximum range in the elevator deflection angles computed by each implementation of the control law software for the various stress test runs. Although the actual values of the elevator deflections varied among the independent implementations they remained in a range of 2.5, as shown in Table 3, which was specified as an acceptable performance parameter during the system tuning runs. The range specification is somewhat arbitrary in that a tight bound on the range increases the rate with which we observe false alarms, and a lenient bound increases the probability of non-detection.

TABLE 3.
APPLICATION TASK ELEVATOR
COMMAND DIFFERENCES AND PITCH RANGE

RUN ID	MAX DIFFERENCE	PITCH RANGE
ssq 1	0.6	[-31.7,34.3]
ssq 2	0.6	[-31.7,37.2]
ssq 3	0.6	[-35.0,34.8]
ssq 4	0.6	[-34.7,40.4]
ssq 5	0.7	[-34.0,48.8]
ssq 6	0.5	[-45.1,44.1]
ssq 7	0.6	[-39.9,55.7]
ssn 1	0.7	[-32.1,34.2]
ssn 2	0.7	[-33.6,39.6]
ssn 3	0.7	[-36.4,36.6]
ssn 4	0.8	[-37.4,43.0]
ssn 5	1.3	[-39.2,56.2]
ssn 6	1.3	[-49.5,48.8]
ssn 7	2.0	[-51.7,66.1]

3.4. Case Analysis Results

Since we were uncertain if the range in elevator deflection commands were a function of mismodeling or if a software error was occurring, (This appears to be a common problem with performance testing of process control software.)[20] we investigated the elevator deflections for 1000 time steps by executing the NVS system using the ssq1 run parameters, setting the permissible range of the elevator deflection commands computed by the three implementations for each time step (i.e., $\Delta\delta_{ij}^{NN}(n)$) to 0, and analyzing the test case when non-zero ranges occurred. The analysis identified four faults in an implementation of the control law, as shown in Table 4. The fault made in computing the derivative term in Modes 4 and 7 was corrected. A re-execution of the ssq1 run resulted in numerical agreement (to the least significant bit) of the elevator deflection command output from the control laws.

TABLE 4. DESCRIPTION OF SOFTWARE FAULTS PRESENT IN AT3

Modes Effected	Correct Term	Term Used
4,7	$\dfrac{\theta_C(n)-\theta_A(n)}{t(n)-t(n-1)}$	$\dot{\theta}_C(n)-\dot{\theta}_A(n)$
5,8	$\dfrac{\dot{\theta}_C(n)-\dot{\theta}_A(n)}{t(n)-t(n-1)}$	$\dfrac{\dot{\theta}_C(n)-\dot{\theta}_C(n-1)-\dot{\theta}_A(n)-\dot{\theta}_A(n-1)}{t(n)-t(n-1)}$
3,6,9	$\dot{\theta}_C(n)-\dot{\theta}_A(n)$	$\dfrac{\dot{\theta}_C(n)-\dot{\theta}_A(n)}{t(n)-t(n-1)}$
6,9	$\dfrac{\dot{\theta}_C(n)-\dot{\theta}_A(n)}{t(n)-t(n-1)}$	$\dfrac{\dot{\theta}_C(n)-\dot{\theta}_A(n)-\dot{\theta}_C(n-1)+\dot{\theta}_A(n-1)}{t(n)-t(n-1)}$

where

n is the time step index

$t(n)$ is the clock time at step n

$\theta_C(n)$ is the pitch command at step n

$\theta_A(n)$ is the actual pitch of the aircraft at step n

$\dot{\theta}_C(n)$ is the change in pitch command, i.e. $(\theta_C(n)-\theta_C(n-1))$

$\dot{\theta}_A(n)$ is the change in actual pitch, i.e. $(\theta_A(n)-\theta_A(n-1))$

AT3 is the third implementation of the control law

4. CONCLUDING REMARKS

N-version testing of three implementations of a PA28 pitch axis control law with over 14 million pitch commands (about 78,000 flight hours) having varying levels of additive input and feedback noise surfaced only four software faults in an implementation, thus precluding the analysis of software error bursts. This small number of detected faults may be due to the overly simple example chosen for the control law software and the programmers' ability to exploit this simplicity in validating it. As software components of the flight control systems increase in functionality and complexity, the full implication of software errors remains to be seen.

5. REFERENCES

References

1. R. E. Spradlin, *Boeing 757 and 767 Flight Management System,* Boeing Commercial Airplane Company (November 20-21, 1980). Presented at the RTCA 1980 Technical Symposium and Annual Assembly Meeting, Washington, D.C.

2. K. M. Hornback, "Development Tools - Case Study for Large Systems," *Proceedings of the AIAA/IEEE 6th Digital Avionics Systems Conference*, American Institute of Aeronautics and Astronautics, (December 1984).

3. R. Troy and C. Baluteau, "Assessment of Software Quality for the AIRBUS A310 Automatic Pilot," *Proceedings of the Fifteenth Annual International Symposium on Fault-Tolerant Computing*, IEEE Computer Society Press, (June 1985).

4. J.T. Murray et al., "DS-28703-01 Machine Language Program, FSW, NH3500AN," *Honeywell Avionics Division*, (June 1982).

5. "RTCA/DO178 Software Considerations in Airborne Systems and Equipment Certification," *Radio Technical Commission for Aeronautics Secretariat*, (November 1981).

6. *Advisory Circular 25.1309-1 System Design Analysis,* U.S. Department of Transportation Federal Aviation Administration, Washington, D.C. (September 1982).

7. J. R. Dunham and G. E. Migneault, "An Experiment in Software Reliability," Talk presented at Seventh Minnowbrook Workshop on Software Performance Evaluation, Blue Mountain Lake, New York (July 84).

8. John R. Garman, "The "Bug" Heard Round the World," *Software Engineering Notes* 6 pp. 3-10 ACM Sigsoft, (October 1981).

9. J. R. Dunham and J. C. Knight, eds., "Production of Reliable Flight Crucial Software: Validation Method Research for Fault-Tolerant Avionics and Control Systems Sub-Working-Group Meeting," NASA Conference Publication 2222, NASA (1982).

10. G. E. Migneault, "Emulation Applied to Reliability Analysis of Reconfigurable, Highly Reliable, Fault-Tolerant Computing Systems," *AGARD Conference Proceeding* **No. 261**(1980).

11. J. R. Dunham and J. L. Pierce, "An Experiment In Software Reliability," NASA CR 172553, NASA, Langley Research Center (March 1985).

12. K. J. Szalai, et al., *Digital Fly-By-Wire Flight Control Validation Experience,* NASA Technical Memorandum 72860 (December 1978).

13. Alfred Spector and David Gifford, "Case Study: The Space Shuttle Primary Computer System," *Communications of the ACM* **27**(9)(September 84).

14. J. M. Corney, "The Development of Multiple Redundant Flight Control Systems for High Integrity Applications," *Aeronautical Journal*, Marconi Avionics Limited, (October 1980).

15. David J. Martin, *Dissimilar Software in High Integrity Applications in Flight Controls,* Marconi Avionics Limited, Rochester, England (1983).

16. R. B. Smith, "Flight Clearance of the Jaguar Fly-By-Wire Aircraft (part II)," *The Royal Aeronautical Society,* (April 1982).

17. K. G. Wilkinson and M. L. Shooman, "Probabilistic Models for Software Reliability Prediction," *The Royal Aeronautical Society,* pp. 485-502 Academic Press, (1972).

18. A. Avizienis, "Fault Tolerance: The Survival .Attribute of Digital Systems," *Proceedings of the IEEE* 66(10)(October 1978).

19. J. C. Knight, N. G. Leveson, and L. D. St. Jean, "A Large Scale Experiment in N-Version Programming," *The 15th International Conference on Fault-Tolerant Computing Digest of Papers* **FTCS 15** pp. 135-139 IEEE Computer Society Press, (June 19-21, 1985).

20. Ronald R. Luman, "Practical Kalman Filter Software Performance Testing & Validation," *IEEE Transactions on Reliability* **R-33** pp. 219-226 (August 1984).

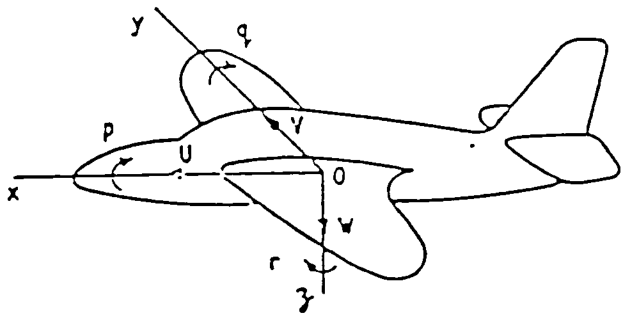
Appendix A. Pitch Axis Control Law Specification

1. BACKGROUND

a. Aircraft Control Surfaces And Axes Of An Aircraft In Flight

Aircraft control surfaces are divided into two groups; primary and secondary. The primary control surfaces are the ailerons which are located at the trailing edge of the wings, the elevators which are located at the rear portion of the horizontal tail assembly (in some aircraft the entire horizontal tail is movable), and the ruder which is the rear portion of the vertical tail assembly. The secondary control surfaces are the trim tabs, balance tabs, and servo tabs. They are used to reduce the force required to activate the primary control surfaces, and for trimming and balancing the airplane in flight. These tabs are in actuality small airfoils attached to,; or recessed into the trailing edge of the primary control surfaces.

There are three axes about which an aircraft rotates about whenever it changes its altitude with respect to the earth, or inertial, or moving axis coordinate system. These axes are the longitudinal, lateral, and vertical axes. Roll is motion about the longitudinal axis. This motion is controlled by the ailerons. Pitch is motion about the lateral axis. This motion is controlled by the elevators. Yaw is motion about the vertical axis. Yaw is controlled by the rudder.



The above diagram illustrates the axes of an aircraft in flight. The set of rectangular axes is oxyz where o is the center of gravity of the aircraft.

$\vec{Ox}$ is the longitudinal axis

$\vec{Oy}$ is the lateral axis

$\vec{Oz}$ is the vertical axis

U, V, W are the velocity components of the center of gravity along $\vec{Ox}$, $\vec{Oy}$, and $\vec{Oz}$ respectively.

P, q, r are the components of angular velocity of the axis frame Oxyz about $\vec{Ox}$,$\vec{Oy}$, and$\vec{Oz}$ respectively. Hence, p is the aircraft's angular velocity in roll, q is the aircraft's angular velocity in pitch, and r is the aircraft's angular velocity in yaw.

b. Flight Dynamics

1. Eqns. of Motion

Flight dynamics deals with the motion of an aircraft under the influence of forces. These forces are of six types.

(i) Inertia forces, arising from the mass distribution and linear and angular acceleration of the aircraft.

(ii) Aerodynamic forces and moments, depending on angular velocities of the aircraft (sometimes called rotary forces and moments).

(iii) Aerodynamic forces and moments depending on the linear velocities of the aircraft (sometimes called static forces and moments, since they depend on the altitude of the aircraft relative to the airstream and not on its angular velocities).

(iv) Aerodynamic forces and moments due to the application of controls (usually only the forces and moments due to control deflection are of importance; these are sometimes called static forces and moments due to controls).

(v) Gravitational forces, and

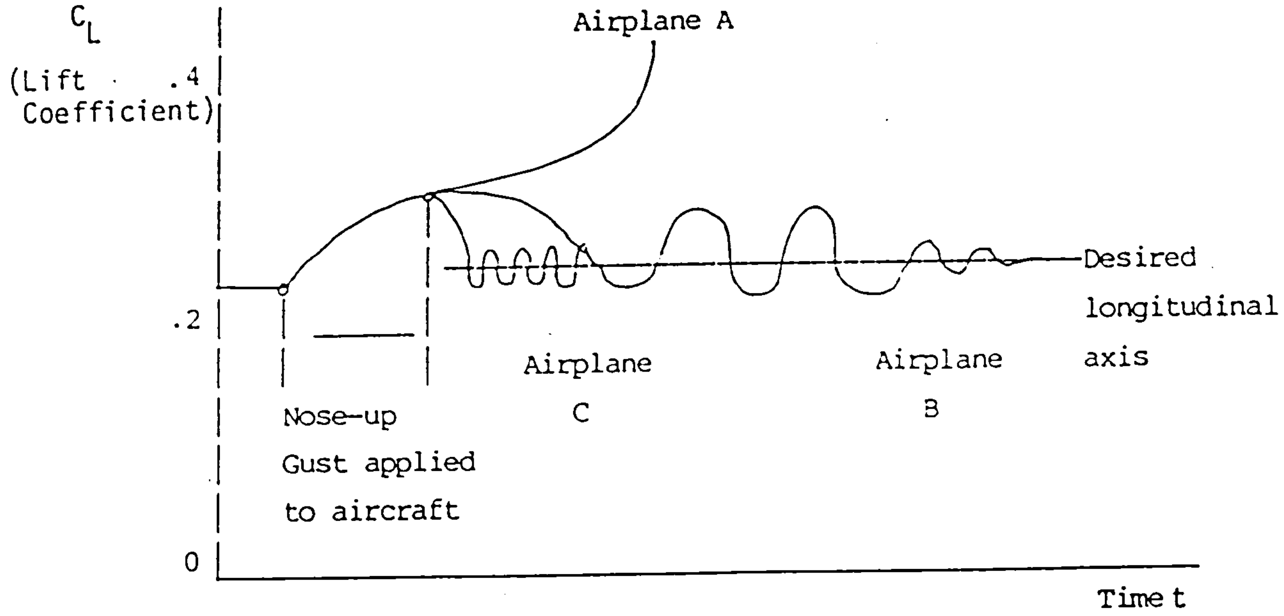
(vi) Propulsive forces.

The equations of motion of an aircraft can be completely determined by considering these forces. These equations constitute a set of non-linear differential equations, which are separated with respect to the lateral and longitudinal axes ana linearized. These equations can be found in texts on aircraft aerodynamics.

2. STABILITY

A dynamical system is said to be stable or possess stability, if, when slightly disturbed from a state of equilibrium or steady motion, it tends to return and remain in that state, the disturbance acting for only a finite time. In general, an aircraft must be both statically and dynamically stable. An aircraft has static stability if, immediately on being disturbed, it has a tendency to return to equilibrium conditions. An aircraft has dynamic stability if its motion subsequent to the initial disturbance is characterized by its decreasing amplitude as equilibrium is restored. Static stability is prerequisite to dynamic stability. The design of an aircraft must make it both statically and dynamically stable with respect to the three axes. Because symmetry keeps translation and rotation in the vertical plane from producing forces in the other planes, longitudinal and lateral-directional stability are considered separately.

To illustrate, assume that we are interested in the stability in pitch for three aircrafts. Stability in pitch represents longitudinal stability, i.e., stability about the airplanes' lateral axis.



NOTES: Airplane A is unstable and continually diverges.

Airplane B is both statically and dynamically stable, but the return to equilibrium takes too long.

Airplane C is acceptable because it returns to equilibrium in a relatively short time.

c. Artificial Stability and Automatic Control

The stability and response characteristics of an aircraft without automatic control, at a given speed and altitude, are determined completely by the design and distribution of mass of the aircraft. There exists, however, various mechanisms to optimize the aircraft's stability without changing the aircraft design. One way is by shifting the mass distribution, or center of gravity during various stages of flight (e.g.,shifting from subsonic to supersonic speeds). Another way is by automatic control of the control surfaces in a manner which compensates for one or more components of the flight disturbance. The goal for both these methods is to artificially improve the stability and response of the aircraft.

The automatic control systems utilized may be an open-loop systems in which input data, say Θ_i are fed into a servo-system which produces an output Θ_0. Θ_i could be a signal from a gyroscope or an accelerometer. The output Θ_0 might represent an elevator deflection. The automatic control system may be a closed-loop system or a system with feedback control. Closed-loop systems are applied in cases where the output of the open-loop system depends upon the deviation of the aircraft from the datum state. The essential requirement of a closed-loop system is that the error between the desired state and the existing state is constantly monitored.

This describes the detailed computer program specifications for the Pitch Axis Control Problem. It is anticipated that you will code at least three successive control strategies of increasing complexity. In order, these are:

STRATEGY	FORM
(1) Proportional:	$Y=a_0 + a_1x$
(2) Proportional-Derivative:	$Y=a_0 + a_1x + a_2\, dx/dt$
(3) Proportional-Integral-Derivative:	$Y=a_0 + a_1x + a_2\, dx/dt + a_3\int_0^t x\, dt$

where Y and the integral are to be computed, and the terms in x and the a_i are inputs. Specifications generally pertinent to all 3 control strategies and details of the first are contained herein; details for strategies 2 and 3 will be forthcoming. The overhead (parameter lists, common blocks, clock calls, initialization, etc.,) developed in the coding of strategy 1 will be directly applicable to strategies 2 and 3. Therefore, appropriate editing techniques should expedite development of 2 and 3.

3. SPECIFICATIONS

Language: FORTRAN

Variables: Standard FORTRAN Real*4 and Integer*4 for variables passed via common. Variables may be named as desired.

Module: A subroutine named PROB2 with no parameter list (as in PROB1, Launch Interceptor Problem).

I/O: Variables will be passed through labeled common blocks; PROB2 will generate no read/write operations except as necessary during development. There should be no I/O operations in the delivered module.

Files: The source code for PROB2 and any supporting subprograms should be in a file named prob2.for; any other subprograms or main "driving" programs not delivered should be maintained on separate files.

Acceptance: There will be no formal acceptance test as there was in the Launch Interceptor Problem; you may generate your own test data as you desire.

Data Validity: Assume all input data are valid; no validity tests are necessary.

INPUTS: COMMON/INPUTS/

POSITION	NAME	TYPE	COMMENTS
1	Initialize	I*4	IF = 1, zero-out all output variables and return.
2	Mode	I*4	IF = 0, automatic control is not in effect; return without computing; otherwise, determines the control strategy and the measurement variable.
3	t(n)	I*4	The current value of dimensionless time.
4	$\Theta_c(n)$	R*4	Desired value of pitch at current time-step, n.
5	$\Theta_A(n)$	R*4	Actual pitch of aircraft at current time-step, n.
6	$\dot{\Theta}_c(n)$	R*4	Desired pitch rate at current time-step, n.
7	$\dot{\Theta}_A(n)$	R*4	Actual pitch rate of aircraft at current time-step, n.
8	a_i	R*4	Coefficients; i = 1,2,...,5

OUTPUTS:
COMMON/OUTPUT/

1	$\delta(n)$	R*4	Elevator deflection angle, computed.
2	$\epsilon(n)$	R*4	Error in measurement quantity (as determined by mode).
3	$\Delta\Theta_A/\Delta t$	R*4	Pitch rate, computed.

Saved variables: COMMON/SAVED/

Note: The saved variable name should differ from the I/O variable name.

POSITION	NAME	TYPE	COMMENTS
1	$\Theta_c(n)$	R*4	Current Θ_c ; to become $\Theta_c(n-1)$ upon next entry of subroutine.
2	$\delta(n)$	R*4	Current computed elevator deflection; to become $\delta(n-1)$ upon next entry of subroutine.
3	$\Theta_A(n)$	R*4	Current Θ_A; to become Θ_A (n-1) upon next entry of subroutine.
4	t(n)	I*4[1]	Current value of dimensionless time; to become t(n-1) upon next entry of subroutine.
5	$\epsilon_\Theta(n)$	R*4	Current error in Θ; to become $\epsilon_\Theta(n-1)$ upon next entry of subroutine.
6	$\epsilon_{\dot\Theta}(n)$	R*4	Current error in $\dot\Theta$; to become $\epsilon_{\dot\Theta}(n-1)$ upon next entry of subroutine.
7	$\dot\Theta_c(n)$	R*4	Current $\dot\Theta_c$; to become $\dot\Theta_c(n-1)$ upon next entry of subroutine.
8	$\dot\Theta_A(n)$	R*4	Current $\dot\Theta_A$; to become $\dot\Theta_A(n-1)$ upon next entry of subroutine.
9	$\Sigma\epsilon_\Theta$	R*4	Cumulative sum of $\epsilon_\Theta \Delta t$ since $t = 0$.
10	$\Sigma\epsilon_{\dot\Theta}$	R*4	Cumulative sum of $\epsilon_{\dot\Theta} \Delta t$ since $t = 0$.

[1] The type specification for t(n) was omitted from the specification provided to the programmers.

4. ALGORITHMS

In any of the forms, the quantity x represents the measured error, ϵ , in the quantity being controlled which can be either pitch, Θ , or pitch rate, $\dot{\Theta} = d\Theta/dt$. The quantity y is the output from the controller and represents the aircraft elevator deflection δ.

The controller will compute x, the measured error. Whether it computes $x(\Theta)$ or $x(\dot{\Theta})$ will depend on the value of MODE. In addition, whether $\dot{\Theta}$ is furnished as an input or is to be computed as $(\Theta(n) - \Theta(n-1)/\Delta t)$ by the controller, will also be determined by MODE.

Table 1 shows the control strategy and measured error relationship as governed by values of MODE.

Table A - Control Law Mode			
Strategy	Error Term		
	$\Theta_C - \Theta_A$	$\dot{\Theta}_C - \dot{\Theta}_A$	$\frac{\Delta\Theta_C}{\Delta_t} - \frac{\Delta\Theta_A}{\Delta_t}$
P: Proportional	1	2	3
PD: Proportional/Derivative	4	5	6
PID: Proportional/Integral/Derivative	7	8	9

TABLE 1: Control Strategy to be employed and Measured Error to be Computed for Different Values of MODE

THE FOLLOWING STEPS SHOULD BE CODED:

Strategy 1, Proportional, $y = a_0 + a_1 x$

1. Test Initialize (see INPUTS comments)
2. Test Mode
3. COMPUTE X per mode. If MODE=3, use values saved (in common block) from previous call of your module.
4. COMPUTE Y using inputs a_0 and a_1
5. RETURN

Strategy 2, Derivative, $y = a_0 + a_1 x + a_2 dx/dt$

(Same steps as for Strategy 1)

Strategy 3, Integral, $y = a_0 + a_1 x + a_2 dx/dt + a_3 \int_0^t x dt$

(The same steps should be used as for 1 and 2, but in addition provision should be made for initialization and computation of the integral as a cumulative sum.)

Appendix B. Results of Stress Test Runs

STRESS TEST RUN SSQ1

TOTAL CASES EXECUTED 1,000,000

TEST CASE GENERATION

Pitch Command Waveform SQ(20,40,20,40,60,-30)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(0.0,0.0)	Distribution	N(0.0,0.0)
Truncation Points	[0.0,0.0]	Truncation Points	[0.0,0.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .021$ $a_3 = .042$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-6.2,11.9]
δ_E^N	No	[-6.8,9.3]
$\hat{\theta}_A^F$	No	[-30.8,49.2]
$\hat{\theta}_A^N$	No	[-31.7,34.3]
$\lvert E_\theta^F \rvert$	No	63.0
$\lvert E_\theta^N \rvert$	No	121.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.6
$\lvert \Delta\delta_i^{NM} \rvert$	No	7.0
$\lvert \Delta\delta_i^{NF} \rvert$	No	3.1
$\lvert \Delta\delta_E^{NF} \rvert$	No	5.0

STRESS TEST RUN SSQ2

TOTAL CASES EXECUTED 1,000,000

TEST CASE GENERATION

Pitch Command Waveform SQ(20,40,20,40,60,-30)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(3.0,1.0)	Distribution	N(0.0,0.0)
Truncation Points	[2.0,4.0]	Truncation Points	[0.0,0.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .021$ $a_3 = .042$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-6.6,11.8]
δ_E^N	No	[-7.4,9.7]
$\hat{\theta}_A^F$	No	[-31.5,51.0]
$\hat{\theta}_A^N$	No	[-31.7,37.2]
$\lvert E_\theta^F \rvert$	No	67.0
$\lvert E_\theta^N \rvert$	No	138.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.6
$\lvert \Delta\delta_i^{NM} \rvert$	No	8.0
$\lvert \Delta\delta_i^{NF} \rvert$	No	3.2
$\lvert \Delta\delta_E^{NF} \rvert$	No	5.1

STRESS TEST RUN SSQ3

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SQ(20,40,20,40,60,-30)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(0.0,0.0)	Distribution	N(3.0,1.0)
Truncation Points	[0.0,0.0]	Truncation Points	[2.0,4.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .021$ $A_3 = .042$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-7.4,11.2]
δ_E^N	No	[-8.4,7.9]
$\hat{\theta}_A^F$	No	[-34.7,47.4]
$\hat{\theta}_A^N$	No	[-35.0,34.8]
$\lvert E_\theta^F \rvert$	No	57.0
$\lvert E_\theta^N \rvert$	No	113.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.6
$\lvert \Delta\delta_i^{NM} \rvert$	No	8.0
$\lvert \Delta\delta_i^{NF} \rvert$	No	3.1
$\lvert \Delta\delta_E^{NF} \rvert$	No	4.3

STRESS TEST RUN SSQ4

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SQ(20,40,20,40,60,-30)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(3.0,1.0)	Distribution	N(3.0,1.0)
Truncation Points	[2.0,4.0]	Truncation Points	[2.0,4.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .021$ $a_3 = .042$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-8.2,11.4]
δ_E^N	No	[-9.4,9.5]
$\hat{\theta}_A^F$	No	[-34.4,49.3]
$\hat{\theta}_A^N$	No	[-34.7,40.4]
$\lvert E_\theta^F \rvert$	No	61.0
$\lvert E_\theta^N \rvert$	No	129.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.6
$\lvert \Delta\delta_i^{NM} \rvert$	No	7.5
$\lvert \Delta\delta_i^{NF} \rvert$	No	3.1
$\lvert \Delta\delta_E^{NF} \rvert$	No	5.3

STRESS TEST RUN SSQ5

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SQ(20,40,20,40,60,-30)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(7.5,2.5)	Distribution	N(0.0,0.0)
Truncation Points	[5.0,10.0]	Truncation Points	[0.0,0.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .021$ $a_3 = .042$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-8.5,12.3]
δ_E^N	No	[-10.9,13.4]
$\hat{\theta}_A^F$	No	[-32.3,53.7]
$\hat{\theta}_A^N$	No	[-34.0,48.8]
$\lvert E_\theta^F \rvert$	No	77.0
$\lvert E_\theta^N \rvert$	No	163.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.7
$\lvert \Delta\delta_i^{NM} \rvert$	No	9.5
$\lvert \Delta\delta_i^{NF} \rvert$	No	3.3
$\lvert \Delta\delta_E^{NF} \rvert$	No	6.6

STRESS TEST RUN SSQ6

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SQ(20,40,20,40,60,-30)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(0.0,0.0)	Distribution	N(7.5,2.5)
Truncation Points	[0.0,0.0]	Truncation Points	N[5.0,10.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .021$ $a_3 = .042$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-11.0,10.5]
δ_E^N	No	[-14.0,11.2]
$\hat{\theta}_A^F$	No	[-41.9,44.8]
$\hat{\theta}_A^N$	No	[-45.1,44.1]
$\lvert E_\theta^F \rvert$	No	51.0
$\lvert E_\theta^N \rvert$	No	102.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.5
$\lvert \Delta\delta_i^{NM} \rvert$	No	9.5
$\lvert \Delta\delta_i^{NF} \rvert$	No	4.8
$\lvert \Delta\delta_E^{NF} \rvert$	No	10.5

STRESS TEST RUN SSQ7

TOTAL CASES EXECUTED 1,000,000

TEST CASE GENERATION

Pitch Command Waveform SQ(20,40,20,40,60,-30)

Pitch Command Noise		Pitch Response Noise	
Distribution	N(7.5,2.5)	Distribution	N(7.5,2.5)
Truncation Points	[5.0,10.0]	Truncation Points	[5.0,10.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .021$ $a_3 = .042$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-11.9,12.0]
δ_E^N	No	[-15.4,16.1]
$\hat{\theta}_A^F$	No	[-39.1,54.6]
$\hat{\theta}_A^N$	No	[-39.9,55.7]
$\lvert E_\theta^F \rvert$	No	70.0
$\lvert E_\theta^N \rvert$	No	140.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.6
$\lvert \Delta\delta_i^{NM} \rvert$	No	8.0
$\lvert \Delta\delta_i^{NF} \rvert$	No	5.3
$\lvert \Delta\delta_E^{NF} \rvert$	No	11.9

STRESS TEST RUN SSN1

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SINE (30,.16,0,0)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(0.0,0.0)	Distribution	N(0.0,0.0)
Truncation Points	[0.0,0.0]	Truncation Points	[0.0,0.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .042$ $a_3 = .084$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$
$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-4.7,8.8]
δ_E^N	No	[-6.9,8.1]
$\hat{\theta}_A^F$	No	[-29.5,36.8]
$\hat{\theta}_A^N$	No	[-32.1,34.2]
$\lvert E_\theta^F \rvert$	No	47.0
$\lvert E_\theta^N \rvert$	No	54.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.7
$\lvert \Delta\delta_i^{NM} \rvert$	No	1.1
$\lvert \Delta\delta_i^{NF} \rvert$	No	1.3
$\lvert \Delta\delta_E^{NF} \rvert$	No	2.7

STRESS TEST RUN SSN2

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SINE (30,.16,0)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(3.0,1.0)	Distribution	N(0.0,0.0)
Truncation Points	[2.0,4.0]	Truncation Points	[0.0,0.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .042$ $a_3 = .084$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-5.8,8.7]
δ_E^N	No	[-8.8,9.8]
$\hat{\theta}_A^F$	No	[-30.0,38.6]
$\hat{\theta}_A^N$	No	[-33.6,39.6]
$\lvert E_\theta^F \rvert$	No	54.0
$\lvert E_\theta^N \rvert$	No	64.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.7
$\lvert \Delta\delta_i^{NM} \rvert$	No	1.4
$\lvert \Delta\delta_i^{NF} \rvert$	No	1.9
$\lvert \Delta\delta_E^{NF} \rvert$	No	4.5

STRESS TEST RUN SSN3

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SINE (30,.16,0,0)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(0.0,0.0)	Distribution	N(3.0,1.0)
Truncation Points	[0.0,0.0]	Truncation Points	[2.0,4.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .042$ $a_3 = .084$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-6.8,8.2]
δ_E^N	No	[-9.8,8.3]
$\hat{\theta}_A^F$	No	[-33.5,35.0]
$\hat{\theta}_A^N$	No	[-36.4,36.6]
$\lvert E_\theta^F \rvert$	No	44.0
$\lvert E_\theta^N \rvert$	No	51.0

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.7
$\lvert \Delta\delta_i^{NM} \rvert$	No	1.3
$\lvert \Delta\delta_i^{NF} \rvert$	No	2.5
$\lvert \Delta\delta_E^{NF} \rvert$	No	5.7

STRESS TEST RUN SSN4

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SINE (30,.16,0,0)

Pitch Command Noise		Pitch Response Noise	
Distribution	N(3.0,1.0)	Distribution	N(3.0,1.0)
Truncation Points	[2.0,4.0]	Truncation Points	[2.0,4.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .042$ $a_3 = .084$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-7.1,8.1]
δ_E^N	No	[-11.3,11.4]
$\hat{\theta}_A^F$	No	[-32.3,38.4]
$\hat{\theta}_A^N$	No	[-37.4,43.0]
$\lvert E_\theta^F \rvert$	No	52
$\lvert E_\theta^N \rvert$	No	61

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	0.8
$\lvert \Delta\delta_i^{NM} \rvert$	No	1.5
$\lvert \Delta\delta_i^{NF} \rvert$	No	3.1
$\lvert \Delta\delta_E^{NF} \rvert$	No	7.0

STRESS TEST RUN SSN5

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SINE (30,.16,0,0)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(7.5,2.5)	Distribution	N(0.0,0.0)
Truncation Points	[5.0,10.0]	Truncation Points	[0.0,0.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .042$ $a_3 = .084$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-7.7,10.1]
δ_E^N	No	[-14.6,17.7]
$\hat{\theta}_A^F$	No	[-31.2,45.0]
$\hat{\theta}_A^N$	No	[-39.2,56.2]
$\lvert E_\theta^F \rvert$	No	68
$\lvert E_\theta^N \rvert$	No	84

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	1.3
$\lvert \Delta\delta_i^{NM} \rvert$	No	2.2
$\lvert \Delta\delta_i^{NF} \rvert$	No	4.1
$\lvert \Delta\delta_E^{NF} \rvert$	No	9.9

STRESS TEST RUN SSN6

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SINE (30,.16,0,0)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(0.0,0.0)	Distribution	N(7.5,2.5)
Truncation Points	[0.0,0.0]	Truncation Points	[5.0,10.0]
Sccd	1234567808	Sccd	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .042$ $a_3 = .084$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Rcquirement	Enabled	Extreme Values
δ_E^F	No	[-10.2,8.1]
δ_E^N	No	[-18.0,14.4]
$\hat{\theta}_A^F$	No	[-40.1,40.7]
$\hat{\theta}_A^N$	No	[-49.5,48.8]
$\lvert E_\theta^F \rvert$	No	18.5
$\lvert E_\theta^N \rvert$	No	29.5

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	1.3
$\lvert \Delta\delta_i^{NM} \rvert$	No	2.1
$\lvert \Delta\delta_i^{NF} \rvert$	No	6.0
$\lvert \Delta\delta_E^{NF} \rvert$	No	14.2

STRESS TEST RUN SSN7

TOTAL CASES EXECUTED	1,000,000

TEST CASE GENERATION

Pitch Command Waveform	SINE (30,.16,0,0)

Pitch Command Noise		**Pitch Response Noise**	
Distribution	N(7.5,2.5)	Distribution	N(7.5,2.5)
Truncation Points	[5.0,10.0]	Truncation Points	[5.0,10.0]
Seed	1234567808	Seed	9876542464

CONTROL LAW PARAMETERS

$a_0 = .000$ $a_1 = .167$ $a_2 = .042$ $a_3 = .084$

AIRCRAFT RESPONSE MODEL PARAMETERS

$\alpha_1 = 17.5$ $\alpha_2 = 39.4$ $\alpha_3 = 17.0$

$\beta_1 = 1.02$ $\beta_2 = 7.01$ $\beta_3 = 18.70$ $\beta_4 = 8.60$ $\beta_5 = 2.84$

SYSTEM PERFORMANCE REQUIREMENTS

Requirement	Enabled	Extreme Values
δ_E^F	No	[-11.0,11.2]
δ_E^N	No	[-22.3,21.9]
$\hat{\theta}_A^F$	No	[-37.1,52.2]
$\hat{\theta}_A^N$	No	[-51.7,66.1]
$\lvert E_\theta^F \rvert$	No	63
$\lvert E_\theta^N \rvert$	No	77

INEQUALITY CONSTRAINTS

Requirement	Enabled	Extreme Values
$\lvert \Delta\delta_{ij}^{NN} \rvert$	Yes	2.0
$\lvert \Delta\delta_i^{NM} \rvert$	No	2.5
$\lvert \Delta\delta_i^{NF} \rvert$	No	7.7
$\lvert \Delta\delta_E^{NF} \rvert$	No	17.4

Standard Bibliographic Page

1. Report No. NASA CR- 178058	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle An Empirical Study of Flight Control Software Reliability	5. Report Date March 1986 6. Performing Organization Code 412U-2094-22
7. Author(s) Janet R. Dunham and John L. Pierce	8. Performing Organization Report No.
9. Performing Organization Name and Address Research Triangle Institute P.O. Box 12194 Research Triangle Park, N.C. 27709	10. Work Unit No. 11. Contract or Grant No. NAS1-17964
12. Sponsoring Agency Name and Address National Aeronautics & Space Administration Langley Research Center Hampton, VA 23665	13. Type of Report and Period Covered Contractor Report 14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitor: G. Earle Migneault
Final Report

16. Abstract

The report documents the results of a small-scale laboratory experiment in flight control software reliability. The experiment tests a small sample of implementations of a pitch axis control law for a PA28 aircraft with over 14 million pitch commands having varying levels of additive input and feedback noise. The testing which uses the method of n-version programming for error detection surfaced four software faults in one implementation of the control law. This small number of detected faults precluded the conduct of the error burst analyses and may be due to the overly simple example chosen for the control law software and the programmer's ability to exploit this simplicity in validating it.

The pitch axis problem was chosen to provide data for use in constructing a model for predicting the reliability of software in systems with feedback. The experiment is one in a series of experiments being pursued by the Systems Validation Branch of NASA - Langley Research Center to find a means of credibly performing reliability evaluations of flight control software.

17. Key Words (Suggested by Authors(s)) Software Reliability N-Version Testing Life-Critical Software Software Engineering Experiments	18. Distribution Statement Unlimited

19. Security Classif.(of this report) Unclassified	20. Security Classif.(of this page) Unclassified	21. No. of Pages 40	22. Price

For sale by the National Technical Information Service, Springfield, Virginia 22161