Design of Optimal Motion for Flight Simulators

by

Jehuda Ish-Shalom

B.S.E.E., Technion, Israel Institute of Technology
(1974)
M.S.E.E., Technion, Israel Institute of Technology
(1978)

Submitted in Partial Fulfillment
of the Requirements for the
Degree of

Doctor of Philosophy

at the
Massachusetts Institute of Technology
December, 1982

Signature of Author______________________________
Interdepartmental Program in Bio-Medical Engineering December 21, 1982

Certified by______________________________
Laurence R. Young Committee Chairman

Certified by______________________________
William M. Siebert Committee Member

Certified by______________________________
Bernard C. Levy Committee Member

Certified by______________________________
Robert V. Kenyon Committee Member

Accepted by______________________________
HAROLD Y. WACHMAN, CHAIRMAN
DEPARTMENTAL GRADUATE COMMITTEE

To my parents Moshe and Dina
Who brought me up to reach this day

To my wife Niti and children Amir and Alon
Who suffered the burden of this thesis

Design of Optimal Motion for Flight Simulators

By

Jehuda Ish-Shalom

Submitted to the Interdepartmental Program in Bio-Medical Engineering on December 21, 1982 in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Bio-Medical Engineering.

Abstract

An abstract simulator design problem is formulated as follows: Given a dynamic system, S^a, called the *actual* system and another dynamic system, S^s, called a *simulator* for S^a, and given a function which drives the system S^a, the problem is to find an operator, properly constrained, which will generate the input to S^s on the basis of the input to S^a, such that the discrepancy between the outputs of S^a and S^s will be as small as possible. This abstract simulator design problem is formulated as an optimal control problem and in the linear-quadratic case this problem is decomposed into two separately solvable subproblems: (i) deterministic and (ii) stochastic. Both subproblems are solved; the stochastic one for the Gaussian case only. Examination of the properties of the solution reveals a parallel decomposition theorem and the dependence of the simulator design on the given parameters. These and other properties of the solution enable the extension of the solution to include output nonlinearities in S^s and S^a and for a time varying system representation for the expected input to S^a. These properties make it possible to develop a methodology for the design of optimal simulators.

Next, the solution of the abstract simulator problem is applied to the design of motion generation for moving base flight simulators. The optimization criterion selected is a norm of the difference between the physiological outputs of the vestibular organs of a pilot in an imaginary reference *airplane* and those of a pilot in the *simulator*. Vestibular models based on physiological and psychophysical experiments were used. As a consequence, a new design methodology is suggested for the design of the motion of moving base flight simulators.

As a demonstration of this methodology several design examples were solved and simulated. The results conform the set of empirically found design rules used by experienced engineers to determine the filter parameters of flight simulator motion generation systems. In addition several designs were implemented and were tested by twenty pilots. These designs were implemented for the pitch and surge axes on a Link GAT-1 general aviation flight simulator. These tests also suggest a possible reason for many general aviation accidents that occur due to a stall during a landing approach. Last, a generalization to a nonlinear motion generation system was implemented, which can be easily applied to the full six-degrees-of-freedom case.

The design method that we have obtained can also be used for model following or robotic motion design.

Thesis Supervisors:

Laurence R. Young	Professor of Aeronautics and Astronautics
William M. Siebert	Professor of Electrical Engineering
Bernard C. Levy	Assistant Professor of Electrical Engineering
Robert V. Kenyon	Assistant Professor of Aeronautics and Astronautics

Acknowledgments

I wish to thank Prof. Laurence R. Young for introducing me to research connected with the vestibular system and for providing the environment and support for this work; Prof. Rafhael Sivan for suggesting the topic and for his help during varying stages of this work; Prof. Bernard C. Levy for being a friend and for guidance and discussion throughout this work; Prof. William M. Siebert for advice, guidance of the general direction of this thesis and for his cordial support and encouragement at anytime and in particular—at difficult times. Thanks are due also to Prof. Robert V. Kenyon for his interest in this work and for comments on chapter VII and for supporting the construction of the GAT-1 OWS by serving as faculty advisor to the 16.62 students.

Thanks to Jen-Kuang Huang who helped in preparing the two degree-of-freedom example given in chapter V and to Paul A. Wetzel for his help with the statistical analysis in chapter VII. Comments on various chapters by Anthony P. Arrott (VII, VIII), Russ Taylor (II, IV), Prof. Shankar Sastry (IV), Ching-Meng Chew (III) and Sew Wah Tay (VII) are gratefully acknowledged.

For introducing me to various aspects related to this work I would like to mention with thanks: Chuck M. Oman—introduction to vestibular function and helpful comments as a subject in the simulator experiments; Prof. Walter M. Hollister for giving me the feeling of real flight in his 16.64 course and being a subject in the experiments; Russell Parrish and Roland Bowles from NASA Langley Research Center, W. Harrington from Wright Patterson AFB, Dick Bray, A. Daivd Jones from NASA AMES Research Center for demonstrating and discussing large research simulators; my two brothers Carmel and Yaron who shared their pilot experience with me.

The guest welcoming atmosphere of the MIT Artificial Intelligence Lab. made it possible to substantially enhance my work through the use of MACSYMA. I would also like to thank George J. Carrette (GJC) for extending MACSYMA for my application, Jeffrey P. Golden (JPG) and Ellen Golden for their help with the MACSYMA system and to Oded A. Feingold (OAF).

The assistance of the 16.62 students George C. Ranney and Frahad Zarinechi, and that of Allan R. Shaw and Robert L. Renshaw in the experimental work was much appreciated.

And last I wish to express gratitude to Sherry Modestino for her help in the Lab. and in preparing this manuscript and to all the subjects in the experiments and in particular to Ian McArthur, Jerry Weiss and John Hansman for their long hours in the "cockpit"; and to my wife Niti for her infinite patience and hard work on the nice figures and in typing parts of this thesis.

This work was supported by NASA Ames Research Center Grant NSG 2230 and NAG 2-12, and in part by the United States Energy Research and Development Administration under Contract E(11-1)-3070 and by NASA Grant 1323.

Biography

Jehuda Ish-Shalom received the B.Sc. degree in 1974 and the M.Sc. in 1978, both in Electrical Engineering from the Technion, Israel Institute of Technology, and the Ph.D. in Bio-Medical Engineering from MIT in 1982.

In 1970, he developed an ultrasonic mobility aid for the blind which won first prize in the 1971 contest "Models and Essays in Natural Science and Mathematics," Weizman Institute, Israel. During his undergraduate studies he was awarded a special grant while working at the Medical Electronics Lab. in the E.E. Dep. in Technion I.I.T.. During 1973 he worked as a test engineer at "Elbit Computers," Haifa, Israel. His M.Sc. thesis was on the "Measurement of Eye Movement with a Ferromagnetic Contact Ring." His Ph.D. work was conducted at the MIT Man-Vehicle Lab. on the "Design of Optimal Motion for Flight Simulators" using vestibular system models. Dr. Ish-Shalom is currently working as a Research Staff Member in the Robotics Lab. of IBM at IBM Thomas J. Watson Research Center, Yorktown Heights, New-York.

Dr. Ish-Shalom is a member of the Sigma Xi Society. His major research interests are in robotics, medical instrumentation, sensory systems and VLSI circuit design.

Publications and Reports:

R. Sivan, J. Ish-Shalom and J.-K. Huang, "An Optimal Control Approach to the Design of Moving Flight Simulators," Transaction of IEEE on Systems, Man, and Cybernetics, November, 1982. Also included in the IEEE proceedings of the international conference on cybernetics and society, Oct. 1980.

Y.Y. Zeevi and J. Ish-Shalom "Measurement of Eye Movement with a Ferromagnetic Contact Ring," Transactions of IEEE on Bio-Medical Engineering Vol. BME-29, pp. 511–522 July, 1982. Also included in the proceedings of the conference of the Israeli section of IEEE, Tel-Aviv Israel, Sep., 1977.

J. Ish-Shalom, "Head Rotation Monitoring Device," Man Vehicle Laboratory Report, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, June. 1979.

J. Ish-Shalom, "An Ultrasonic Probe—An Aid to the Blind." Submitted the contest "Models and Essays in Natural Sciences and Mathematics," WEIZMAN INSTITUTE, Israel, 1971.

Design of Optimal Motion for Flight Simulators

By

Jehuda Ish-Shalom

Summary

An abstract simulator design problem is formulated as follows: given a dynamic system, S^a, called the *actual* system and another dynamic system, S^s, called a *simulator* for S^a, and given a function which drives the system S^a, the problem is to find an operator, properly constrained, which will generate the input to S^s on the basis of the input to S^a, such that the discrepancy between the outputs of S^a and S^s will be as small as possible. This abstract simulator design problem is formulated as an optimal control problem and in the linear-quadratic case this problem is decomposed into two separately solvable subproblems: (i) deterministic and (ii) stochastic. Both subproblems are solved; the stochastic one for the Gaussian case only. An examination of the properties of the solution puts in evidence a parallel decomposition theorem and provides an interpretation of the dependence of the simulator design on the given parameters. These and other properties simplify the solution so as to enable the extension of the solution to include several nonlinear effects.

The study of the nonlinear effects includes three topics. The first topic is an extension of the deterministic-stochastic decomposition to include nonlinear dynamic system equations using a quadratic cost function. This decomposition shows a general method of how to separate and then combine the "open-loop" (deterministic) and the "closed-loop" (stochastic) solutions for the abstract simulator design problem. This is also true for the general control problem appearing in robot control design.

The second topic is the development of a *Pseudo Linear Quadratic* controller (PLQ) for linear and nonlinear dynamic systems. The PLQ controller is derived from the standard *Linear Quadratic* (LQ) optimal control solution by solving for

a quasi-quadratic cost and a quasi-linear system for each value of the state. This results in a feedback with a leading linear term, i.e. using feedback gains that are functions of the system state rather then constants. Though PLQ is not a solution to any known formulated optimization it is an extension of the standard LQ control. Furthermore, in the cases tested it has properties that match those of known optimal nonlinear controllers derived for linear dynamic systems using a nonquadratic cost. On the other hand, PLQ is easier to compute and easier to implement, due to its "linear" form. Many of the PLQ properties still need to be developed including conditions for global stability for the multi dimension dynamic system case. It is expected that the resulting PLQ controller would show similar robustness properties as the LQ controller.

The third nonlinear effect discussed is a *sign sensitive cost* formulation and solution. The cost function is put into a form that includes a correlation function term that is evaluated between the outputs of the systems S^a and S^s. It is shown that, any antisymmetric compressive memoryless output function, cascaded to the linear dynamics of both S^a and S^s, would lead to a cost function that should include a *sign sensitive* term. This problem is put into a LQ form which no longer has a positive definite cost. It is shown that a unique solution exists for the abstract simulator design problem. Finally, putting all these elements together, enables one to develop a methodology for the design of abstract optimal simulators.

Next, the solution and properties of the abstract simulator problem are applied to the design of motion generation for moving-base flight simulators. The optimization criterion selected is a quadratic norm of the difference between the physiological outputs of the vestibular organs of a pilot in an imaginary reference *airplane* and those of a pilot in the *simulator*. Vestibular models based on physiological and psychophysical experiments were used, including consideration of vestibular sensor saturation and the multiplicative nature of the physiological noise in the nervous system, modeled by an antisymmetric compressive memoryless output nonlinearity. The LQ abstract simulator properties imply a 2-2-1-1 physical axis decoupling theorem for the feedback gains, i.e. Pitch-Surge, Roll-Sway, Yaw, Heave axis group decoupling. The 2 axis coupling is due to gravity. This 2-2-1-1 rule is well known to designers of simulator motion systems. What is usually overlooked is that the feed-forward gains do not decouple the same way due to the effect of the airplane dynamics coupling. When axis transformations are included in the motion system implementation, coupling between all six physical axes is obtained—a property not existing in current designs. An example of this effect is the proper motion generation for the falsely called "Coriolis motion sensation" which usually requires a simulator with full 360^o rotation capabilities. Furthermore, a method for use of head rotation measurement is developed which further improves the simulator motion sensation. As a consequence, a new design methodology is suggested for the design of the motion of moving-base flight simulators.

As a demonstration of this methodology, several design examples were solved and simulated. The results conform to the set of empirically found design rules used by experienced engineers: to determine the structure (2-2-1-1 theorem), the initial setting of the pole locations, the expected lower motion fidelity as the poles' frequency increases, cross coupling gain between the linear and rotation motion input (called g tilt) used in flight simulator motion generation systems.

Twenty pilots tested several of these designs which were implemented for the pitch and surge axes on a Link GAT-1 General Aviation flight simulator Trainer. These tests confirm the suggested design method, including equal weighting for the normalized vestibular linear and rotation components. Furthermore, these tests show the effects of motion on the pilot's control for a sudden unexpected flaps-down transition during level flight. This experiment also suggests a possible reason for many general aviation accidents that occur due to a stall during a landing approach due to its similarity to the above experiment. It was found that even very experienced pilots with more then ten thousand flight hours can easily be confused initially by the motion and make a wrong elevator control. However, they report making the right control. Lastly, PLQ was used to build a nonlinear-motion-generation system for the Link GAT-1 simulator. This nonlinear design can be implemented easily for the full six-degrees-of-freedom case.

The examples and pilot tests presented in this thesis are preliminary investigations into the feasibility of the optimal simulator design approach. The results so far are promising. The causal, linear, time-invariant "optimal" motion system derived here has parameters of the same order of magnitude as the conventional motion systems in use today. However, unlike these systems, the "optimal" motion system can be "tuned" by a non-expert using this computer design method to satisfy a variety of additional conditions such as: different travel lengths of the simulator, different flight trajectories, and different emphasis on motion cues. Furthermore, it makes use of expected future airplane motions, accounts better for hard limits by use of PLQ and takes into account axis transformations and head movements. It is simpler to implement and as a bonus gives the control system design for the motion-base itself.

It is recommended that this design method be transformed into an *optimal motion system design compiler* that is capable of transforming a simple minded, non-expert specification of the required motion system into a flight simulator motion-generation system. The design method that we have obtained can also be used for model-following or robot motion design.

Table of Contents

Abstract 5

Acknowledgments 7

Biography 9

Summary 11

Contents 15

I Introduction 19

I.1. Moving-Base Flight Simulator **19**
I.2. Survey of Moving Base Flight Simulator Types **20**
I.2.1. Cascaded systems 20
I.2.2. Hexapod System 21
I.2.3. Articulated Beam System 22
I.2.4. Centrifuge Motion System 22
I.2.6. In-Flight Simulators, T-33 and TIFS 22
I.3. Survey of Present Washout Filters **32**
I.4. Basic Approach **33**
I.5. What to Read in order to Design an Optimal Washout System **36**

References 39

II Formulation of the Motion Problem 41

II.1. The Flight Simulator **42**
II.2. Modeling of the Airplane Anticipated Motions **44**
II.3. Pilot Block Diagram **46**
II.3.1. Conceptual Building Blocks 47
II.3.2. Pilot Information Flow and its Uncertainty 50
II.4. Comparison of Airplane Flight to Simulator Flight **52**
II.4.1. Comparison of Cockpit Motions 53
II.4.2. Comparison of Sensory Measurements 54
II.4.3. Comparison of Orientation Estimate 54
II.4.4. Comparison of Control Effort 58
II.4.5. Comparison of Pilots Performance 59
II.4.6. Experimental Use of Control Effort and Performance Comparison . 59
II.5. Cost Function **59**
II.6. Flight Simulator Motion Design Problem **61**
II.7. Can the Vestibular Organ Represent All the Inertial Motion Sensors? **67**
II.8. Vestibular Organ Modeling **69**
II.8.1. Vestibular Organ Description 69
II.8.2. Assumptions in the Vestibular Model Used 70
II.9. Axis Systems **71**

II.9.1. Axis Systems Involved . . . 72
II.9.2. Notation . . . 74
II.9.3. Handling of axis systems . . . 75
II.9.4. Including head movements . . . 78
II.9.5. Consideration of the Motion Limitation Axis System . . . 79
II.10. Abstract Optimal Simulator Problem Statement . . . **81**

References . . . 83
Appendix II.A. Equations of Motion of the Link GAT-1 Simulator . . . 87

III Solution of the Linear Quadratic Case . . . 95

III.1. Statement of the Linear Quadratic Case . . . **97**
III.2. Separation Into Deterministic And Stochastic Problems . . . **98**
III.3. Deterministic Problem Solution . . . **100**
III.4. Stochastic Problem Solution . . . **103**
III.4.1. Solution of the L.Q.G. Optimal Simulator . . . 104
III.4.2. Derivation of the L.Q.G. Optimal Simulator Filter . . . 106
III.5. Properties of the time-invariant L.Q.G. Solution . . . **108**
III.5.1. Algebraic Structure of the Algebraic Riccati Equation . . . 109
III.5.2. Dimension Reduction . . . 115
III.5.2.1. Riccati equation (P^s =?) . . . 115
III.5.2.2. Linear equation (P^n =?) . . . 118
III.5.3. Simulation Filter Simplification in the Symmetric Case . . . 123
III.6. Time-Varying Stochastic Problem Solution . . . **125**

References . . . 128
Appendix III.A. Derivation of Block Equations . . . 129
Appendix III.P. MACSYMA Program . . . 141

IV Nonlinearities . . . 143

IV.1. Deterministic-Stochastic Problem Decomposition . . . **146**
IV.2. Pseudo Linear Quadratic Control . . . **153**
IV.2.1. Result for the linear case . . . 154
IV.2.2. Stability proof for the linear case . . . 156
IV.2.3. Example of PLQ control for a linear plant . . . 161
IV.2.4. PLQ Control of Nonlinear Plants . . . 162
IV.2.5. Example of PLQ Control for a Nonlinear Scalar Plant . . . 164
IV.3. Alternative Performance Criteria . . . **167**
IV.3.1. Introduction . . . 168
IV.3.1.1. Sensor Saturation . . . 168
IV.3.1.2. Firing Rate Statistics . . . 170
IV.3.1.3. Correlation Cost . . . 175
IV.3.2. Problem Formulation . . . 175
IV.3.3. Solution . . . 177
IV.3.3.1. Deterministic Solution . . . 177

IV.3.3.2. Stochastic Solution 178
IV.3.4. Special Properties of the Solution 180

References 183
Appendix IV.A. Derivation of Best Linear Approximation 185
Appendix IV.B. Derivation of Block Equations For Sign Sensitive Cost . . 187
Appendix IV.C. Derivation of Equations With Cost Singularity Removed 195
Appendix IV.P-A. MACSYMA Program A 203
Appendix IV.P-B. MACSYMA Program B 205
Appendix IV.P-C. MACSYMA Program C 207

V Design Examples 209

V.1. A One-Degree-of-Freedom Example **209**
V.1.1. Derivation of the Washout Filter 209
V.1.2. Discussion of the Results 214
V.2. A Two-Degree-of-Freedom Example **215**
V.2.1. The Model for the Vestibular System 215
V.2.2. Derivation of the Optimal Washout Transfer Matrix 219
V.2.3. Simulations 222
V.3. Conclusions **222**

References 230

VI Washout System Implementation 231

VI.1. Optimal Washout System Form of the Stochastic Solution **233**
VI.2. Merging the Deterministic and Stochastic Solutions **235**
VI.3. Axis Transformation **237**
VI.4. Head Rotations **242**
VI.5. Time-Varying Optimal Washout System **243**
VI.6. Computation Delay **244**
VI.7. Sampling **244**
VI.8. Implementation With a Sign Sensitive Cost **245**
VI.9. Pseudo Linear Quadratic System Implementation **245**

References 248

VII Experimental Evaluation 249

VII.1. Introduction **250**
VII.1.1. Pilot's Performance and Control 251
VII.1.2. Pilot Opinion 251
VII.1.3. Detect Washout Changes 252
VII.1.4. Detect Airplane Changes 254
VII.2. Materials and Methods **255**
VII.2.1. The Link GAT-1 Flight Simulator 255
VII.2.2. Pitch Optimal Washout Implementation 257
VII.2.3. Nonlinear washout and interpolator 264

VII.2.4. Experimental Design . . . 268
VII.2.4.1. Experiment 1—Blind Test . . . 269
VII.2.4.2. Experiment 2—Washout Detection . . . 269
VII.2.4.3. Experiment 3—Flaps-down Detection . . . 271
VII.2.4.4. Experiment 4—Take-off . . . 274
VII.3. Results . . . **276**
VII.3.1. Experiment 1—Blind test . . . 276
VII.3.2. Experiment 2—Washout Detection . . . 277
VII.3.3. Experiment 3—Flaps-down Detection . . . 281
VII.3.3.1. Initial Response Results . . . 281
VII.3.3.2. After Training Results . . . 285
VII.3.4. Experiment 4—Take-off . . . 302
VII.4. Discussion . . . **311**

References . . . 316
Appendix VII.A. Suggested Experiments on VMS . . . 317
Appendix VII.B. Washout Design Programs . . . 329
Appendix VII.C. GAT-1 OWS Operating Instructions . . . 383
Appendix VII.D. Pitch Motion-Base Model . . . 387
Appendix VII.E. OWS Circuit Tests . . . 389
Appendix VII.F. GAT-1 Modification for Use of the Pitch OWS . . . 393
Appendix VII.G. Design Parameters and simulations of OWS #0 and #2 . 395
Appendix VII.H. Design Parameters of the OWSs Used in the Experiment 425
Appendix VII.I. Flaps-Down Detection Experiment Results . . . 445
Appendix VII.J. Experience of Pilots Used . . . 457
Appendix VII.K. Experiments to Demonstrate the OWS, used in 16.36 . . . 459

VIII Conclusions and Further Research . . . 483

VIII.1. Elements in the Design of an Optimal Washout System . . . **484**
VIII.2. Conclusions From the Experiments . . . **485**
VIII.3. Suggestions for Further Research . . . **487**

Chapter I

Introduction

In this chapter we describe how current flight simulators provide motion: what the motion base principle of operation is and how current methods provide control to it such that it does not go beyond its boundaries. A very good annotated bibliography on motion in flight simulators was written by Puig, Harris, and Ricard [Puig78], it includes a review of equipment, control methods, effect of motion and evaluation of motion in flight simulators and references 682 documents. Much of the material in this chapter is a short summary of this reference.

One should remember that physically moving a pilot is not the only way to provide a pilot with motion sensation through his inertial motion sensors. As many people know drinking alcohol (or heavy water) can give a rotation sensation. Unfortunately, so far, there is no practical method to use this or other effects to give a pilot motion sensation through his inertial motion sensors. Therefore in this thesis we address the question of how to provide "best" pilot motion using the limited motion capability of a ground based motion flight simulator.

1. Moving-Base Flight Simulators

A ground based motion flight simulator is an airplane cockpit installed on a motion system which has a certain motion capability. The purpose of the motion in the simulator is to provide the pilot who "flies" the simulator with motion cues which will aid in development of pilot control techniques and in assessment of simulated aircraft systems. The simulator, being a ground installation is, of course, constrained to stay within some bounds. One has to generate the simulator motion, on the basis of the actual airplane motion, so as to (i) give relevant motion cues to the pilot, and (ii) stay within the simulator constraints. The system which generates the motion commands for the simulator typically consists of a *washout filter* as well as limiting and transformation functions. In this thesis we define a new term, *washout system*, which includes all these elements:

> WASHOUT SYSTEM
>
> That part of the simulator display unit which computes the simulator motion-base commands on the basis of the computed airplane motions, so as not to exceed the motion-base constraints, yet retain the simulation flight "realism" as best as possible.

2. Survey of Moving Base Flight Simulator Types

Several motion-base types are described below. These include variable stability airplanes which are used when ground base installation does not suffice. In Figure 1 we show the naming convention of the six degrees-of-freedom of a flight simulator:

1. Surge, fore-aft linear motion, x axis.
2. Sway, lateral linear motion, y axis.
3. Heave, vertical linear motion, z axis.
4. Roll, angular motion, ϕ rotation.
5. Pitch, angular motion, θ rotation.
6. Yaw, angular motion, ψ rotation.

2.1. Cascaded systems

Many flight simulators use a cascaded motion system, that is, a cascade of six motion elements, one for each motion axis. The rotation is provided by a set of gimbals, one gimbal for each rotation axes. Thus, three nested gimbals are needed in order to have all three rotation axis. The rotation angle of each gimbal is given by its corresponding Euler angle. Each gimbal is driven by a separate motor, and the limitations of such a system are given in terms of the individual limits of each axis; the maximum Euler angles, Euler angular velocities and Euler angular accelerations.

The simulator linear motion is provided by a cascade of linear tracks, one for each linear motion axis used, and the motion limitations are similar to those imposed for the gimbal system. Usually these limitations also include a parabolic-limiter, which limits the motions to less than the maximum track length, such that the linear motion drive can stop the simulator cab before hitting the end stops (at the minimum and maximum linear travel limits). This type of limitation takes into consideration the current position and velocity of the simulator cab and the available motion drive power that can be used to stop the cab just short of the end stop. This limiter is called a parabolic-limiter since the position limit (where normal motion is stopped) is a parabolic function of the cab velocity. This type of limiter is also used for rotation motions produced by gimbal systems. These limiters are referenced in most descriptions of cascaded motion systems, such as the FSAA flight simulator at NASA AMES [Sinacori77A] (Figure 2, 3).

2.2. Hexapod System

A much more complicated set of constraints on the simulator cab motion capabilities, are inherent in the use of a "hexapod" moving base system, also called a "synergistic" (in Greek means work together) motion system. A synergistic motion simulator is one wherein the actuators must work in concert in order to display motion purely in a single degree-of-freedom. The hexapod is a very clever way of generating motion in six degrees-of-freedom, with very simple hardware. It was invented independently in 1965 by Peterson and Cappel, and is the most common flight simulator motion system today (Figure 4). Figures 5 is taken from Peterson's patent and describes the operation of the hexapod motion base. In principle six-degrees-of-freedom of motion are obtained by the six independent controls of the lengths' of the six legs of the motion base. The legs' lengths are controlled by hydraulic pistons. It is clear that the limitations on the motion are given in terms of the minimum and maximum leg lengths, their maximum rate of length change and maximum force capabilities. The maximum rate of length change is limited by the maximum hydraulic fluid flow rate, and the maximum force is limited by the maximum fluid pressure. These limitations are fairly simple in the hexapod coordinate system, but become very complicated when transformed to any of the other coordinate systems that are involved in the other parts of the flight simulation (e.g. the airplane equations of motion). The coordinate transformation and an algorithm for real time calculation are discussed by Parrish [Dieudonne72]. In this thesis, the problems involved in using such complicated motion limitations are considered in Chapter II and in Chapter IV using PLQ (Pseudo Linear Quadratic). The performance capabilities of the hexapod motion base system at NASA Langley are shown in Table 1. In the Vertical Motion Simulator (VMS), at NASA AMES Research Center, a hexapod is used to provide the three rotation motions instead of a gimbal system (Figure 6).

2.3. Articulated Beam System

An interesting type of motion system is the articulated beam, or boom type, research motion base manufactured by Northrop. Figure 7 shows the one used at Wright-Patterson Air Force Base. It is named "LAMARS" for Large Amplitude Multi-mode Aerospace Research Simulator. It is a five degree of freedom motion system which consists of a cab inside a sphere (six meters in diameter), that is hinged on the end of a 10 meter beam. The cab is gimbaled so that it can pitch, roll and yaw with respect to the beam. The beam itself can move in a vertical, and in a horizontal plane. The degree of synergy is limited to the interaction between yaw and horizontal beam travel; pitch and vertical beam travel—for lateral and heave motions respectively. The motion limitations of this system are naturally given by the hydraulic actuators limitations that drive the motion system. A summary of the performance of this system is given in Table 1.

2.4. Centrifuge Motion System

All the above motion systems cannot provide a sustained acceleration beyond half a g unit ($5\,m/sec^2$). Therefore centrifuge motion base systems were designed (Figure 8). These simulators consist of a gimbaled capsule mounted on the end of a long arm that rotates at high angular velocities ($30\,rpm$ and accelerations up to $10\,rad/sec^2$). Common linear accelerations values obtained are up to $40\,g$ (with human subjects the accelerations are limited to lower values). The design of a *washout system* for such a motion system involves further complications due to the rotating environment the pilot is in, and is not discussed in this thesis.

2.5. In-Flight Simulators, T-33 and TIFS

In Figure 9 we show two variable stability airplanes which are used as a flying flight simulators. In a simulation of an airplane through the use of variable-stability airplane, the evaluation pilot is, of course, in an aircraft in flight. This sense of actually being in an aircraft improves the simulation. Also, all the degrees of freedom of an aircraft are present, and the motions duplication in the simulator can be done quite well. One can see in Figure 9 that the "Total In-Flight Simulator" (TIFS) has two cockpits one in front for the test pilot and one on top further back for the safety pilot. Furthermore there are vertical aerodynamic surfaces attached to the two ends of the wing in order to aid the simulation of side (sway) forces.

The longitudinal characteristics normally consist of two oscillatory modes, short period and phugoid. In the T-33, the short-period natural frequency can be varied from approximately 1.5 Hz to values less than zero (sic). The phugoid natural frequency can be varied from approximately 0.05 Hz to values less than zero (sic). The pilot control forces are obtained through feel servos and thus their stick force per stick displacement can be varied. Both the natural frequency and the damping of the Dutch roll mode can be varied from 1.0 Hz to less than zero (sic) and damping ratio from 1.0 to 2.0. Other parameters can also be changed.

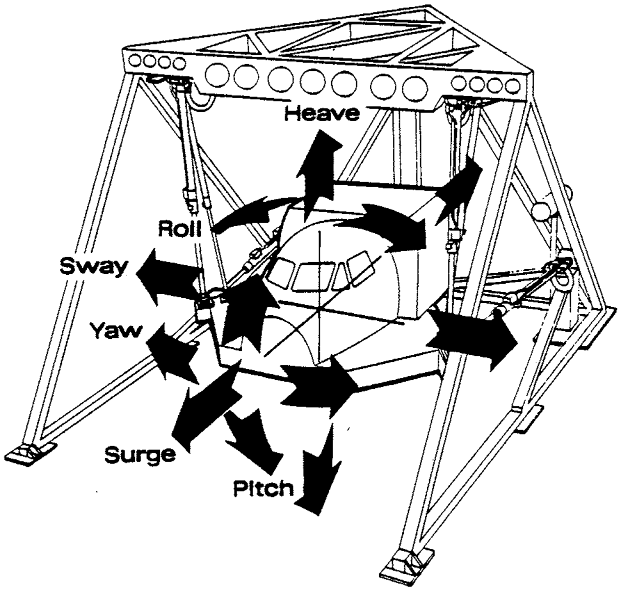


Figure 1. The six degrees-of-freedom of a flight simulator (A Redifon suspended 6 degree-of-freedom motion platform) [Martin80].

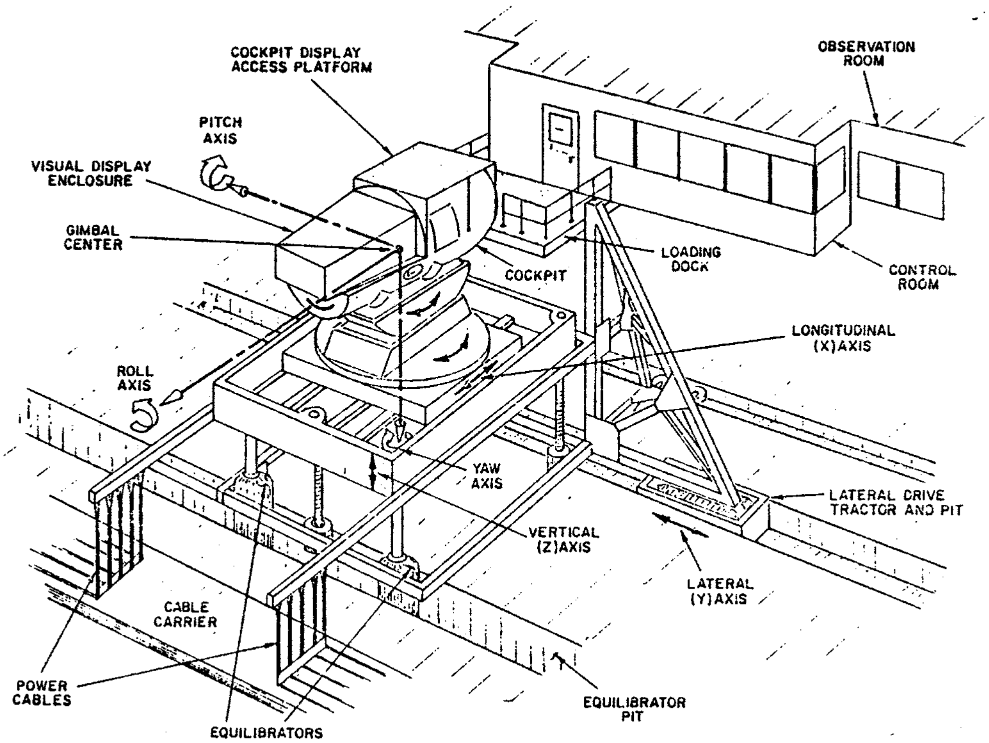


Figure 2. Schematic arrangement of the FSAA flight simulator motion system [Sinacori77A].

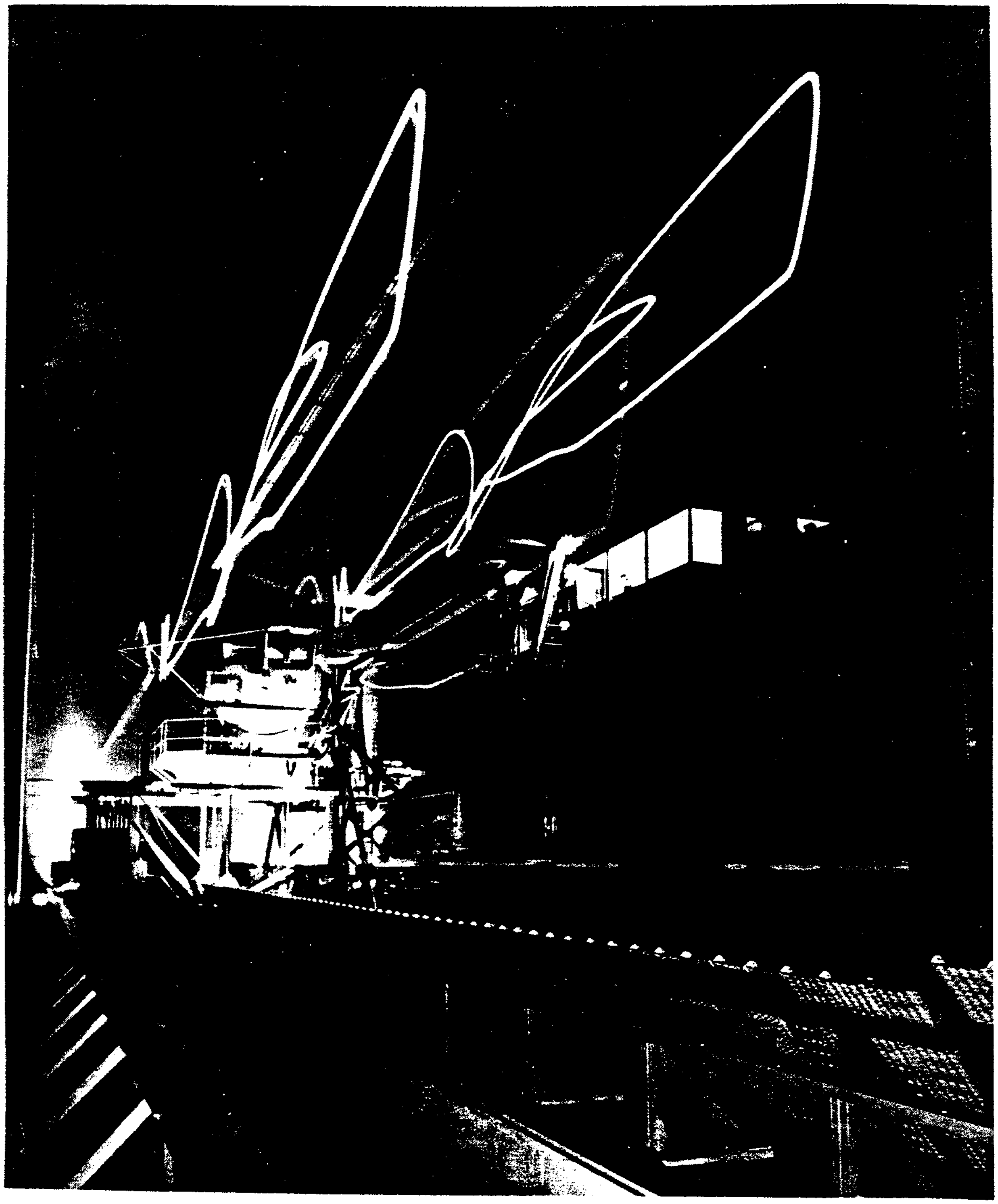


Figure 3. Showing the lateral travel of the FSAA 6 degree-of-freedom motion platform [Martin80].

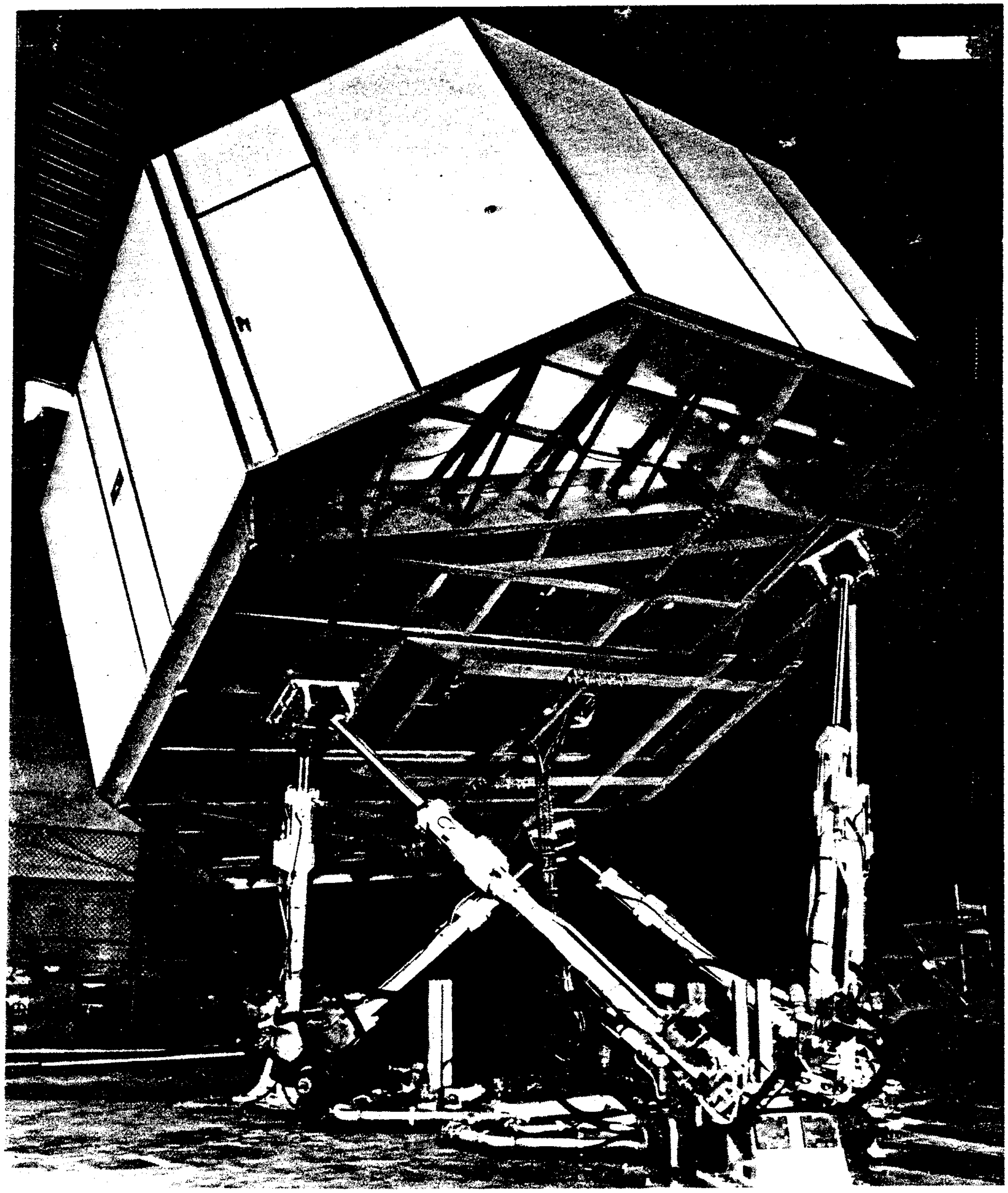
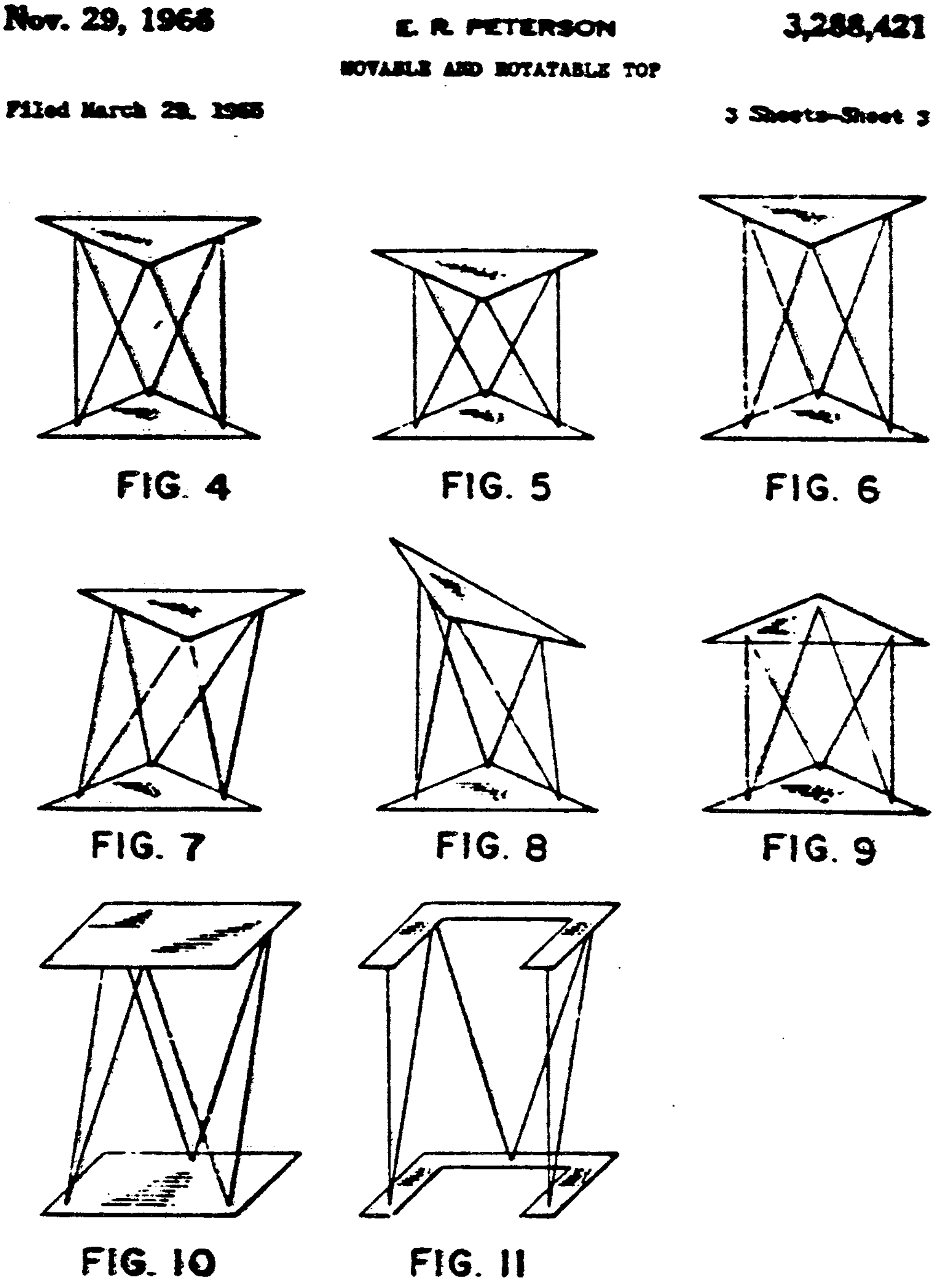


Figure 4. Link hexapod 6 degree-of-freedom motion platform. A typical "6-post" configuration [Martin80].

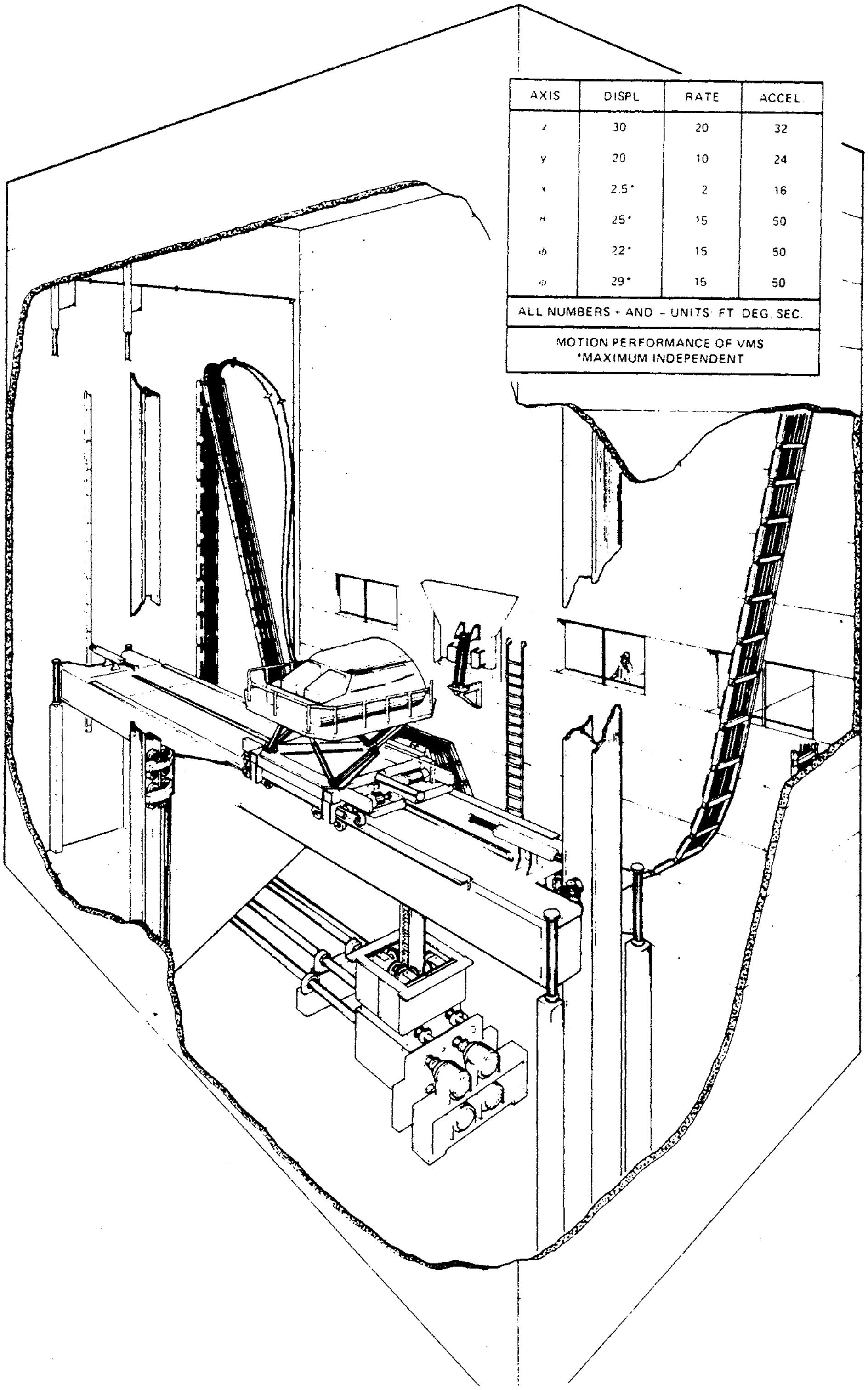


Nov. 29, 1966 E. R. PETERSON 3,288,421

MOVABLE AND ROTATABLE TOP

Filed March 29, 1965 3 Sheets-Sheet 3

Figure 5. Positions of a hexapod motion base. From Peterson's patent diagrams [Puig78].



AXIS	DISPL	RATE	ACCEL
z	30	20	32
y	20	10	24
x	2.5*	2	16
θ	25*	15	50
ϕ	22*	15	50
ψ	29*	15	50
ALL NUMBERS + AND − UNITS: FT DEG. SEC.			
MOTION PERFORMANCE OF VMS *MAXIMUM INDEPENDENT			

Figure 6. Vertical Motion Simulator (VMS) at NASA AMES Research Center [Jones80].

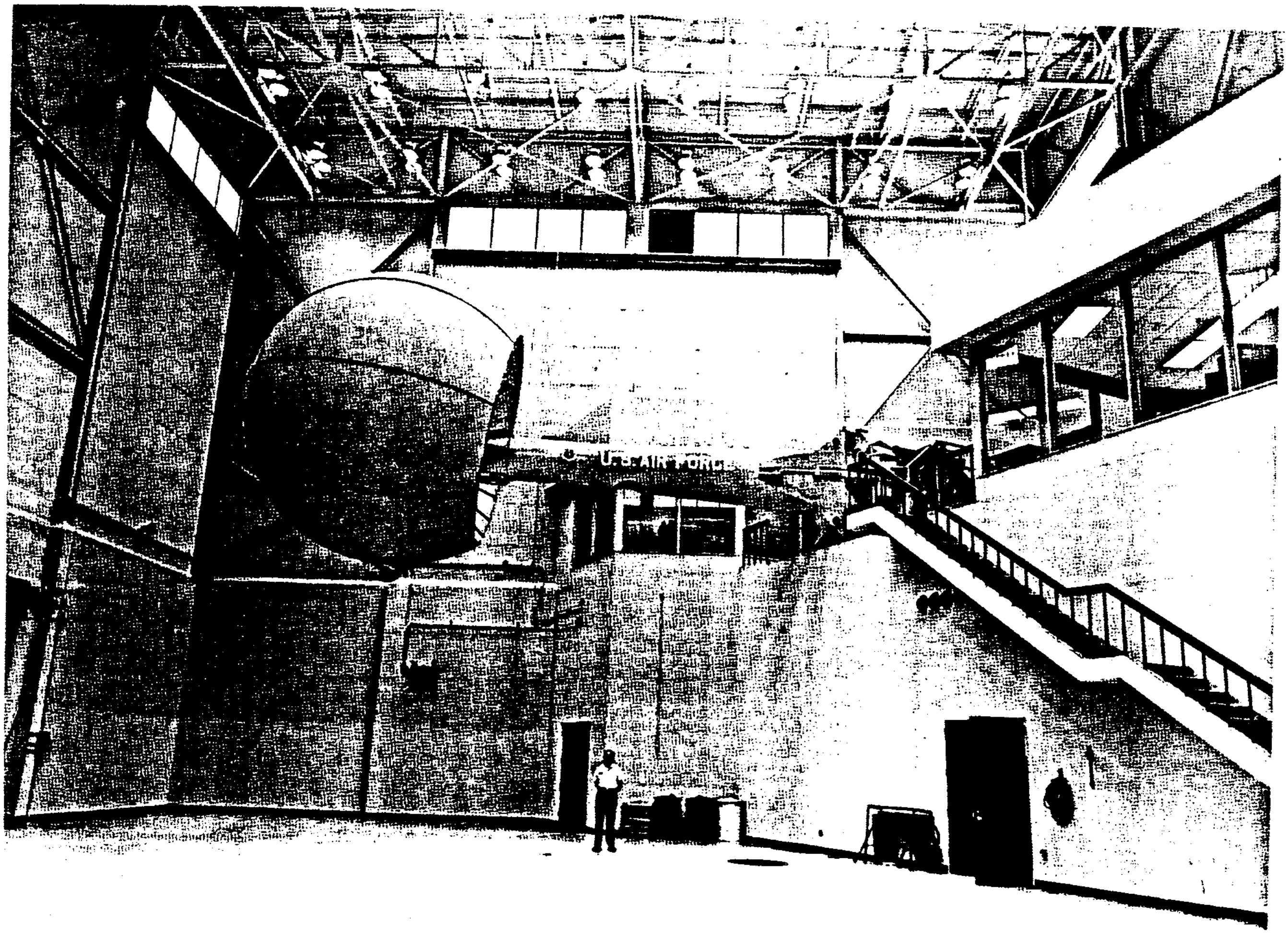


Figure 7. LAMARS 5 degree-of-freedom beam type motion platform [FCDL80].

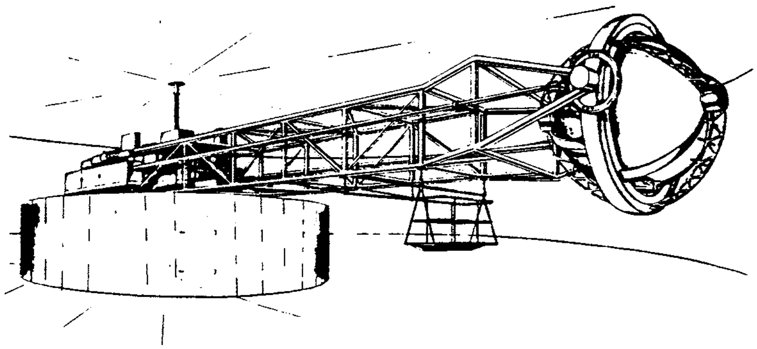
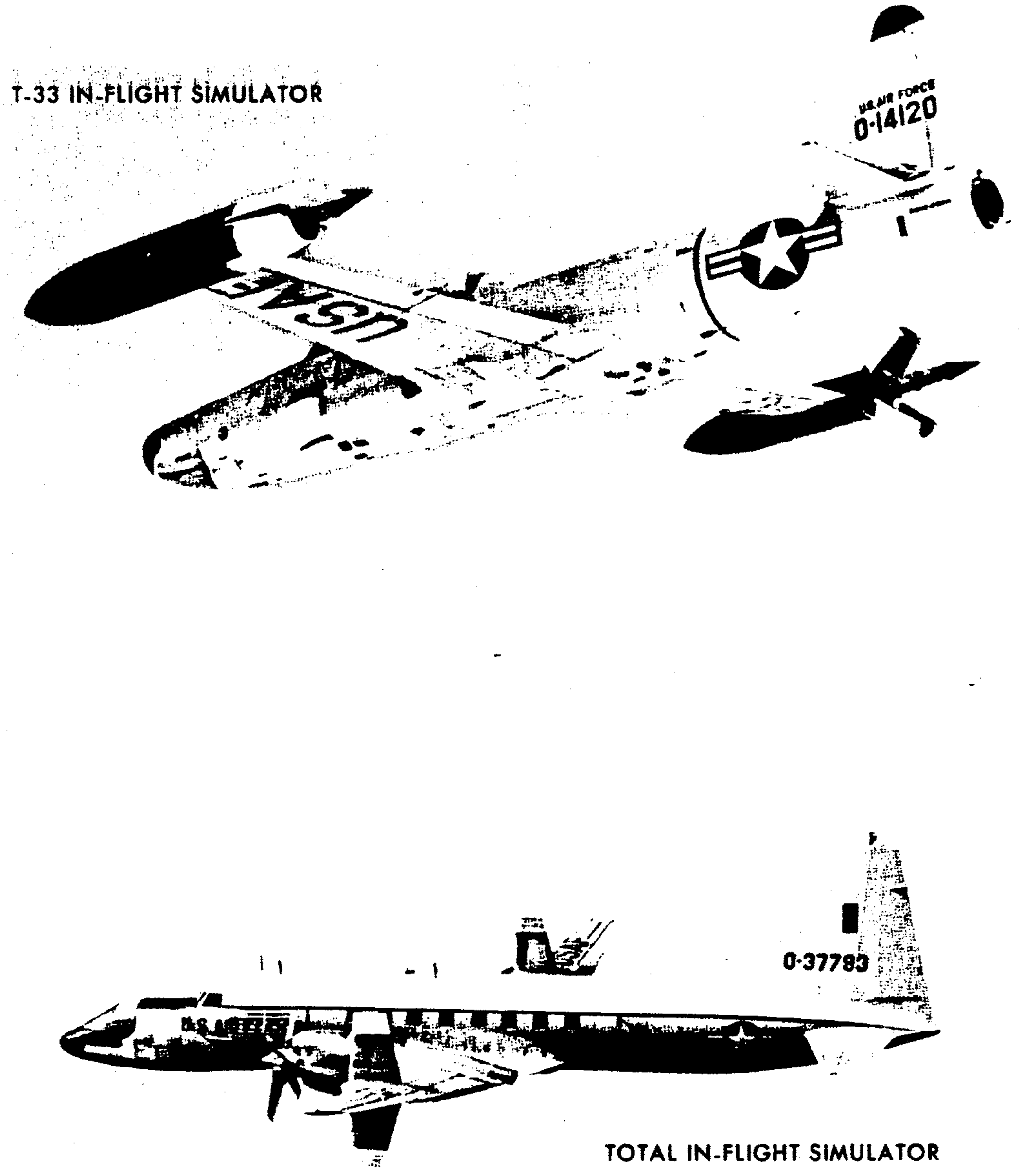


Figure 8. Artist's conception of the modified naval human centrifuge [Von-Gierke61].



T-33 IN-FLIGHT SIMULATOR

TOTAL IN-FLIGHT SIMULATOR

Figure 9. In-Flight Simulators, T-33 and TIFS [FCDL80].

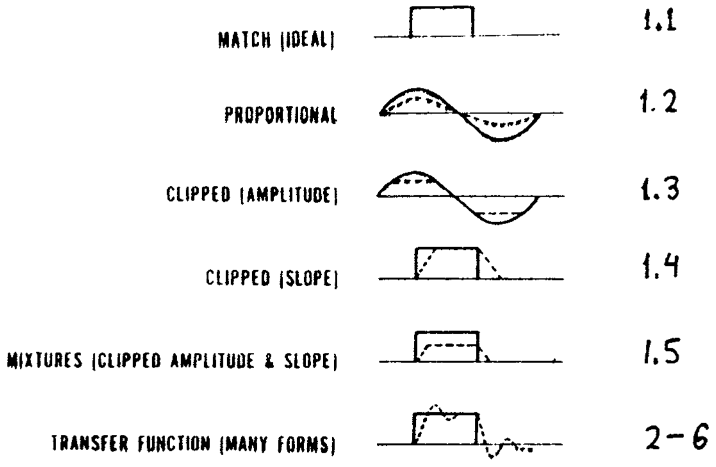


Figure 10. Characteristic response of several washout filter types (adapted from [Puig78]).

		RECOMMENDATIONS FOR FIGHTER ACFT ATDS.	VERTICAL MOTION SIMULATOR (VMS)	FLIGHT SIM FOR ADVANCED AIRCRAFT (FSAA)	USAF "6-POST" SPECIFICATIONS.	REDIFON SUSPENDED 6-DOF	LAMARS BEAM 5-DOF
EXCURSION							
HEAVE	(FT)	± 0.33	± 30	± 5.	± 2.8	± 4.0	± 10.
LATERAL	(FT)	-	± 20	± 50.	± 2.8	± 6.0	± 10.
SURGE	(FT)	-	± 2.5	± 4.	± 2.8	± 2.9	-
ROLL	(DEG)	± 11.5	± 22	± 45.	± 20.	± 19.	± 25.
PITCH	(DEG)	± 11.5	± 25	± 22.	± 25.	± 28.	± 25.
YAW	(DEG)	± 11.5	± 29	± 30.	± 20.	± 13.	± 25.
VELOCITY							
HEAVE	(FT/SEC)	± 2.3	± 20	± 8.6	± 2.0	± 2.5	± 13.
LATERAL	(FT/SEC)	-	± 10	± 17.0	± 2.0	± 2.5	± 10.
SURGE	(FT/SEC)	-	± 2	± 6.3	± 2.0	± 2.5	-
ROLL	(DEG/SEC	± 25.8	± 15	±101.0	± 20.0	± 12.	± 60.
PITCH	(DEG/SEC)	± 25.8	± 15	± 40.0	± 20.0	± 17.	± 60.
YAW	(DEG/SEC)	± 25.8	± 15	± 40.0	± 20.0	± 11.	± 50.
ACCELERATION							
HEAVE	(GS)	0.5	1	0.37	0.8	.75	3.0
LATERAL	(GS)	-	0.75	0.37	0.6	.25	1.6
SURGE	(GS)	-	0.5	0.31	0.6	.50	-
ROLL	(DEG/S/S)	57.3	50	229.	60.	80.	460.
PITCH	(DEG/S/S)	57.3	50	115.	60.	80.	400.
YAW	(DEG/S/S)	57.3	50	115.	60.	80.	200.

Table 1. Comparison of motion platform operating envelopes [Martin80]

3. Survey of Present Washout Filters

Currently only washout systems that do not include the limiting logic, the axis transformation and the control system for the motion-base are used. In this case the washout system is termed a washout filter. In simple cases the washout filter can degenerate to just a constant gain. The following concepts are used in the design of washout filters i.e. the transformation between the computed airplane motion and the simulator motion (the first six are quoted from [Puig78]):

1. Memoryless, linear (1.1–1.2) and nonlinear (1.3–1.5):

    1.1. The aircraft acceleration concept—The magnitude of the motion system acceleration is equal to the magnitude of the aircraft acceleration; ideal, wishful case.

    1.2. The proportional concept—The magnitude of the motion system acceleration is always proportional to the magnitude of the aircraft acceleration.

    1.3. The clipped magnitude concept—The slope of the motion system acceleration cannot exceed a set limit.

    1.4. The clipped slope concept—The slope of the magnitude of the motion system acceleration cannot exceed a set limit.

    1.5. The mixed concept—Any combination of the concepts, 2, 3, 4 above.

    These concepts are also applied to velocity and position variable as well as acceleration. An example of concept 1.2 is the pitch and roll motions on the Link GAT-1; the simulator pitch angle is 1/2 of the computed pitch angle and the simulator roll is similarly 1/6. Figure 10 depicts these general washout categories.

2. Linear time-invariant system referenced as a "transfer function onset and washout concept[1] —The magnitude of the motion system acceleration and phase is determined by shaping filter techniques, i.e., aircraft acceleration subjected to a predetermined transfer function" [Puig78]. The filters used are up to 4th order low pass, band pass and high pass filters. Based on experience a set of design rules was put together by Sinacori [Sinacori77S]. An extensive effort to define a cost function and us it to optimize the parameters of the washout filter for the LAMARS is presented in [Hofman79]. Hosman further elaborated by optimizing the washout parameters using also a vestibular model. A comparison study

[1] A washout concept is defined as the methodology in determining the motion of the simulator cockpit in order to washout the results of the onset cue, i.e., the velocity and position change, at subthreshold levels to allow the motion system to either return to the neutral position or a position such that the gravity vector is substituted for sustained linear acceleration (gravity align or g-tilt)

of several washout filter implementations was done recently by Michaeli [Michaeli81].

3. Adaptive washout filter (nonlinear system)—on-line optimization of the parameters of a linear washout [Parrish73].
4. LQ optimal washout filters—Linear system based on a Quadratic cost function which is designed using LQ optimal control [Kurosaki78], [Sturgeon81], [Sivan82]. In this method the structure of the optimal washout is fond, based on the assumptions made in the problem formulation. The current work of this thesis is based on this concept but it also uses vestibular model in the formulation of the cost function [Sivan82].
5. Nonlinear optimal washout filters—Nonlinear system design based on quadratic or "higher" than quadratic cost function (not using any model for the pilot). One design was done Friedland et. al. and is based on approximation to optimal control [Friedland66], [Friedland68], [Friedland70], [Friedland73]. Another conceptual design example was derived by Kosut [Kosut79] assuming a linear plant but a quartic cost function which leads to a nonlinear washout filter.
6. Washout system—A washout filter combined with the control system for the motion-base. A model following structured system was suggested by Sturgeon [Sturgeon81]. In this thesis an optimal washout system (OWS) concept is discussed in Chapter VI. In our OWS implementation we also include the axis transformations.

4. Basic Approach

An abstract simulator design problem is formulated as follows: Given a dynamic system, S^a, called the *actual* system and another dynamic system, S^s, called a *simulator* for S^a, and given a function which drives the system S^a, the problem is to find an operator, properly constrained, which will generate the input to S^s on the basis of the input to S^a, such that the discrepancy between the outputs of S^a and S^s will be as small as possible. This abstract simulator design problem is formulated as an optimal control problem and in the linear-quadratic case presented in Chapter III this problem is decomposed into two separately solvable subproblems: (i) deterministic and (ii) stochastic. Both subproblems are solved; the stochastic one for the Gaussian case only. An examination of the properties of the solution puts in evidence a parallel decomposition theorem and provides an interpretation of the dependence of the simulator design on the given parameters. These and other properties simplify the solution enabling its extension to include several nonlinear effects developed in Chapter IV and Chapter VI.

The study of the nonlinear effects in Chapter IV includes three topics. The first topic is an extension of the deterministic-stochastic decomposition to include nonlinear dynamic system equations using a quadratic cost function. This decomposition shows a general method of how to separate and then combine the "open-loop" (deterministic) and the "closed-loop" (stochastic) solutions for the abstract simulator design problem.

The second topic is the development of a *Pseudo Linear Quadratic* controller (PLQ) for linear and nonlinear dynamic systems. The PLQ controller is derived from the standard *Linear Quadratic* (LQ) optimal control solution by solving for a quasi-quadratic cost and a quasi-linear system for each value of the state. This results in a feedback with a leading linear term, i.e. using feedback gains that are functions of the system state rather than constants. Though PLQ is not a solution to any known formulated optimization it is an extension of the standard LQ control. Furthermore, in the cases tested it has properties that match those of known optimal nonlinear controllers derived for linear dynamic systems using a nonquadratic cost. On the other hand, PLQ is easier to compute and easier to implement, due to its "linear" form. Many of the PLQ properties still need to be developed including conditions for global stability for the multi dimension dynamic system case. It is expected that the resulting PLQ controller would show similar robustness properties as the LQ controller.

The third nonlinear effect discussed is a *sign sensitive cost* formulation and solution. The cost function is put into a form that includes a correlation function term that is evaluated between the outputs of the systems $\mathcal{S}^a$ and $\mathcal{S}^s$. It is shown that any antisymmetric compressive memoryless output function cascaded to the linear dynamics of both $\mathcal{S}^a$ and $\mathcal{S}^s$ leads to a cost function that includes a *sign sensitive* term. This problem is put into a LQ form which no longer has a positive definite cost. It is shown that a unique solution exists for the abstract simulator design problem. Putting all these elements together enables one to develop a methodology for the design of abstract optimal simulators.

Next, the solution and properties of the abstract simulator problem are applied to the design of motion generation for moving-base flight simulators. The formulation and approximation processes used to fit the flight simulator motion problem into the form of the general abstract simulator problem are discussed in Chapter II. The optimization criterion selected is a quadratic norm of the difference between the physiological outputs of the vestibular organs of a pilot in an imaginary reference *airplane* and those of a pilot in the *simulator*. In the design of the motion for a flight simulator vestibular models based on physiological and psychophysical experiments are used. This includes consideration of vestibular sensor saturation and the multiplicative nature of physiological noise in the nervous system. This latter is modeled by an antisymmetric compressive memoryless output nonlinearity (in Chapter IV). The LQ abstract simulator properties imply a 2-2-1-1 physical axis decoupling theorem for the feedback gains, i.e. Pitch-Surge, Roll-Sway, Yaw, Heave axis group decoupling. The 2 axis coupling is due to gravity. This 2-2-1-1 rule is well known to designers of simulator motion systems. What is usually overlooked, however, is that the feed-forward gains do not decouple the same way due to the effect of the airplane dynamics coupling (Chapter III). Axis transformations are included in the motion system implementation in Chapter VI. In this case coupling between all six physical axes is obtained, a property not existing in current designs. Using a similar method we also include head rotations by considering a head axis system for each pilot, airplane and simulator (Chapter II.9.4 and Chapter VI), which should further improve the simulator motion sensation.

Putting all these elements together we obtain a new design methodology for the design of motion for moving-base flight simulators. This design methodology is demonstrated by several design examples that are solved and simulated in Chapter V. The examples and solution properties conform to the set of empirical design rules used by experienced engineers to determine: the structure (2-2-1-1 theorem), the initial setting of the pole locations; the expected lower motion fidelity as the poles' frequency increases; and the cross coupling gain between the linear and rotation motion input (called g tilt) used in flight simulator motion generation systems.

Finally the design method was implemented and tested by twenty pilots using several experiments (Chapter VII). These designs were implemented for the pitch and surge axes on a Link GAT-1 General Aviation flight simulator Trainer.

These tests confirm the suggested design method, including equal weighting of the normalized vestibular linear and rotation components. In this system, a PLQ (Pseudo Linear Quadratic, developed in Chapter IV) was used to build a nonlinear-motion-generation system for the Link GAT-1 simulator that better accounts for the hard limits of the pitch motion.

5. What to Read in order to Design an Optimal Washout System

First read the following:

(i) Summary—over view of the whole thesis.

(ii) Chapter VIII, Section 1—elements in the design of an Optimal Washout System (OWS).

(iii) Chapter I, introduction and Sections: 1—what is the definition of the problem, 4—basic approach used.

(iv) Chapter II, Section 1, paragraph 1—washout system definition.

(v) Chapter II, Figure 1—main approach used here.

(vi) Chapter II, Section 2, paragraphs 1, 2 and last one (see Figure 3)—class of airplane motion definition.

(vii) Chapter II, Section 5, paragraph 1—cost function.

(viii) Chapter II, Section 6, paragraph 1, 2, 3, Figures 5 and 6—Optimal Washout System (OWS) Design Problem, use of sensory comparison as performance criteria.

(ix) Chapter II, Section 6, last titled paragraph—optimization criteria.

(x) Chapter II, Subsection 9.1, 9.2 and 9.5—axis systems.

(xi) Chapter III, introduction.

(xii) Chapter III, Section 1—Linear Quadratic (LQ) problem statement.

(xiii) Chapter III, Section 2 and Figure 1—deterministic-stochastic problem separation.

(xiv) Chapter IV, introduction.

(xv) Chapter IV, Subsection 2.3 example of PLQ (Pseudo Linear Quadratic)—how to do a nonlinear design to better account for the finite limits of the simulator motion.

(xvi) Chapter V, introduction, Section 2 and 3—example of two-degree-of-freedom motion design. Study this example thoroughly.

(xvii) All Chapter VI, skip derivations in equations (18)–(25)—how to implement an Optimal Washout System (OWS) and take into account the nonlinearities due to axes transformations.

(xviii) Chapter VII, Figure 2—detailed example of OWS block diagram.

(xix) Chapter VII, use equations (1)–(6) for your vestibular model realization.

(xx) Chapter VII, if your motion-base is unstable (with no control) read also Subsection 2.2 paragraph one before last and look at Figure 6—limiting logic for an unstable motion-base.

(xxi) Chapter VIII, introduction, Section 1—elements in the design of an Optimal Washout System (OWS).

(xxii) Chapter VIII, Section 2—conclusions from the use of an OWS with the Link GAT-1 three degree-of-freedom (rotations) flight simulator.

In the second reading also go over:

(i) Chapter II, Section 2—modeling of the airplane anticipated motion.

(ii) Chapter II, Section 5—cost function formulation.

(iii) Chapter II, end of Section 6 after Table 1—choice of sensory comparison for a performance index.

(iv) Chapter II, Section 9—axes system and head motion consideration.

(v) Chapter II, Section 10—abstract optimal simulator design problem statement.

References

[Dieudonne72] Dieudonne, J.E., Parrish, R.V. and Bardusch, R.E., "An Actuator Extension Transformation for a Motion Simulator and an Inverse Transformation Applying Newton-Raphson's Method," NASA Langley Research Center report TN D-7067, November, 1972.

[FCDL80] U.S. Air Force, "Flight Control Development Laboratory," Technical Brochure, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

[Friedland66] Friedland, B., Thau, F.E., Cohen, V.D., and Ellis, J., "Study of Quasi-Optimum Feedback Control Techniques," NASA Report CR-527, AMES Research Center, Aug., 1966.

[Friedland68] Friedland, B., Thau, F.E., Welt, S., Ling, C.K., and Schilder, M., "Additional Studies of Quasi-Optimum Feedback Control Techniques," NASA Technical Report CR-1099, AMES Research Center, July, 1968.

[Friedland70] Friedland, B. and Ling, C.K., "Quasi-Optimum Design of Control Systems for Moving-Base Simulators," NASA Report CR-1614, AMES Research Center, Oct., 1970.

[Friedland73] Friedland, B., Ling, C.K. and Hutton, M.F., "Quasi-Optimum Design of a six Degree of Freedom Moving-Base Simulator," NASA Report CR-2312, AMES Research Center, Oct., 1973.

[Hofman79] Hofman, L.G. and Riedel, S.A., "Manned Engineering Flight Simulation Validation, part I: Simulation Requirements and Simulator Motion System Performance," Systems Technology , Inc., AFFDL-TR-78-192, Feb., 1979.

[Hosman79] Hosman, R.J.A.W., van der Vaart, J.C. and van de Moesdijk, G.A.J., "Optimalization and Evaluation of Linear Motion Filters," *Annual Conference on Manual Control*, March, 1979.

[Jones80] Jones, A.D., "OPerations Manual: Vertical Motion Simulator (VMS) S.08," NASA Technical Memorandum 81180, May, 1980.

[Kosut79] Kosut, R.L., "Nonlinear Optimal Cue-Shaping Filters for Motion Base Simulators," *Journal of Guidance and Control*, **2**, pp. 486–490, Dec., 1979.

[Kurosaki78] Kurosaki, M. "Optimal Washout for Control of a Moving Base Simulator," Proceeding of the Seventh Triennal World Congress of Ffac, Helsinki, Finland, **2**, pp. 1311–1318, June, 1978.

[Martin80] Martin, E.A., "Motion and Force Simulation Systems," Simulator Course at Wright-Patterson Air Force Base, Ohio, 1980.

[Michaeli81] Michaeli, S., "Principles and Realizations of the Drive-Logic in Flight-Simulators," M.Sc. Final Paper, Technion—Israel Institute of Technology, Haifa, Israel, July, 1981.

[Parrish73] Parrish, R.V., Dieudonne, J.E., Bowles, R.L. and Martin, Jr. D.J.,, "Coordinated adaptive Washout for Motion Simulators," presented at AIAA Conference, Palo Alto, California, 1973.

[Puig78] Puig, J.A., Harris, W.T. and Ricard, G.L., *Motion in Flight Simulation: An Annotated Bibliography*, Technical Report: NAVTRAEQUIPCEN IH-298, Naval Training Equipment Center, Orlando, Florida,July, 1978.

[Sinacori77A] Sinacori, J.B., Stapleford, R.L., Jewell, W.F. and Lebman, J.M., "Researcher's Guide to the NASA AMES Flight Simulator for Advanced Aircraft (FSAA)," NASA CR-2875, August, 1977.

[Sinacori77S] Sinacori, J.B., The Determination of Some Requirements for a Helicopter Flight Research Simulation Facility, Technical Report no. 1097-1, Contract NAS2-9421 (CR-152066), Systems Technology, Inc., pp. 2–12, Sep., 1977.

[Sivan82] Sivan, R., Ish-Shalom, J. and Huang, J.-K., "On the Design of Optimal Simulators with Application to Flight Simulators", *IEEE Transactions on Man Systems and Cybernetics*, SMC, Nov., 1982.

[Sturgeon81] Sturgeon, W. R., "Controllers for Aircraft Motion Simulators," *Journal Guidance and Control*, 4, no. 2, pp. 184–191, April, 1981.

[Von-Gierke61] Von Gierke, H.E. and Steinmetz, E., *Motion Devices for Linear and Angular Oscillation and for Abrupt Acceleration Studies on Human Subjects (Impact)*, Publication 903, National Academy of Sciences—National Research Council, Washington, D.C, 1961.

Chapter II

Formulation of the Motion Problem

The objective of this chapter is to arrive at a mathematical formulation of the *simulator motion design problem.* A simplified version of this formulation is solved in the next chapter by the use of optimal control techniques, which leads to an *optimal design* for the simulator motion. The content of each section is as following:

1. Presentation of the flight simulator parts and their limitation.
2. The class of input motions that the optimization has to consider is defined by a model that generates the anticipated simulated airplane motion.
3. Conceptual pilot block diagram construction.
4. Comparison—in order to develop a motion quality criteria for the motion the flight simulator gives to the *simulator pilot,* we postulate an idealized *imaginary* reference pilot, called the *airplane pilot.* The *airplane pilot* is in an *imaginary* airplane and flies the same task as the *simulator pilot.* Beyond the difference in "airplanes" both pilots respond to a given stimulus identically. Comparison between the *simulator pilot* and the *airplane pilot* gives us several possible criteria for optimal design of flight simulator motion:
   (i) Cockpit motions—try to match the simulator motion to that of the airplane.
   (ii) Sensory measurements—try to match the output of the inertial motion sensory models of both pilots.
   (iii) Orientation estimate—try to match the output of the "orientation sensation model" of both pilots.
   (iv) Control effort—try to match the *simulator pilot* controls to the expected *airplane pilot* controls, predicted from a pilot model.
   (v) Task performance—try to match the *simulator pilot* performance to that expected from the *airplane pilot.*

   In section 6 we argue for our choice to use the *sensory measurements* comparison.

5. An optimization function (cost) is constructed on the basis of the above comparison criteria and the given simulator motion limitations.
6. Construction of our view of the *Optimal Flight Simulator Design Problem.*
7. The optimal flight simulator solution in chapters III, V–VII, uses the vestibular organ to represent all the inertial motion sensors; a discussion of some experimental evidence for this assumption is presented.
8. A description of the vestibular organ and the limitations of the models used in the next chapters to represent it.
9. Which axis systems are involved in this problem, and how do we simplify the problem to obtain a *Linear Quadratic* version of the *Optimal Flight Simulator Design Problem*?
10. A general *Abstract Optimal Simulator Design Problem* is constructed. This general formulation can fit other problems such as model following and robotic manipulators.

A *Linear Quadratic* version of the *Abstract Optimal Simulator Design Problem* is solved in chapter III and some nonlinear extensions are discussed in chapter IV.

1. The Flight Simulator

Flight simulator parts

The flight simulator has two major parts which are shown in Figure 1. The SIMULATOR FLIGHT COMPUTATION block takes the simulator pilot's controls and computes the simulated airplane's motion while taking into account the expected type of flight disturbances. This computed airplane motion is then used by the SIMULATOR DISPLAY UNIT to compute and send proper commands to the flight-instrument readings, pilot-control reactions, drive commands for the pilot's visual display, and the flight simulator cockpit motion. The washout system is defined as follows:

> WASHOUT SYSTEM
>
> It is the part of the simulator display unit which computes the simulator motion-base commands on the basis of the computed airplane motions, so as not to exceed the motion-base constraints, yet retain the simulation flight realism as much as possible.

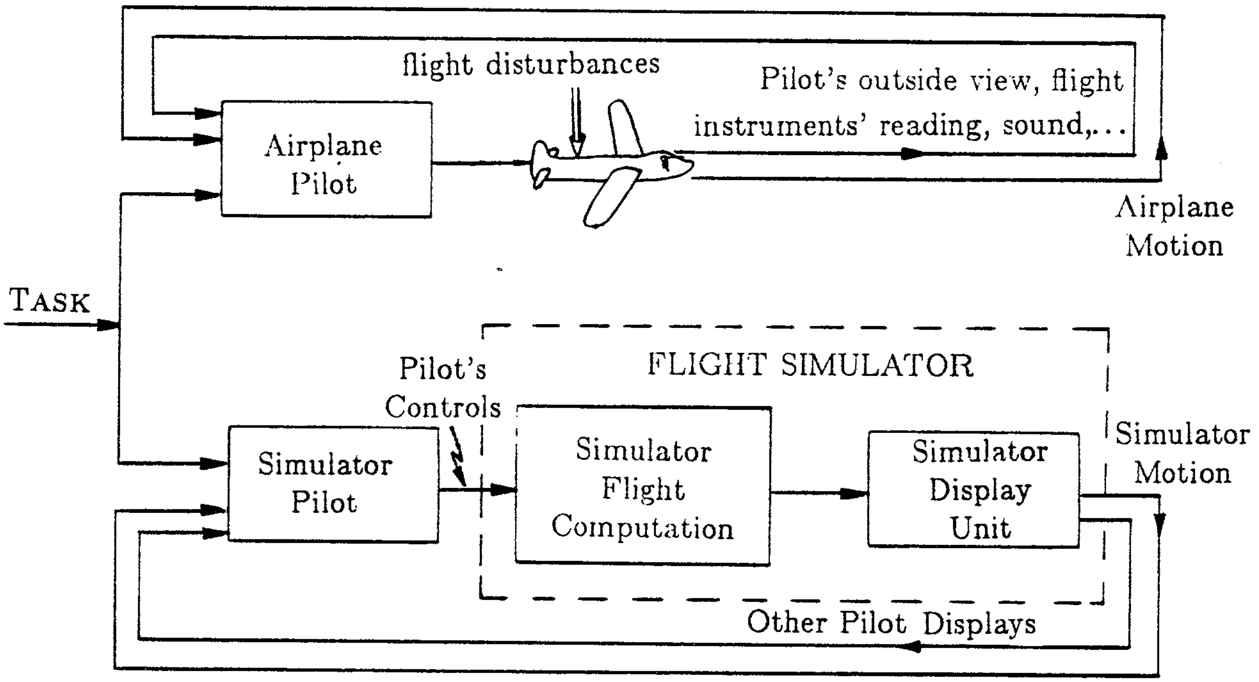


Figure 1. Comparison of airplane flight to simulated flight.

Flight simulator limitations

Typical discrepancies arising from the substitution of a flight simulator for an actual airplane are described below. The major discrepancy is in the airplane computed motions which arise from the airplane dynamics computation limitations. These are limited by the lack of knowledge of the exact airplane characteristics and by the inability to compute even all the known airplane dynamics in real time. Less important but still crucial, the visual display does not give a realistic picture of the world outside the cockpit; even a good display gives only the most important information the pilot requires. Third, the simulator motion, which is very difficult to generate properly. Last, the responses of the pilot's controls are also a common problem.

This thesis addresses the simulator motion generation problem. This problem is quite difficult, even when using the largest ground base motion flight simulator in the world (located at NASA AMES, which has up to 24 meters (80 feet) of linear travel). In cases where motion is crucial, the best solution, thus far, is to use a *flying airplane* as a simulator [FCDL80, Von-Gierke61].

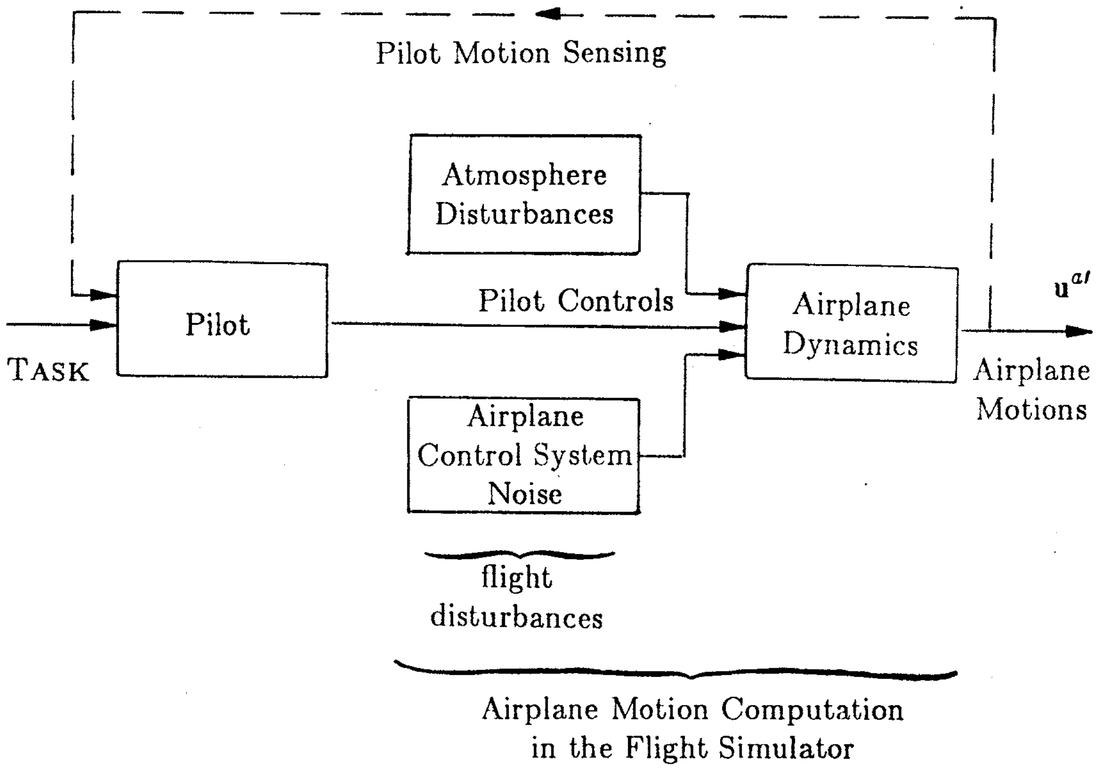


Figure 2. Functional building blocks that generate the airplane motion in an actual and simulated airplane.

2. Modeling of the Airplane Anticipated Motions

Purpose and strategy

One **purpose** of modeling the airplane anticipated motions, $\mathbf{u}^a$, is to describe to the mathematical procedure that designs the washout system our knowledge of the class of inputs the washout system has to handle. Secondly this model is used to construct several inputs that are used in the washout system implementation.

The **strategy** behind constructing this model has two conflicting elements: (a) Incorporating in it as much knowledge of the anticipated airplane motion as possible. (b) Restricting it so that it will still be sufficiently simple in the following two senses: (i) it could be "handled" mathematically in the design procedure; (ii) results in a simple enough washout system that is "reasonable" to implement.

Building blocks

Let us look at the functional building blocks of the system that generates the airplane motion (Figure 2). The system's input is the pilot's TASK and its output is the airplane motion. We are only interested in the airplane *computed* motion of the flight simulator, therefore our undertaking of modeling the anticipated airplane motion is considerably simplified. We do not deal here with how well the flight simulator motion computation matches that of the real airplane, but how to "reasonably" model the *implemented* airplane motion which is driven by our "unpredictable" human pilot. In contrast the FLIGHT DISTURBANCES due to ATMOSPHERE DISTURBANCES and AIRPLANE CONTROL SYSTEM NOISE are generated by the flight simulator and are only unpredictable to simulator pilot, but deterministic from the point of view of the simulator motion designer. The TASK is also a deterministic input since it is dictated by the simulator operators. The task specifications can be given as specific instructions, or could involve other inputs such as the motions of an enemy airplane the simulator pilot is trying to shoot down. Over all, the pilot's behavior is the only element in the airplane motion generation system that is not completely deterministic—which is somewhat surprising.

Airplane dynamics modeling

The AIRPLANE DYNAMICS are at least a double integration of the acceleration in the six degrees-of-freedom of the airplane—three linear and three angular. These equations are nonlinear and may involve table lookups for their parameters in different flight regimes (possibly "more" linear using Euler parameters [Ramnath80]). Furthermore the linear degrees of freedom are usually computed in wind axes and the angular ones in body axes [Etkin72], which requires us to use nonlinear axes transformations in order to obtain the computed airplane motion in the coordinate system used by the washout system design program. A *simple* example is the set of equations used by the Link GAT-1 flight simulator given in appendix A.

Airplane disturbance modeling

The ATMOSPHERE DISTURBANCES generation are many times generated by non-stationary pseudo random processes that are shaped to have a given spectrum and possibly other required properties. The AIRPLANE CONTROL SYSTEM NOISE may have similar characteristics.

Pilot modeling

The pilot is the most problematic element to model. We can use human operator models such as the classical control cross-over model [McRuer65], or the optimal control model [Baron76, Kleinman71], but they still do not give a complete answer to predict the pilot behavior in flight, which changes his flight training advances. A further difficulty is that we should take into account all the different ways the pilot obtains information about the airplane motion, i.e. his visual seen out of the window, his flight instruments and his inertial motion sensation.

Effect of the difference between the simulator and airplane motions

A further problem is the that the simulator motion depends on the washout system design. This problem is overcome by the following assumption.

Assumption: One or both of the following statements are assumed to be true:

(i) The simulator motion is well designed so that the simulator motion sensation is *identical* to that in the actual airplane in the same flight situation.

(ii) The motion has very little or no influence on the pilot's controls to the airplane.

Using this assumption we simplify the washout design by saying that the *airplane motion generator* is a black box whose output is independent of the washout design. This simplification is a good assumption as long as the washout system design and the rest of the simulator displays give the simulator pilot a sensation that is sufficiently similar to that perceived in the actual airplane.

Airplane motion generator characteristics

It should be noted that usually the airplane motion has a mean value different from zero (Figure 3). The mean is the time varying expected value of the airplane motion over the set of all repetitions of one pilot performing a given task, and/or the repetitions of a group of pilots which fly the flight simulator that specific task.

3. Pilot Block Diagram

In this section we discuss a conceptual model to describe both the *airplane pilot* and *simulator pilot*. This pilot model is used to compare the motion sensation

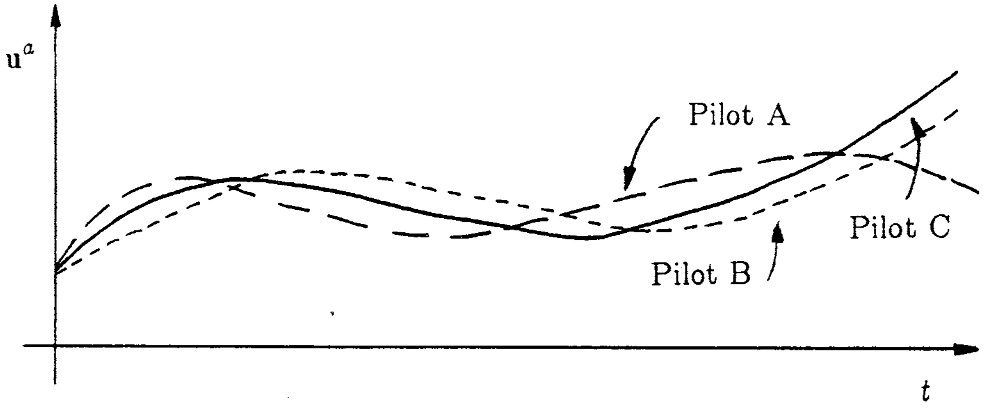


Figure 3. Schematic motions of an airplane, $\mathbf{u}^a(t)$, for three pilots that are required to perform the same task. Note that $\mathfrak{E}\{\mathbf{u}^a(t)\} \neq \mathbf{0}$.

of the two pilots so as to specify the simulator motion quality criteria used in the simulator motion optimization problem formulation.

3.1. Conceptual Building Blocks

Pilot block diagram

An airplane pilot can be looked upon as a feedback controller of the airplane he is flying. His outputs are the control commands to the airplane, such as stick movements and throttle settings. His direct input is his task, and beyond that he has many sensory inputs, which among other things enable him to orient himself. These can be divided into two categories: inertial motion sensors and other orientation sensors. The most important human inertial motion sensors are the vestibular and tactile. Of these two, the most sensitive one is the vestibular organ, which is sometimes called the balancing organ. The other orientation sensors include the visual and auditory sensors: the visual sense takes in both the information from the flight instruments and what is seen out through the window. We use these two categories of sensory inputs as the front end blocks in the postulated functional block diagram for the airplane pilot, described in Figure 4(a). These two categories of sensors are inputs to an orientation estimator, which is the conceptual part of the pilot's brain which ascertains the pilot's orientation as well as that of the airplane. This best orientation estimate, combined with the required task, are the inputs to the part of the brain, considered here as the "real" airplane controller,

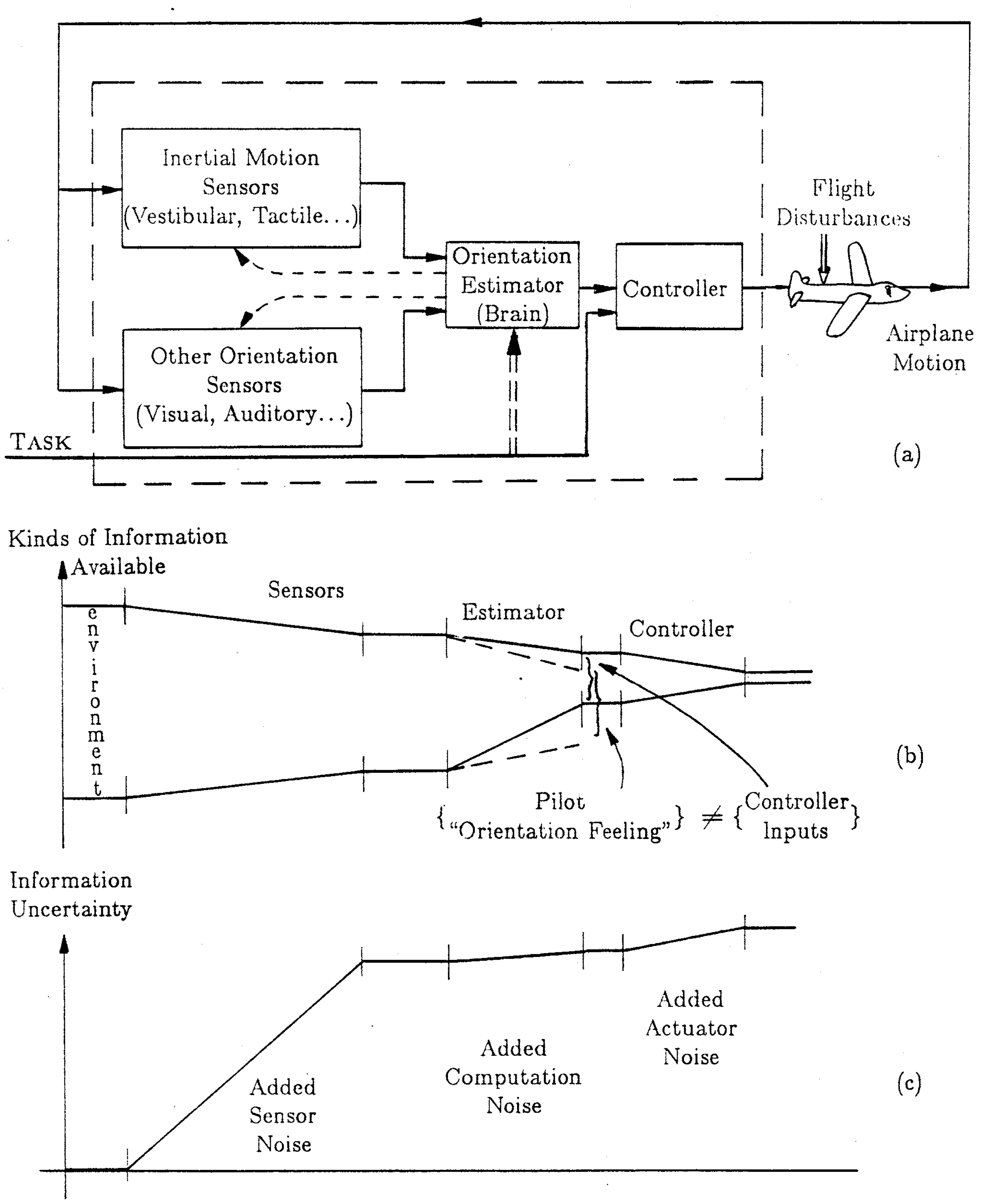

Inertial Motion Sensors (Vestibular, Tactile...)
Other Orientation Sensors (Visual, Auditory...)
Orientation Estimator (Brain)
Controller
Flight Disturbances
Airplane Motion
TASK
(a)
Kinds of Information Available
environment
Sensors
Estimator
Controller
{Pilot "Orientation Feeling"} ≠ {Controller Inputs}
(b)
Information Uncertainty
Added Sensor Noise
Added Computation Noise
Added Actuator Noise
(c)

Figure 4. Airplane pilot: (a) functional block diagram; (b) information flow; (c) information uncertainty growth.

which instructs the pilot's hands how to move the airplane controls.

Assumption of no feedback from the CNS to the sensory organs

Let us go back and specify the assumptions made in constructing the above functional block diagram for the pilot's operation. We assumed *no interaction* or influence between the two categories of motion sensors. This assumption is quite plausible, since we consider the sensors to be at the periphery, and anatomically or physiologically, there is no known interaction between sensors at that level. Another assumption is that there is *no feedback* between blocks contained in the pilot's block diagram (solid lines). The absence of feedback implies some restrictive assumptions on how the pilot operates, and therefore is discussed. First of all, we eliminated the so called *efferent nerve* fibers (given as single dashed lines in Figure 4) that transmit information or commands from the Central Nervous System (CNS) to the peripheral sensory organs. These are known to influence the *afferent nerve* fibers' firing rate (the nerves leading from the peripheral sensory organs' output to the CNS). The influence of the efferents on the afferents' output is instantaneous, in many cases, and thus could have a significant role in the response characteristics of the sensory organ to external stimulation. But, the role of the efferent fibers in the vestibular system (or even the auditory system), is not understood, and "no significant effect" (more than a factor of five) of efferents on afferents' output has been reported in the case of the vestibular sensors [Dechesne80]. In conclusion, for now, we best **assume** for our design that this **effect does not exist.** Furthermore we **assume** that there are **no other pathways** from the CNS to influence the vestibular output. In summary, we assume that there is no feedback from our postulated orientation estimator and controller to the pilot's sensors.

Orientation estimator independence

A second assumption made (partially represented in Figure 4 by the double dashed lines that are ignored) is that either the orientation estimator is not influenced by the task, or that all the details of the task are known to it in advance. The latter is usually impossible, since the task here is the whole future desired orientations and trajectory of the pilot's airplane and possibly those of other moving objects (e.g. an enemy airplane, or the tanker in air refueling). Furthermore, the airplane's orientation and trajectory are also influenced by flight disturbances, which add to the uncertainty that has to be resolved by the pilot's orientation estimator. Thus the pilot's orientation estimator problem involves both noisy measurements (due to his sensory organs) and at best only partial a priori information about the overall system that he is controlling (namely the airplane and its environment). It is assumed here that the orientation estimator is independent in order to simplify the block diagram, and to emphasize the problem of constructing a specific model for the pilot's orientation estimator. This orientation estimator independence assumption is used during the discussion of possible solution approaches; but is not used in the final *simulator motion design problem* formulation—this is the key reason that the approach in this thesis is successful.

Other pilot sensors

Actually the "controller" has additional inputs—the pilots' "feel", through his hands, of the airplane responses. We consider this sensory input in the category of the other orientation sensors, although it is not strictly an orientation sensor, and moreover, it is partially inertial. In general, the pilot has other sensors that are relevant to the flight simulator problem which were not mentioned, since they seem to have a minor effect or they drop out in our idealized simulator motion design problem, as discussed later on [Martin80].

3.2. Pilot Information Flow and its Uncertainty

Separation of information kinds and their uncertainty

In the design of the washout system we are faced with motion-limitations that have to be disguised from the *simulator pilot.* After exploiting the physical laws involved, the main tool we have at our disposal is the pilot's limited ability to observe, sense, and interpret his environment. In order to take advantage of these limitations we assume an information structure. The first part of it is the postulated pilot block diagram discussed before and shown in Figure 4(a). Beyond that we make further assumptions on two aspects of the information flow: (a) the different possible kinds of information, (b) the uncertainty associated with each kind of information. By "kinds of information", we mean the different types of orientation information that are available at a certain point in the pilot block diagram; i.e.

yaw rotation and pitch rotation can be considered separate kinds of information if they cannot be derived from each other. These "kinds of information" can be termed the different orientation features the pilot obtains at different points in the pilot block diagram.

For example, in mathematical terms, different kinds of information can be considered as the Karhunen-Loeve information-function expansion axes. The uncertainty associated with each kind of information is described by the Karhunen-Loeve random variable associated with each ortho-normal-expansion-function [Van-Trees68].

We first consider the information flow associated with each kind of information and then we describe the amount of uncertainty added to each kind of information as it propagates through the pilot's block diagram.

Pilot information flow

The small number of different sensors the pilot has is the first fact that reduces the kinds of information available to him. This is shown in Figure 4(b) as a reduced span of kinds of information at the output of the pilot's sensory organs.

The orientation estimator further reduces the information span due to two factors:

(i) The same kind of information is available from several sensors. For example, orientation information is available both from the vestibular organ and the visual system. The orientation estimator combines these two information sources into one orientation estimate on the basis of some a priori assumption about the kind of information gathered and the "quality" of each individual sensor. Behavioral support for this assumption comes from visual motion illusions and the sensory conflict theory that are discussed later on in subsection 4.3.

(ii) What is the orientation estimate for? That is to say, is it to be used as an input to the pilot's controller or, is it just the pilot "feel" of his current orientation. For these two alternatives the orientation estimator would not necessarily extract an overlapping set of kinds of information, and thus could make different assumptions during the estimation process of each one as shown by the flaps-down experiment (Chapter VII, Sections 3.3.2 and 4).

Next the controller further narrows down the span of kinds of information to only those required to control the airplane. From the foregoing it is clear that the span of kinds of information keeps narrowing down from the pilot's sensory inputs to his control outputs. In mathematical terms we say that each one of the blocks, in the pilot block diagram, is a noninvertible mapping of input to output.

Pilot information uncertainty flow

The kinds of information decrease along the pilot block diagram; but the information uncertainty, associated with each kind of information, generally increases, because of noise added along the way, as diagrammed in Figure 4(c). First added is the sensory noise; then, the estimator "computation noise"; and last, the controller "computation noise" and control "actuator noise", (because of the "noisy" actuation of the airplane controls by the pilot's limbs). It is **assumed** that the main source of uncertainty is because of the sensory noise, and that the "actuator noise" is reduced to an insignificant level due to the feedback to the pilot from the airplane controls (if the airplane "controls feel" is simulated properly and when the pilot is well trained). On the basis of this information structure we discuss in the next section the comparison of the *simulator pilot* flight to that of the *airplane pilot.*

4. Comparison of Airplane Flight to Simulator Flight

Motion generation is our problem

From here on, we assume that the *motion generation* is the only problem on hand, and all the rest of the parts of the flight simulator are perfect. Just having motion imperfection implies that only the inertial motion sensors of the *simulator pilot* may receive different motion inputs from those of the *airplane pilot.* Thus all the pilot's noninertial motion sensors are ignored.

Comparison metric

Under these conditions we can compare the *airplane pilot's* situation to that of the *simulator pilot's* at five corresponding points referenced A to E in Figure 5. A *comparison metric* is a comparison operator which gives a single number as a measure of the distance between the two signals compared, at every instance in

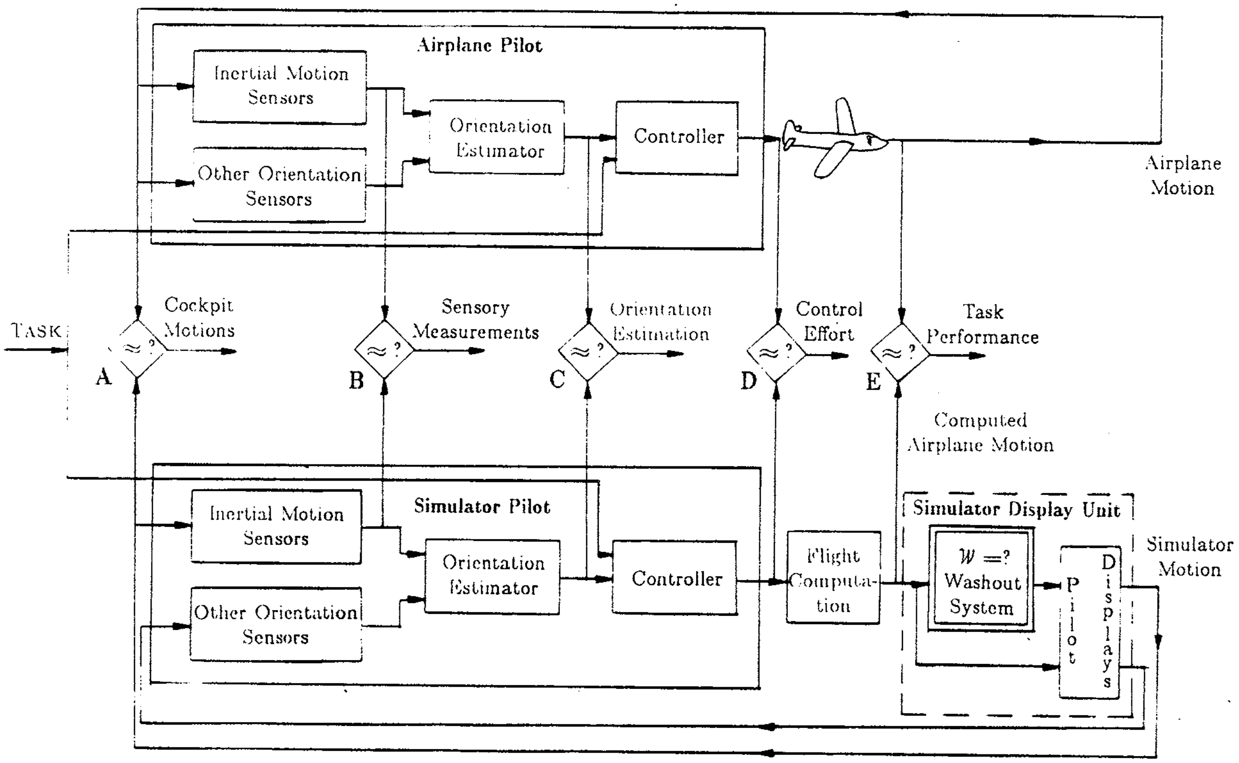


Figure 5. Comparison of *airplane pilot* to *simulator pilot* flight.

time (note the **assumption** of **single instance comparison**). The simplest comparison metric is the difference between the two signals. This is indeed our initial choice for a motion quality criteria in the washout design, but is further discussed and modified in chapter IV. The effect of a specific choice of a comparison metric on the washout design concomitant with the increased restrictions on the simulator motion has a major effect on the resulting *"optimal" flight simulator motion design*. The design results in this thesis seem to indicate that the simple difference comparison metric is adequate for the existing very-large motion simulators but could be quite questionable for smaller motion simulators.

4.1. Comparison of Cockpit Motions

The first, straightforward comparison point is the motion of the cockpits (A, Figure 5) of the two pilots. In an ideal case, these two motions would be equal and the simulation would be perfect. It seems plausible to try our best to achieve this perfection. Indeed this is the motion quality criterion in most washout system designs. This criterion is usually formulated as a requirement to minimize the difference between the two cockpits' three linear accelerations and three angular velocities. This formulation is only a partial comparison, which takes into account the fact that the vestibular organ roughly senses these six variables [Parrish73, Friedland73]. The drawback of a comparison at point A is that it can take only a limited account of the shortcomings of the human inertial motion sensory organs. It is quite plausible that these are not perfect, and taking advantage of their limitations (noise and information kinds) should enable us to get better motion quality for the same restrictions on the simulator motions.

4.2. Comparison of Sensory Measurements

At point B, the *sensory measurements'* outputs of the peripheral inertial motion sensors of the two pilots are compared. Obviously when the two pilots have identical *sensory measurements* they cannot detect any difference between the airplane and the flight simulator situations. This comparison may seem like an odd thing to do, since we cannot connect our equipment to the pilot's brain, and measure a signal corresponding to these sensory outputs. Instead we have to use models for these sensory organs, and compare the outputs of these models. An important limitation is the accuracy of the models we use, and with poor models we may obtain results that are *worse* than for the comparison of cockpit motions. We use a model of the vestibular system, which is known to be the most sensitive inertial motion sensor, and its output is chosen to represent the output of all the inertial motion sensors, as discussed later in subsection 7. Another reason for this choice is that the other inertial motion sensors are distributed and their models would have to be very complex, or oversimplified in order to be used. A vestibular model was used for the first time by [Hosman79] as an optimization criterion for the design of a washout system.

4.3. Comparison of Orientation Estimate

Here too we can take advantage of the limitations of the pilot's orientation estimator (less information kinds and increased uncertainty=noise), and compare the two pilots' *orientation estimates* (C in Figure 5). This should make it possible to use more restricted motion in the simulation. In some cases, on the basis of this comparison, we may conclude that even with *zero motion* the simulator pilot has an orientation sensation which is indistinguishable from that of the airplane pilot (based on the *visual scene* generated by the flight simulator).

Examples of visual orientation and motion illusions

Actually many people experience this strong visually induced illusion of motion, which is used in amusement parks, cinerama movies, and other movies projected on a very large screen, such as "To Fly" presented in the Air and Space Museum in Washington, D.C.. The common feature in all these visually induced motion and orientation sensations is the very wide angle visual presentation.

Let us go on an amusement park tour of an abandoned silver mine and follow the illusions presented on the way. First you enter a dark room tilted up by, say, 3 degrees. After everybody enters the room, the outside door is closed and after a few minutes the lights are turned on. Now you have a very strange feeling that you are not standing upright, but tilted relative to the vertical, because you assume that the visual scene in the room gives you the vertical orientation reference that you would have in a normal room and thus you are deceived: you perceive the wrong direction of the gravity.

Now the tour proceeds down to the deep mine itself. Here there is an excellent chance to give you the sensation of linear motion, during the long ride down in the elevator. You step into the elevator and after the door is closed, it starts to shake a little and some dim strips of light go up through the cracks of the elevator, so that when the elevator stops, you end up deep down inside the mine and leave the elevator from a door on its other side, so you will not discover that you were deceived, and did not move down even one inch.

We skip the next mine horrors and go to a place where you have to cross a short bridge. This bridge has side rails and around it there is a well-illuminated drum which has spiral red and white stripes painted on its inside. This drum rotates just around you and gives you the sensation that you are tumbling around, and you may fall off the bridge, so hold on to the hand rails, and cross the bridge in a hurry. In the wide screen movie "To Fly", a scene is projected as seen by a pilot flying an airplane in a valley between two mountains. The viewer has the sensation of moving with the airplane as if he were the airplane pilot.

These examples show that indeed a visual scene can give the illusion of an unreal orientation and sensation of motion, that could be used in a flight simulator to replace actual motion (which is hard to generate). Moreover, it is known that the sensation of motion can be generated (to a much lesser extent) using auditory cues as well. These examples seem to indicate that we could use the orientation estimate as our comparison point and thus take advantage of further limitations of the human orientation perception mechanism, so as to give the simulation pilot the feeling of flying, even with less motion or with *no motion* at all. This is the case in fixed base flight simulators, some of which are so good that in some cases people have come to believe that motion in a flight simulator is not necessary at all.

Orientation estimate training

The approach of orientation estimate matching is good for most flight training situations where we want to train the pilot's ability to best accomplish a given task; but where we do not care how the pilot's orientation estimate was obtained. In terms of the pilot's functional block diagram, Figure 4, this is training of the "controller" only. We should however recognize the importance of training of the orientation estimator since humans, in everyday life, navigate and orient themselves on a surface, having only three degrees-of-freedom, while a pilot in an airplane has all six degrees of freedom. According to two army pilot instructors, the skill of orientation in six degrees of freedom, is the first skill to be taught to a pilot cadet, and is learned and improved continuously throughout the two years training of military pilots.

Sensory conflict

If we want to train the pilot's orientation estimator then we should not use the orientation estimate comparison point. Using this comparison point may train the pilot to use an inefficient, bad, estimation process. Since there is no real motion in a purely visual simulator, the simulator pilot's orientation estimator is confronted with conflicting input information from his peripheral sensory organs; on the one hand, he concludes from his visual and auditory senses that he is moving, but his inertial motion sensors "tell" him that he is not moving at all. The pilot's orientation estimator has to resolve this confusing situation and output to the pilot's controller the best estimate of what is happening. This has to be resolved somehow, since a human is an animal that stands only on two legs and cannot, or almost cannot, function properly with an improper orientation estimate of "down" (he will usually fall down). If the sensory information conflict is not too large, then in many cases a person is not even aware of the conflict, and usually is not even aware of his need to obtain an orientation; on the other hand, if the conflict is larger, then the reaction is *motion sickness*, according to the sensory conflict theory [Oman80]. The first symptoms of this sickness vary among individuals and range from a headache, a hot feeling and facial pallor; surprisingly these symptoms do not appear to indicate disorientation. Later symptoms of motion sickness may be enhancement of the above and, in addition, nausea, dizziness and finally vomiting. The dizziness would seem to have an intuitive correlation with the disorientation we would expect in a strong-conflict situation. In high-quality purely visual simulators, such a conflict situation may arise, and the simulator pilot may suffer from *motion sickness* which is termed, under these conditions, *simulator sickness*. This sickness may require grounding the pilot for several hours after the simulator flight and sometimes for longer than that. Many times this sickness is brought up as a good enough reason to include real motion in a flight simulation even when the *"visually induced"* motion sensation is very good [Puig79]. It is noteworthy that in space about 50 percent of the astronauts suffer from similar symptoms and their *motion sickness* is termed *space sickness*, and this sickness is so bothersome that it is one of the main reasons for NASA and the U.S.S.R. to support vestibular research on earth and in space.

Using the orientation estimate comparison point

In cases where we assume that we do not need to teach the pilot's orientation estimator, it seems like a good idea to use the orientation estimate comparison point as the basis for a design of a washout system. As before, we have to use a model for the orientation estimator, since we cannot connect to the pilots' brain. A further complication is, that there is no uniquely defined and known brain signal that could be considered as the orientation estimate, and thus no physiological measurements for such a model are available. Furthermore the orientation estimator is task dependent as discussed before, which makes it a very complicated system, hard to model and, one whose parameters are difficult to measure, and moreover hard to use for a washout system design. These are the most likely reasons why this very attractive idea (suggested by [Oman71]) was not used for a washout system design.

4.4. Comparison of Control Effort

The fourth comparison point (D Figure 5) is the two pilots' whole model outputs, which is a comparison between the pilots' controls applied to the airplane and those applied to the simulator. This comparison point is usually referred to as the control effort. In this case we take into account further restrictions, so that a more limited set of information kinds is used. In the case of the *simulator pilot* this is an actual signal that can be recorded, but still we need a model to represent our imaginary airplane pilot. One immediate problem with this model is that in reality it is not likely that even if we use the same pilot in a real airplane and have him fly the same mission (task), he still would use exactly the same controls; or even in an ideal simulator in a repeat of the same flight. Also, if we want to use this comparison point in the design of a motion washout system then we need to use a model for the simulator pilot as well. Such models for the pilot control strategy such as the optimal-control and the cross-over models are well known and offer some useful results for a sufficiently restricted set of tasks (since these models are essentially curve fittings). Furthermore, using this comparison point, we have to take into account another uncertainty input to the pilots' control outputs, which would represent the differences in the controls used, even by the same pilot flying the same task, under the same "conditions". This tradeoff of the additional uncertainty and, on the other hand, a pilot controller model that restricts the possible pilot control outputs is probably a good one to make. However the real question is whether using this comparison point is a sensible thing to do at all. We tend to say that it is not, since this comparison point is further away from the pilot's sensory motion input or even his orientation sensation and thus, it seems plausible that having the same control efforts would be satisfactory in one case and unsatisfactory in another, as judged by the simulator pilot. Looking at this another way, if the pilot is happy with the simulation quality, then having the same control effort could be used to confirm the good simulation, but we cannot use this argument in the reverse direction as would be necessary for the design of the motion washout system.

4.5. Comparison of Pilots Performance

The last comparison point we can look at is the task performance, which is the comparison of the degree to which the two pilots achieve the given task. It is known that some pilots can achieve the same task performance with or without motion [Puig78]. So we cannot use this comparison point for our motion washout system design. Actually the pilots that can perform the task independent of the simulator motion are well trained test pilots who are very familiar with the flight simulator in question and flying "the usual stable" airplanes. Furthermore it is usually claimed that the control effort is very different in these two cases, which further stresses the point that the task performance is not a good comparison point. This very fact was used before, in section 2, to simplify the washout system design by assuming that the anticipated airplane motion is independent of the washout design.

4.6. Experimental Use of Control Effort and Performance Comparison

We can nevertheless use the task performance combined with the control effort to monitor how well our simulator pilot compares to the imaginary airplane pilot. This corresponds to a statement of how well the flight simulator resembles the real airplane. These last two comparison points are used later in the experimental evaluation of the designed optimal washout.

5. Cost Function

A scalar cost function

The objective of this thesis is to find the best washout system, which is the best transformation, $\mathcal{W}$, from the computed airplane motions to the control commands for the simulator motion base. To find an optimal (best) washout we must have an objective comparison criterion that can let us decide which is the best washout design. This can be a clear-cut decision only if we reduce all the important quantities to one scalar number that represents the design quality, and thus makes the tradeoffs between these quantities explicit. The scalar number is called the *cost*, J, of a specific washout design and the best design has the minimum cost.

Cost parts

Why should there exist an optimal washout, and what are the tradeoffs involved? The whole problem arises because the flight simulator has only restricted motion; it simply cannot move as much as the real airplane. Thus we have to specify the tradeoff between the restricted motion and the vestibular error due to it. In our formulation, the cost has two additive parts, J_ℓ and J_e. The cost function part J_ℓ, due to the restricted motion, and can be formulated by penalizing excess travels, velocities and accelerations as implied from the flight simulator motion base characteristics. Formulating the cost function part J_e, due to the vestibular error is more involved and is a best guess of a reasonable measure to define the "distance" between the vestibular models' outputs of the airplane and simulator pilots. To simplify this "distance" definition, we divide it into two disjoint contributions:

(i) The basic sensitivity, the *perceptual threshold* (J.N.D.—Just Noticeable Difference) of the vestibular system, which depends on the input level at least. This sensitivity function is accounted for by normalizing the vestibular output according to the perceptual threshold and using what we call "threshold units" for the vestibular output. After this normalization it is assumed that the "distance" at each time instance, t, is the difference between the two pilots' normalized vestibular outputs, called the *vestibular error*, and in the general case it is called the *sensory measurement error*. The "threshold units" are further discussed in subsection 8.

(ii) The *perceptual error rating*—which is the relative perceptual scaling of the *vestibular errors* and the way the errors are combined over a time period T to rate the simulator motion quality, as it is "felt" by the simulator pilot. Is a meaningful score the maximum error during some period T, or perhaps the R.M.S. (Root Mean Square) would be the important measure of pilot's "tolerance" to *vestibular errors*? The R.M.S. score is adopted here since it is the one for which we know how to solve, and furthermore, the general question is not answered yet—further research is needed.

Simulator motion cost function—remarks

The cost J_ℓ due to the restricted simulator motion capability has to include a cost on exceeding the simulator motion base capabilities, but has also to include a cost for off-center travel. The cost added due to off-center motion is due to

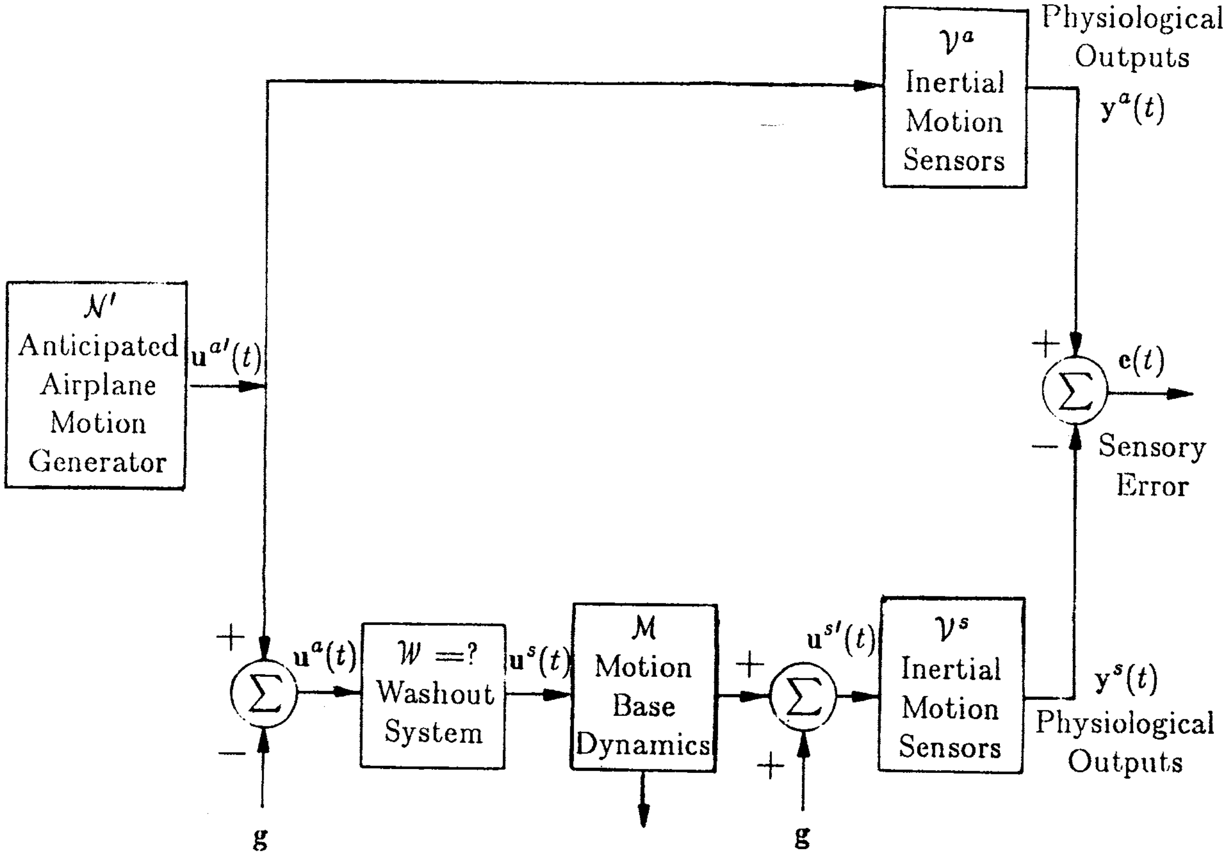


Figure 6. Flight simulator motion design problem.

the uncertainty of the future motion of the simulated airplane—that is under the control of the unpredictable human simulator pilot. The practicality of generating a cost function that allows known optimization algorithms to be used leads to the use of a quadratic cost function (quadratic in the time variables of interest). This includes a cost on integrals of the simulator cab linear displacements and angles in order to provide the required centering effect. The key to the simulator cockpit motion restrictions is the simulator motion base operation method. Although several common methods in use were described in chapter I we specifically solve only for the simplest case of a cascaded motion base system. For more complex situations such as a hexapod motion-base see Subsection 9.5 and the use of PLQ described in Chapter IV, Section 2.

6. Flight Simulator Motion Design Problem

A block diagram that represents our approach to the solution of the flight simulator motion design problem is given in Figure 6. This optimal washout system design problem can be stated as follows:

Flight Simulator Motion Design Problem

> Given a dynamic model of the *anticipated airplane motion, simulator motion-base* and the pilot's *internal motion sensors*, find an optimal operator $\mathcal{W}$ (WASHOUT SYSTEM) which generates $\mathbf{u}^s(t)$, such that we obtain the best flight simulator *motion quality* while not overriding the flight simulator motion base limits.

We choose to judge the flight simulator *motion quality* by looking at the *sensory error* which is the difference between the outputs of the airplane pilot's and simulator pilot's inertial motion sensors (comparison point B in Figure 5). In section 7, this sensory error will be approximated by the vestibular error since the vestibular organ is considered the most sensitive inertial motion sensor. Later on, in Chapter IV, the comparison operator will be modified so that it is more appealing from an information point of view.

Why use sensory comparison?

The sensory comparison is used to judge the *motion quality* since, at the output of the human sensory organs, we take advantage of most of the *information uncertainty* incorporated (an assumption) due to "sensor noise", while not reducing the *number of kinds of information available* to the pilot's brain, as discussed before in subsection 3.2 and shown in Figure 4.

Experimental validation of sensory noise dominance

The assumption that the information uncertainty is mainly due to the sensory noise is a very plausible one, which can be supported by some psychophysical motion detection threshold measurements and the comparison of these measurements to physiological afferent nerve "noise" as made by [Hosman78]. Hosman measured the psychophysical threshold of pilots to linear acceleration in the heave axis and to angular acceleration in the pitch and roll axes. These thresholds were measured as a function of the frequency of a sinusoidal motion stimulus and defined as the minimum stimulus the pilots could detect (the exact probability of 'detection' and 'false alarm' is not specified for these experiments). The pilot's task was to detect, in the dark, on which of the three axes of the simulator (heave, pitch or roll) motion was present and if it were what test frequency did it have. From these measurements, Hosman calculated the frequency response of the vestibular organ (ignoring other human inertial motion sensors) that matched very well with those obtained physiologically at the afferent nerve fiber level for animals. This match indicates that the underlying assumption is correct—namely that the dominant motion detection uncertainty is due to the "sensor noise" and not due to limitations in the brain's detection of the sensory signals. Furthermore, this shows that the dynamic limitations (at least in the measured frequency range) are due to the vestibular organ. Beyond that, Hosman compared the psychophysical thresholds he measured to the optimal detection limitation due to all the regular firing vestibular afferent nerves. This was off by less than a factor of two (see Table 1). (The detection is assumed to be done by summing all of the n afferent nerves and thus reducing the "noise" by a factor of $\sqrt{n}$ compared to a single nerve.) For the detailed assumptions under which this process is optimal, see [Van-Trees68]. From this comparison, we see that the human brain acts very much like an optimal estimator in this task and that the main limitations of motion detection in the dark are due to the vestibular sensor limitation.

	Sensory epithelium	no. of units	σ i.p.s.	σ_n i.p.s.	sensitivity	thresholds *)	T.S.N.
Regular units	crista	1500	5	0.13	2	0.03-0.065	0.46-1
	macula	2100	2.5	0.055	3.4	0.04-0.085	2.47-5.25
All units	crista	3000	16.7	0.30	2	0.03-0.065	0.2 -0.43
	macula	3000	7.6	0.14	3.2	0.04-0.085	0.97-2.06

*) thresholds in $^{\circ}/\text{sec}^2$ (crista) and m/sec^2 (macula).

Table 1. Comparison of psychophysical thresholds to vestibular afferent noise [Hosman78].

Notes:

(i) This comparison is correct as stated under the assumption that the psychophysical thresholds are obtained with 69% probability of detection and 31% probability of false alarm. This is not stated as the condition in the referenced report, but is assumed to be a fair approximation of the situation.

(ii) The regular hair cells have an afferent nerve firing rate that can be approximated by a Gaussian distribution, while the irregular cells can be approximated by a Poisson distribution. Furthermore, there are hair cells that fall within the entire range of the firing rate distribution between these two types of hair cells.

(iii) T.S.N. is defined as the threshold signal to noise ratio:

$$\text{T.S.N.} = \frac{\text{threshold} \times \text{sensitivity}}{\sigma_n}$$

and is the comparison measure of the psychophysical threshold and the physiological afferent nerve noise. A numerical value of one indicates perfect matching. σ_n is the standard deviation of n afferent fibers where each has a standard deviation σ, $\sigma_n = \frac{\sigma}{\sqrt{n}}$.

(iv) Crista = Semicircular-canals, Macula = Otoliths.

There are two other reasons to choose the sensory comparison point over the orientation estimation one . The orientation estimator eliminates many different kinds of information in order to obtain the required orientation estimate. This involves a noninvertible mapping from a high dimensional space of all the kinds of information available to the brain from its sensory organs to a lower dimensional space of an orientation estimate. This orientation estimate space may not be unique and may differ according to its purpose: the pilot's orientation feeling or the pilot's orientation estimate used to further compute the necessary controls to the airplane, or maybe another feeling, concerning the consistency of the input information from his sensory organ which causes the pilot to feel motion sick in conflict situations[1]. This brings up the question of which orientation estimate spaces to examine or whether we should look at all of them. However, looking at all the possible ones would most likely bring us right back to the sensory measurement comparison, since we would have to consider all the kinds of information available from the pilot's sensors.

[1] Two different spaces where shown to exist for: "pilot detection" and "first control" in the flaps-down experiment (Chapter VII, Sections 3.3.2, 4)

Taking the other stand, there is an important case where using the orientation estimate comparison could be very useful. This is when the objective of the flight simulator flight is to train the pilot's CONTROLLER (Figure 4)—namely the pilot's control strategy which also includes his flight planning. In this case, we assume that the only thing we are interested in is that the pilot arrive at the most realistic orientation estimate (due to the simulator motion) and use that to train his control strategy. Now, the modeling of the pilot's orientation estimate is a severe problem. This task is very complex and has only been touched upon so far [Huang79] [Borah82] since it involves considerations of at least inputs from the visual system in addition to those from the inertial motion sensors.

Optimization criteria

The quadratic cost function optimization criteria J (1), is composed of two parts. One J_ℓ—due to the motion limitations of the flight simulator motion base. Two J_e—due to the vestibular error introduced due to the motion limitation

$$J = J_e + \rho J_\ell \tag{1}$$

where

$$J_e = \mathfrak{E}\{\int_{-\infty}^{\infty} \mathbf{e}^T(t) Q \mathbf{e}(t)\, dt\} \tag{2}$$

$$J_\ell = \mathfrak{E}\{\int_{-\infty}^{\infty} \left(\mathbf{u}^{sT}(t)\, ,\ \mathbf{u}^{\ell T}(t)\right) R \begin{pmatrix} \mathbf{u}^s(t) \\ \mathbf{u}^\ell(t) \end{pmatrix} dt\} \tag{3}$$

and ρ is the relative weight design parameter.

The motion limitations are on the commands to the simulator motion base $\mathbf{u}^s(t)$, and on the other dynamically related variables grouped in $\mathbf{u}^\ell(t)$. These usually include at least the motion base travel in each degree-of-freedom and their integrals; the latter used to try to center the simulator cab at all times (leading to integral control) [Sturgeon81]. In a cascaded motion base system, the travel is given as a double integral of the motion base acceleration. In other motion systems, like the hexapod motion base, the limitation of the motion base is not on its travel but rather on the length of the hexapod legs, which are a non-orthogonal axes system which has a nonlinear transformation to the simulator six degrees-of-freedom. This complicates the washout design problem and thus these kinds of motion limitations are not considered here but are considered somewhat in Chapter IV and in references [Sturgeon81] and [Konst79].

In Figure 6, we see a subtraction and addition of one **g** before and after the sought optimal washout system. This operation does not restrict the generality of the solution for the optimal washout system; but is introduced in order to account for the existence of the earth gravity field which the washout system should not "washout". This formulation makes the existence of the one **g** field explicit and thus enables us to use a quadratic cost.

7. Can the Vestibular Organ Represent All the Inertial Motion Sensors?

We choose to judge the flight simulator *motion quality* by the difference between the *airplane* and *simulator* pilots' inertial motion sensory systems outputs (comparison point B in Figure 5). We use the vestibular organ's response to inertial motion to represent all the human inertial motion sensors since the vestibular organ is considered the most sensitive of them and seems possible to approximate by a relatively simple *lumped* model that is widely accepted. The human inertial motion sensors can be divided into two groups, rotation and linear motion sensors. In the vestibular organ, the semicircular canals are the primary rotation sensors and the otoliths are the sensors for linear motion. We first discuss the vestibular rotation sensors and then the linear ones.

Rotation sensors—the Semicircular canals

It is accepted that the semicircular canals are the primary angular rotation detectors in the dark, when precautions are taken to reduce or mask auditory, vibratory and other cues to movement [Guedry74]. Some evidence for this assumption was found as early as 1824 by Flourens and reviewed by Peters [Peters69]. Flourens sectioned the semicircular canals of pigeons and rabbits and associated disturbances of head and body motions in planes that corresponded with those of the injured canals. Another type of evidence for the decisive role of the semicircular canals in rotation detection, can be concluded from the misnamed Coriolis illusion; that can be explained by the dynamics of the semicircular canal alone(the Coriolis illusion involves a sensation of body rotation , and an apparent motion of objects in the visual field, which is caused by tilting the head about one axis during rotation about another axis). The evidence quoted here is by no means complete, but gives some illustrative arguments.

Linear sensors—the Otoliths

The assumption that the otoliths are the primary detectors of linear acceleration is more involved; nevertheless it is supported by several studies of motion thresholds (reviewed [Guedry74]). The main study on linear motion was done by Walsh, who compared estimates of motion thresholds (loosely defined, as the lowest level of motion a person can detect) for linear acceleration in the dark for several cases. First of all, he found that thresholds were only slightly elevated when normal subjects were immersed in water (a condition were most of the known non-vestibular inertial motion sensors are excluded). Secondly, these slightly elevated thresholds were also measured in individuals with high spinal lesions—a condition similar to a normal subject immersed in water. Finally, these two results can be contrasted with measurements obtained from individuals with complete bilateral labyrinthine loss (no vestibular function) who were tested without liquid immersion; mean thresholds were elevated by a factor of four. Presumably, if liquid immersion had been used with these labyrinthine-defective individuals, thresholds would have been elevated even more. Thus, over all, it seems that the vestibular organ has the major role in sensing motion in the dark and can be considered to represent all the inertial motion sensors in the design of the washout system done in this thesis.

Vestibular model representation of all the pilot's inertial motion sensors

According to the above discussion we use for system $\mathcal{V}^a$ and $\mathcal{V}^s$ (Figure 6) models only of the pilot's vestibular organ which will represent all the pilot's inertial motion sensors. The inputs to these vestibular models are the angular rates and the linear accelerations to which the pilot is exposed, and the models outputs are the firing rates of the afferent nerves transmitting the vestibular sensors sensing to the brain. These models are discussed in the next subsection.

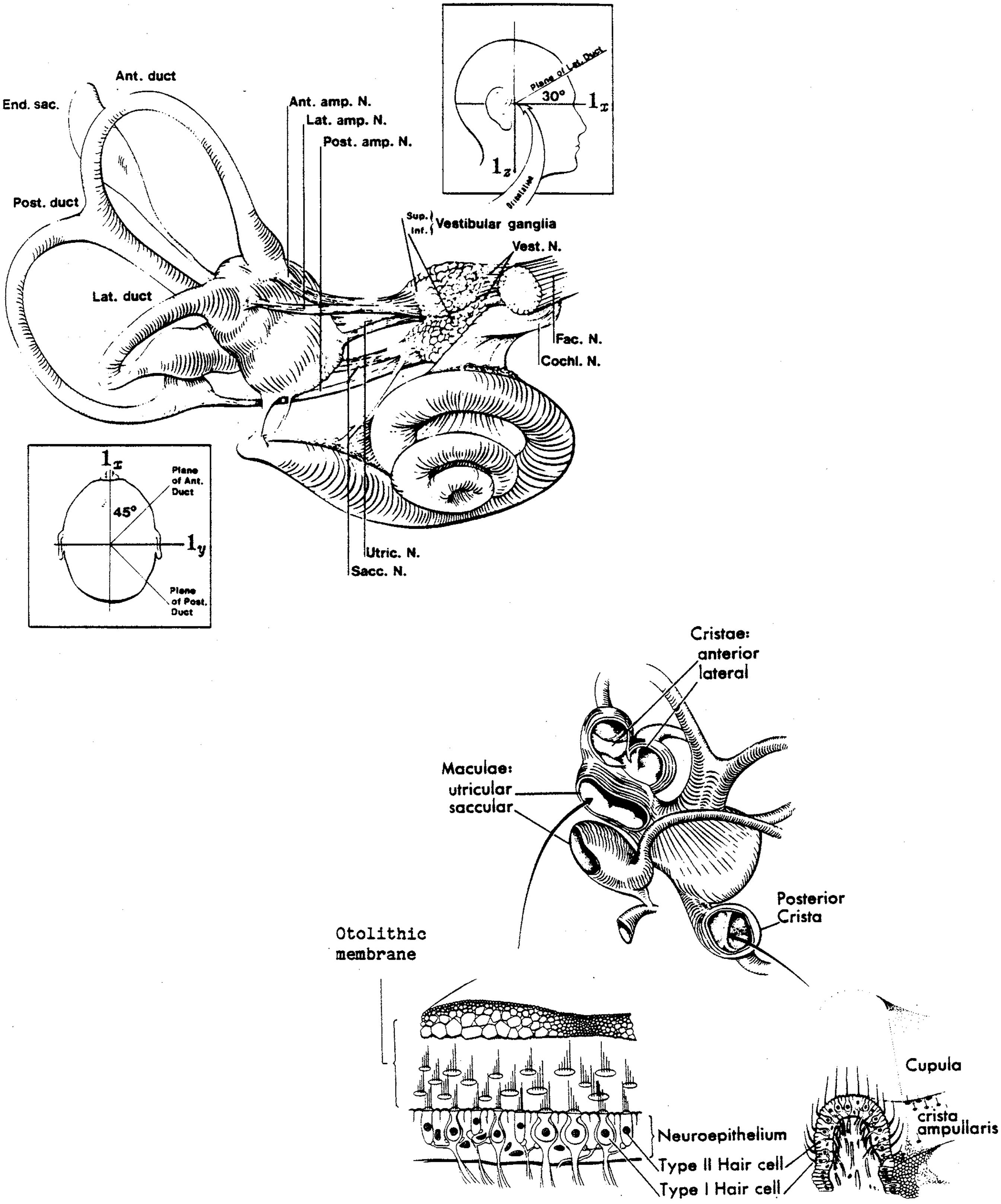


Figure 7. Human vestibular system [Correia78].

8. Vestibular Organ Modeling

8.1. Vestibular Organ Description

Humans have two vestibular organs that are situated in the non-auditory portion of the inner ear. They serve as transducers in the biological inertial guidance system. Each vestibular organ consists of an interconnected set of fluid filled membranous sacs and ducts as shown in Figure 7.

Three mutually orthogonal semicircular canals, thin ducts which open into the "utricular" sac, function as integrating angular accelerometers. During head rotation, fluid in the torus formed by each canal duct and the common portion of the utricular sac takes on a velocity which is proportional to the angular velocity which is proportional to the angular acceleration of the head, due to the viscosity of the fluid and the small diameter of the canal duct. The resulting flow displacement in each duct is coupled to mechanically sensitive hair cells (similar to those found in the hearing organ) located on the "crista" (Figure 7) in an expanded portion of the canal duct, the ampulla. The lumen of the ampulla is occluded by a gelatinous, transparent diaphragm, the cupula, which is in direct contact with the hair cells. During head movement, the hair cells encode a message corresponding approximately to head angular velocity in the firing frequency of the 8th cranial nerve fibers going to the brain.

Whereas the semicircular canals are well suited for detecting angular motion (and are normally insensitive to orientation of the head with respect to gravity), the sensing of linear acceleration and gravity is performed in two other specialized organs, the utricular and saccular otoliths. Each otolith (literally: "earstone") organ is formed by a specialized region of the inside wall of the membranous labyrinth, and is made up of several thousand mechanoreceptive hair cells, covered by a layer of finely grained calcite crystals. These crystals ("otoconia") are bound together and to the underlying mechanoreceptive wall by an elastic membrane. Seen through a microscope, the otoconia appear as a patch of white sand lying on the membranous wall. Since the otoconia have a density greater than that of the fluid which fills the utricular and saccular chambers, the otoconial layer in each of the two organs serves as a seismic mass in a biological linear accelerometer: When the otolithic membrane slides "downhill", the magnitude and direction of the otolithic membrane shearing deformation detected by the hair cells determines the distribution of neural activity across a large number of $8^{\underline{th}}$ cranial nerve neurons. The utricular otolith is located parallel to the floor of the utricular sac, and is thus sensitive to acceleration and gravity components roughly in a horizontal plane with respect to the head, whereas the saccular otolith, due to its orientation, responds to components in the head plane symmetry.

Semicircular canal and otolith organ information travels in the $8^{\underline{th}}$ nerve to relay neurons in the brain stem and cerebellum, where it is now known to combine with other sensory neural inputs which relate to body motion—particularly from the visual system. It has become clear that these brainstem and cerebellar structures play a critical role in body movement control, postural regulation, gaze stabilization, and very probably also in spatial orientation perception (i.e. "Which way am I turning? Which way is down?"[Young82]) [Oman82].

8.2. Assumptions in the Vestibular Model Used

The detailed vestibular model used in the washout system design is given in chapter V. Let us outline the assumptions used in the derivation of those dynamic models:

1. Cyclopean model used, i.e. only one lumped model located at the center of the head is used to represent both vestibular organs. By this assumption we exclude sensing of rotation by the otoliths due to centripetal forces which can be up to $4\,g$ at maximum head rotation speeds of $1500\,deg/sec$ [Ish-Shalom79].
2. Single output lumped model for each of the three semicircular canals and otoliths. In reality there are at lest two types of responses, of the type I hair cells and of type II hair cells.
3. Orthogonal semicircular canals, i.e. most sensitive excitation axes assumed $\mathbf{1}_x$, $\mathbf{1}_y$ and $\mathbf{1}_z$. In reality the first two axes are different. Also, the axes are only approximately orthogonal (Figure 7).
4. Three orthogonal otoliths with most sensitive excitation axes $\mathbf{1}_x$, $\mathbf{1}_y$ and $\mathbf{1}_z$. In reality there are two orthogonal planes (approximately planar) with hair cells sensitive in all directions (Figure 7).
5. Finite dimension linear models are used. Linear—it is not linear since it is known that the organ have a finite saturation; this nonlinear effect is

further discussed in chapter IV section IV.3. Some models found in the literature use infinite dimension models, furthermore some preliminary experiments done by the author measuring vestibular tuning curves, suggest a similar result.

6. In the next section we use identical models for the three rotation axes (semicircular canals) and for the three linear axes (otoliths). This assumption is clearly incorrect since at least the gain factor should be different due to different thresholds in the three axes.
7. The otolith model used is based on psychophysical experiments of detection of linear acceleration; its dynamics do not fit known mechanical models of the otolith organ which have a much higher frequency response [Ormsby74].
8. Semicircular canals are insensitive to linear acceleration. This is known to be an over simplification.

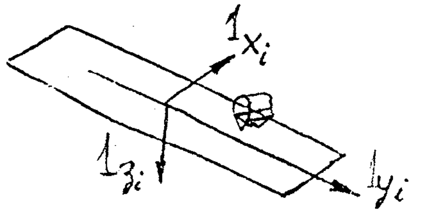
9. Axis Systems

The environment of a flight simulator requires a description of the location and direction of several objects in physical space. These can be described by six orthogonal components—three describing the object's position in space and three its direction. In the solution process, we encounter several coordinate systems, which are associated with the objects in our problem. The airplane trajectory computation usually involves two coordinate systems: the airplane body axes and the wind axes [Etkin72] or Euler parameters [Ramnath80]. The simulator motion base has its own coordinates which express its limitations, possibly in another set of coordinates (e.g. in the hexapod motion base system). We also have the two vestibular systems stimulating coordinates of the *airplane pilot* and the *simulator pilot*. The importance of all these coordinate systems is that although the equations representing each object by itself in its "natural" coordinate system may be simple and linear—when considering the overall system, in one common coordinate system, the set of simple equations becomes more complex and nonlinear.

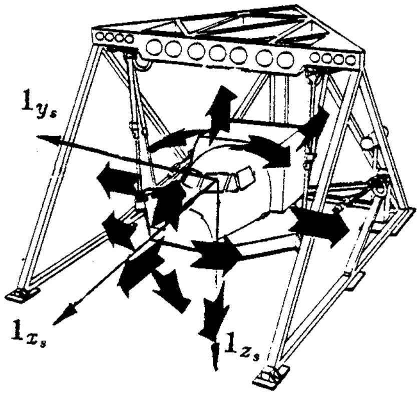
9.1. Axis Systems Involved

In formulating the simulator washout system design problem we consider the following four coordinate systems:

1. Inertial axes—coincide with the simulator's motion base translation axes $\mathbf{1}_{x_i}$, $\mathbf{1}_{y_i}$, $\mathbf{1}_{z_i}$. Its origin is at the center of each translation axis.

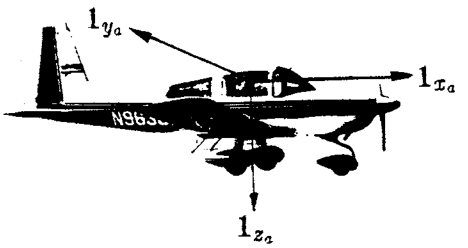


2. Simulator body axes—coincide with the simulator's cab body axes (fixed with reference to the cab) $\mathbf{1}_{x_s}$, $\mathbf{1}_{y_s}$, $\mathbf{1}_{z_s}$. Its origin is at the average center of the pilot's head. It has Euler angles ψ^s, θ^s, ϕ^s with reference to the inertial axes and the rotations are around $\mathbf{1}_{x_s}$, $\mathbf{1}_{y_s}$, $\mathbf{1}_{z_s}$ in this order. Collectively ψ^s, θ^s, and ϕ^s are referred to as:



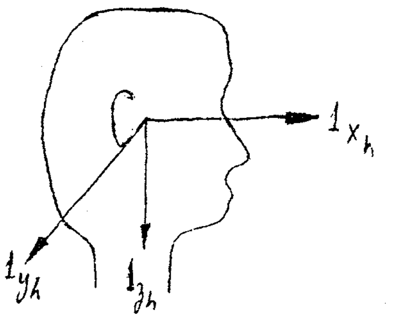
$$\lambda^s = \begin{pmatrix} \phi^s \\ \theta^s \\ \psi^s \end{pmatrix}. \tag{4}$$

3. Airplane body axes—coincide with the airplane body axes system $\mathbf{1}_{x_a}$, $\mathbf{1}_{y_a}$, $\mathbf{1}_{z_a}$. Its origin is at the average center of the pilot's head. It has Euler angles ψ^a, θ^a, ϕ^a with reference to the inertial axes and the rotations are around $\mathbf{1}_{x_a}$, $\mathbf{1}_{y_a}$, $\mathbf{1}_{z_a}$ in this order. Collectively ψ^a, θ^a, ϕ^a are referred to as:



$$\lambda^a = \begin{pmatrix} \psi^a \\ \theta^a \\ \phi^a \end{pmatrix}. \tag{5}$$

4. Pilot head axes—coincide with a coordinate system fixed to the pilot's head. Its origin is at the center of the pilot's head and its linear components $\mathbf{1}_{x_h}$, $\mathbf{1}_{y_h}$, $\mathbf{1}_{z_h}$ are measured from the average center of the pilot's head, which is the origin of the cab body axes for the simulator and the origin of the airplane body axes for the airplane pilot. It has Euler angles ψ^h, θ^h, ϕ^h with reference to the corresponding body axes, where the rotations are defined around $\mathbf{1}_{x_h}$, $\mathbf{1}_{y_h}$, $\mathbf{1}_{z_h}$ in this order. Collectively ψ^h, θ^h,



ϕ^h are referred to as:

$$\lambda^h = \begin{pmatrix} \psi^h \\ \theta^h \\ \psi^h \end{pmatrix}. \tag{6}$$

Since both the simulator pilot and the airplane pilot are assumed to behave exactly the same, $\mathbf{1}_{x_h}$, $\mathbf{1}_{y_h}$, $\mathbf{1}_{z_h}$ and λ^h have the same values for both pilots and are thus referenced by one set of symbols.

It is assumed that the simulator dynamics and limitations can be simply represented in the *inertial axes* and in the *simulator body axes.* This assumption is valid for cascaded motion systems, but skips several difficulties encountered when using a hexapod motion-base system, for example, where the limitations are on the leg extensions, which are in a completely different coordinate system.

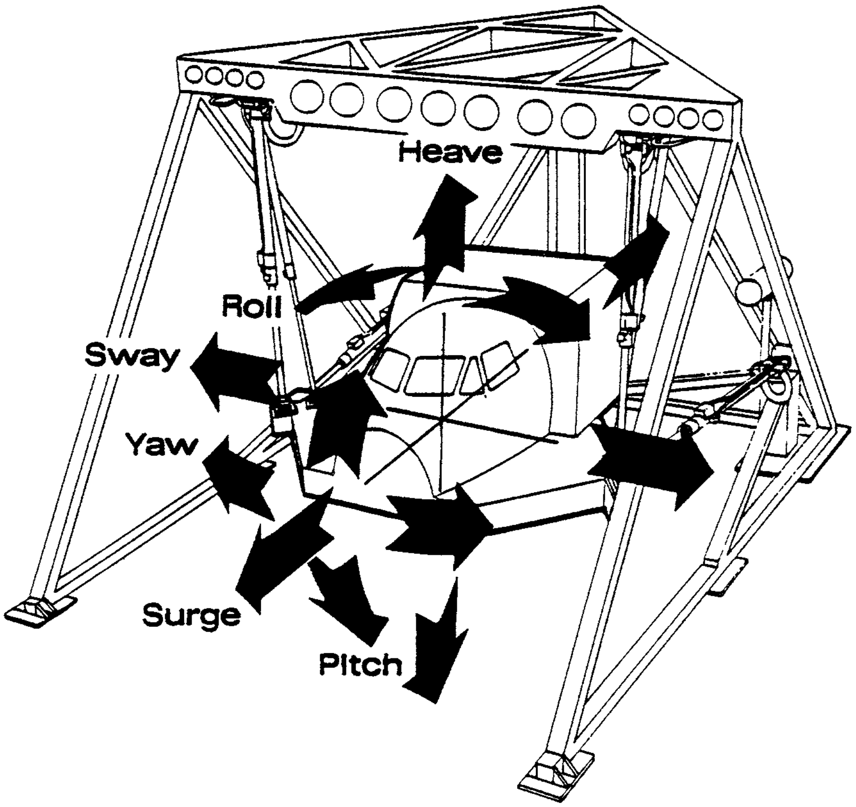


Figure 8. The six degrees-of-freedom of a flight simulator (A Redifon suspended six dof motion-base [Martin80]).

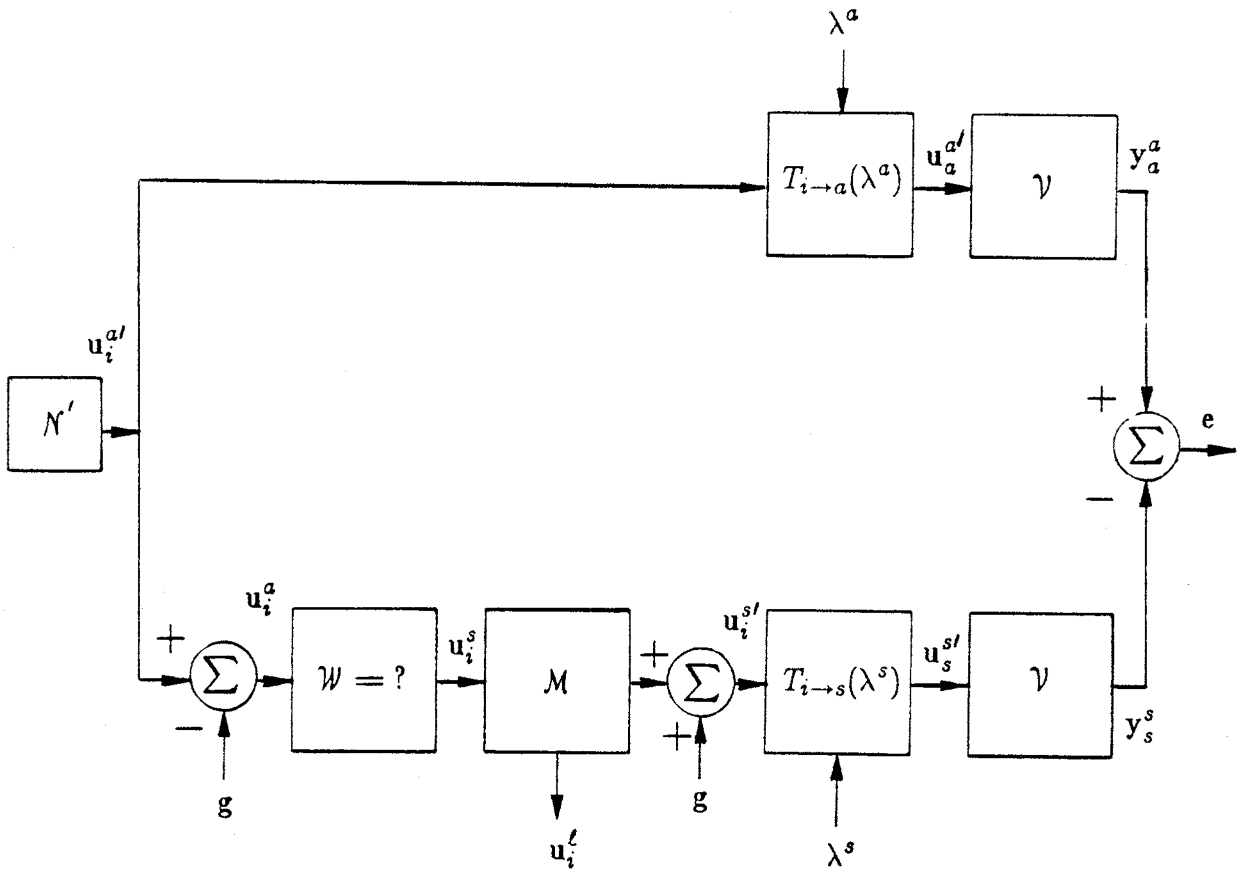


Figure 9. The axes transformations appearing in the flight simulator motion design problem.

9.2. Notation

The terminology for the six degrees of freedom of a flight simulator is described in Figure 8. Further notation conventions used in conjunction with the axes systems are given below.

(i) The axes systems in which the vector $\mathbf{x}$ is given is denoted by the subscript of that variable:

1. $\mathbf{x}_i$ for $\mathbf{x}$ in inertial axes
2. $\mathbf{x}_s$ for $\mathbf{x}$ in simulator body axes
3. $\mathbf{x}_a$ for $\mathbf{x}$ in airplane body axes
4. $\mathbf{x}_h$ for $\mathbf{x}$ in pilot head axes

If no subscript appears or if it is not one of the ones above (i, s, a, h), then that variable is not referenced to any one of these coordinate systems (and usually such a reference is irrelevant).

(ii) $\mathbf{1}_\alpha$ unit vector of coordinate α.

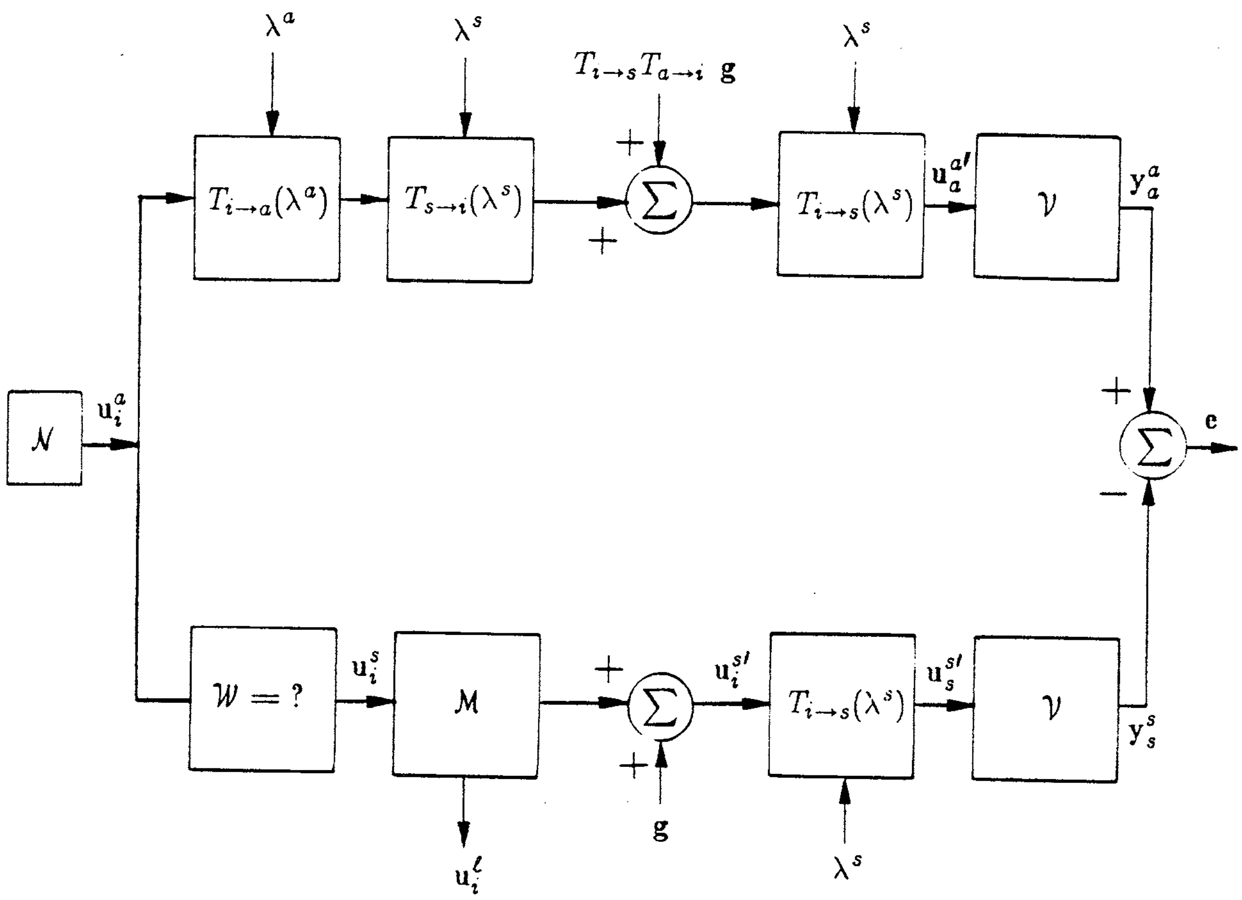


Figure 10. Restructuring the block diagram describing the flight simulator motion design problem.

(iii) The symbol $T_{s\to i}$ denotes a coordinate transformation from s axes to i axes, i.e. $\mathbf{x}_i = T_{s\to i}\mathbf{x}_s$. Note that $T_{s\to i}$ is a function of the Euler angles λ^s. Similarly, we define $T_{i\to a}$.

(iv) The symbol x denotes the states of a linear system. It may include physical and nonphysical states.

(v) The symbol u generally denotes control inputs.

9.3. Handling of axis systems

We have now defined the axes systems and can include them (Figure 9) in the formulation of the flight simulator design problem presented in Section 6 (Figure 6). Note that the additional axes transformation matrices depend on λ^a and λ^s. Our objective is to perform several block diagram manipulations on Figure 9 in order to introduce several simplifying approximations.

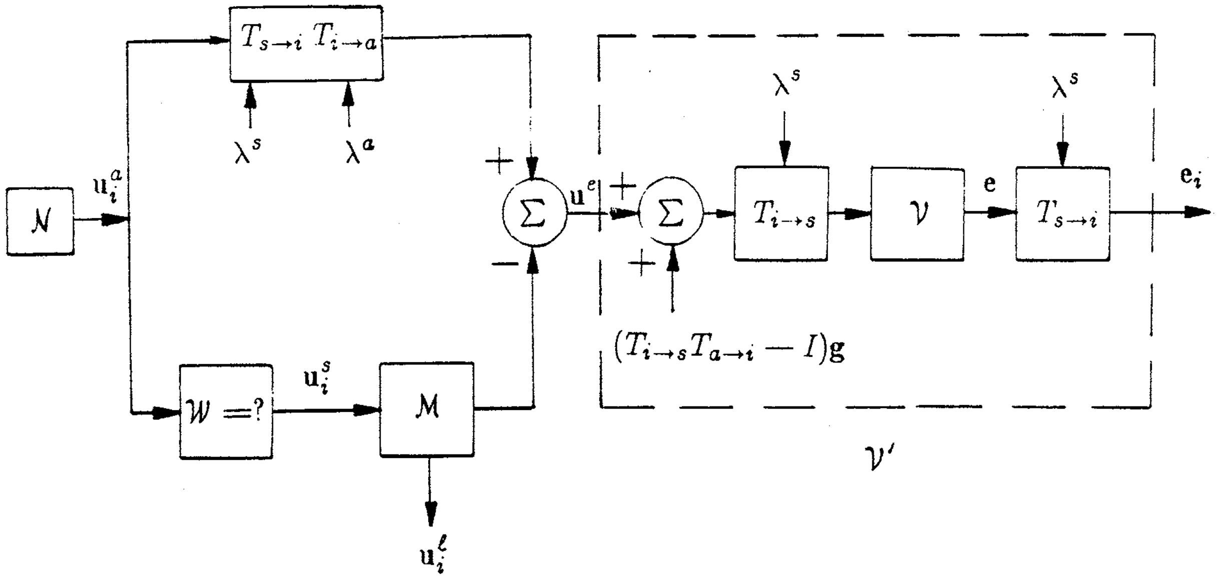


Figure 11. Simplified flight simulator motion design problem when using only one vestibular model $\mathcal{V}$.

The block diagram manipulations used to obtain Figure 10 are:

(i) The input airplane motions $\mathbf{u}_i^{a\prime}$, were replaced by $\mathbf{u}_i^a$, where $\mathbf{u}_i^a \triangleq \mathbf{u}_i^{a\prime} - \mathbf{g}$. Thus, we use the system $\mathcal{N}$ to generate $\mathbf{u}_i^a$ instead of $\mathcal{N}'$ that generated the anticipated airplane motions $\mathbf{u}_i^{a\prime}$. A consequence of this is that we are required to **add** one **g** vector to the actual airplane system input.

(ii) An identity axis transformation, $T_{s\to i}T_{i\to s} = I$, was added after $T_{i\to a}$.

(iii) The added one **g** vector from (i) was transformed to in between the two axis transformations added in (ii).

The next steps were applied in order to obtain the diagram in Figure 11:

(i) The error, **e**, is transformed to the inertial coordinate system. This transformation does not change the magnitude of **e** (**e** is a physical vector), which is used in our optimization criteria. Note that both the otolith and semicircular canal errors are transformed by $T_{s\to i}$ as vectors; as well as linear acceleration inputs to the otolith. The angular rate inputs to the semicircular canal require a different transformation matrix [Friedland73]. We will come back to this comment later on.

(ii) We **assume** that the vestibular models, $\mathcal{V}$, are linear with identical initial conditions so that we can sum the inputs of these systems, rather than their outputs, with no change in the computed error **e**. The effect of this linear approximation of $\mathcal{V}$ on the washout design is discussed in Chapter IV. The assumption of identical initial conditions for the vestibular systems of the *airplane pilot* and the *simulator pilot* is due to the parallel

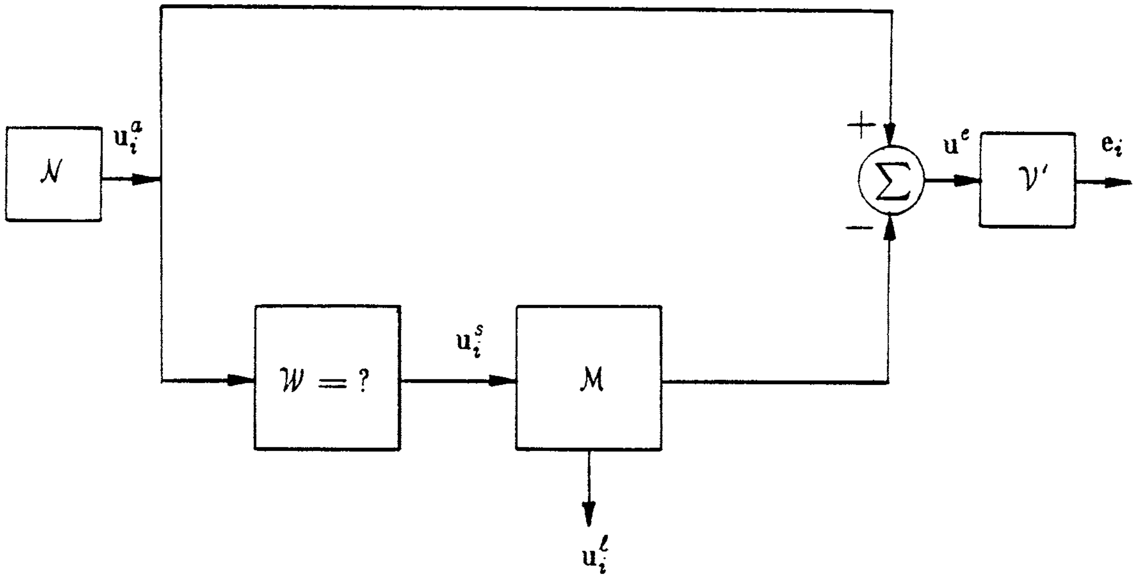


Figure 12. Simplified flight simulator motion design problem when approximating, $T_{s\to i}T_{i\to a} \approx I$.

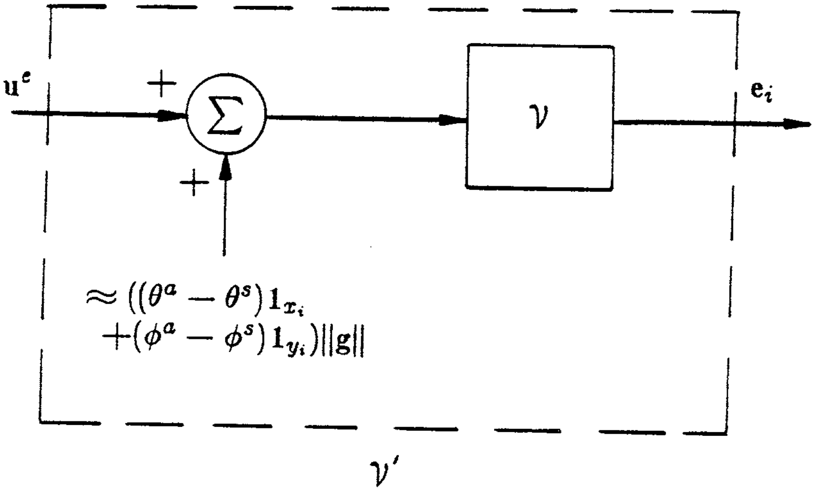

$\mathbf{u}^e$
$+$
Σ
$+$
$\mathcal{V}$
$\mathbf{e}_i$
$\approx ((\theta^a - \theta^s)\mathbf{1}_{x_i}$
$+(\phi^a - \phi^s)\mathbf{1}_{y_i})\|\mathbf{g}\|$
$\mathcal{V}'$

Figure 13. A linearized time invariant approximation of the vestibular model $\mathcal{V}'$.

construction of the imaginary *airplane pilot*. It should be noted that since most washout designs deal with the steady state solution, we obtain simulation discrepancies that are identical to those caused by nonidentical initial conditions.

At this point we proceed by making the approximation (Figure 12):

$$T_{s\to i}T_{i\to a} \approx I \tag{7}$$

i.e. the direction of the airplane and that of the simulator are the same. A linearized

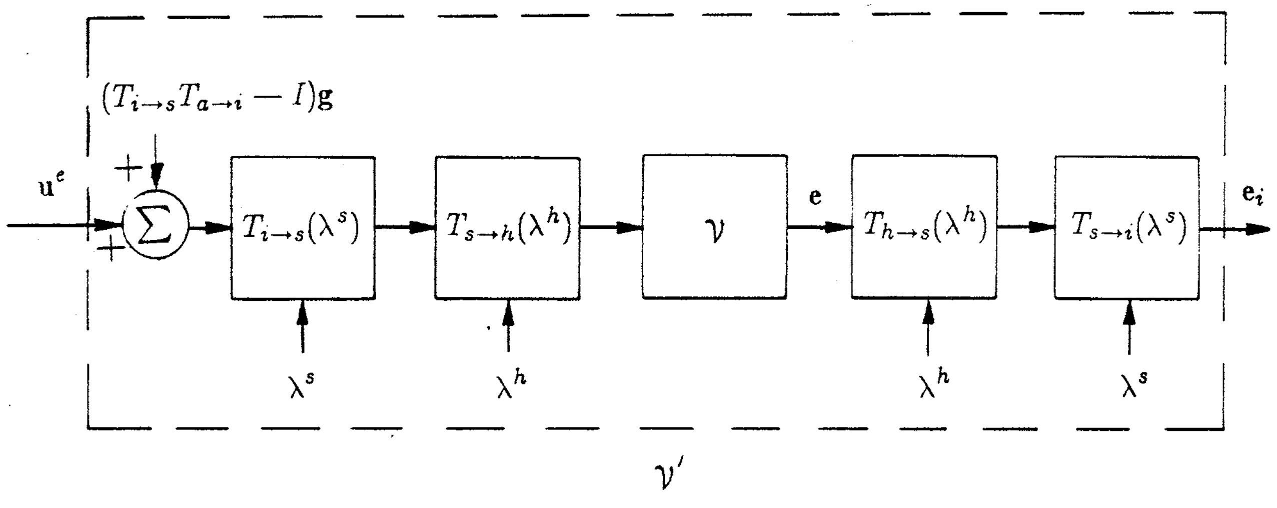


Figure 14. Including head movements in the flight simulator motion design problem.

time invariant approximation to $\mathcal{V}'$ (defined in Figure 11) is given in Figure 13, which is a valid approximation under the **assumption** that the angles $\lambda^s(t)$ are "small" and that they have a "slow" variation with time (compared to the vestibular model time constants). A further **assumption** required is that the vestibular models used to sense the three linear axes motions (otolith sensors) be identical (not even with different gains) and similarly for the semicircular canals. (In reality, the gains for the different axes need to be different.)

9.4. Including head movements

Along the same line of development, we can include pilot head rotations (we ignore translations) that are represented by $\lambda^h(t)$ as shown in Figure 14 (both pilots behave the same, so they have the same head movements $\lambda^h(t)$). The head movements are considered by recalling that the pilot's vestibular system rotates with angles $\lambda^h(t)$ on top of the rotations $\lambda^s(t)$ that we have already considered. Furthermore, as discussed before, we can add an orthogonal transformation to the components of $\mathbf{e}$ without affecting our cost function (depends on the magnitude of $\mathbf{e}$). One may be tempted to simplify (approximate) the new $\mathcal{V}'$ in Figure 14 by the one in Figure 11 and further to the one in Figure 13, but this approximation is quite poor, since head movements are usually much "faster" than the vestibular time constants. Nevertheless, we will ignore head movements in our formal solution.

The incorrect head movement approximation is somewhat "fixed" by the closed loop washout system implementation presented in Chapter VI.

An additional consideration is that head movements of the pilot can be ignored if we assume that the pilot initiates most of his head movements, and senses his head position (and rate) independently of his vestibular system (neck muscle receptors and other sensors), so that he can "correct" his vestibular outputs (or at least his perception), possibly through the vestibular efferent fibers we ignored before. However, this is not always true. An example is that of a pilot in a rotating environment, where the pilot is not aware of the rotation (easily arranged), such that a compensation for head movements cannot be performed. This has been experimental verified and is called the Coriolis illusion [Peterson69].

9.5. Consideration of the Motion Limitation Axis System

In the problem considered here we have limitations on $\mathbf{u}_i^\ell$ (controls and states of the motion base). However these limitations are in another axis system, namely the motion limitation axis system. Thus, the limitations are given by:

$$\mathbf{u}_\ell^\ell = T_{i\to\ell}(\mathbf{u}_i^\ell) \tag{8}$$

where $T_{i\to\ell}(\mathbf{u}_i^\ell)$ is the axis transformation [Dieudonne72]. In the case of a hexapod motion base system, this is a nonlinear, nonorthogonal transformation. The hexapod limitations are on the six leg lengths, ℓ_k, and the leg extension velocities $\dot{\ell}_k$ ($k = 1, \ldots, 6$).

Our motion cost is:

$$J_\ell = \left\{ \begin{array}{ll} 0 & \text{for } L_{\min} < \ell_k < L_{\max} \\ \infty & \text{for } \ell_k \le L_{\min} \quad \text{or} \quad \ell_k \ge L_{\max} \end{array} \right. \qquad \text{for all} \quad 1 \le k \le 6 \tag{9}$$

and similarly for extension rates:

$$J_{\dot\ell} = \left\{ \begin{array}{ll} 0 & \text{for } |\dot\ell_k| < \dot L_{\max} \\ \infty & \text{for } |\dot\ell_k| \ge \dot L_{\max} \end{array} \right. \qquad \text{for all} \quad 1 \le \ell_k \le 6 \tag{10}$$

If we approximate (9)–(10) by a quadratic cost, recalling that all the six hexapod legs are the same, then

$$J = \mathfrak{E}\{L^T L + r\dot L^T \dot L\} \tag{11}$$

where

$$L \triangleq \begin{pmatrix} \ell_1 \\ \ell_2 \\ \ell_3 \\ \ell_4 \\ \ell_5 \\ \ell_6 \end{pmatrix}, \qquad \dot{L} \triangleq \frac{dL}{dt}, \qquad \mathbf{u}_\ell^\ell \triangleq \begin{pmatrix} L \\ \dot{L} \end{pmatrix}. \tag{12}$$

Then from (11)–(12) the cost due to the motion-base limitation is:

$$J_\ell = \mathfrak{E}\{T_{i\to\ell}(\mathbf{u}_i^\ell)^T \begin{pmatrix} I & 0 \\ 0 & rI \end{pmatrix} T_{i\to\ell}(\mathbf{u}_i^\ell)\} \tag{13}$$

which involves higher order terms than quadratic and also introduces coupling between all six degrees of freedom of the motion-base (as we would expect).

In summary a nonlinear nonorthogonal axis limitation coordinate system (such as would be required in a hexapod motion-base) introduces into the problem a nonquadratic cost with coupling between all axes.

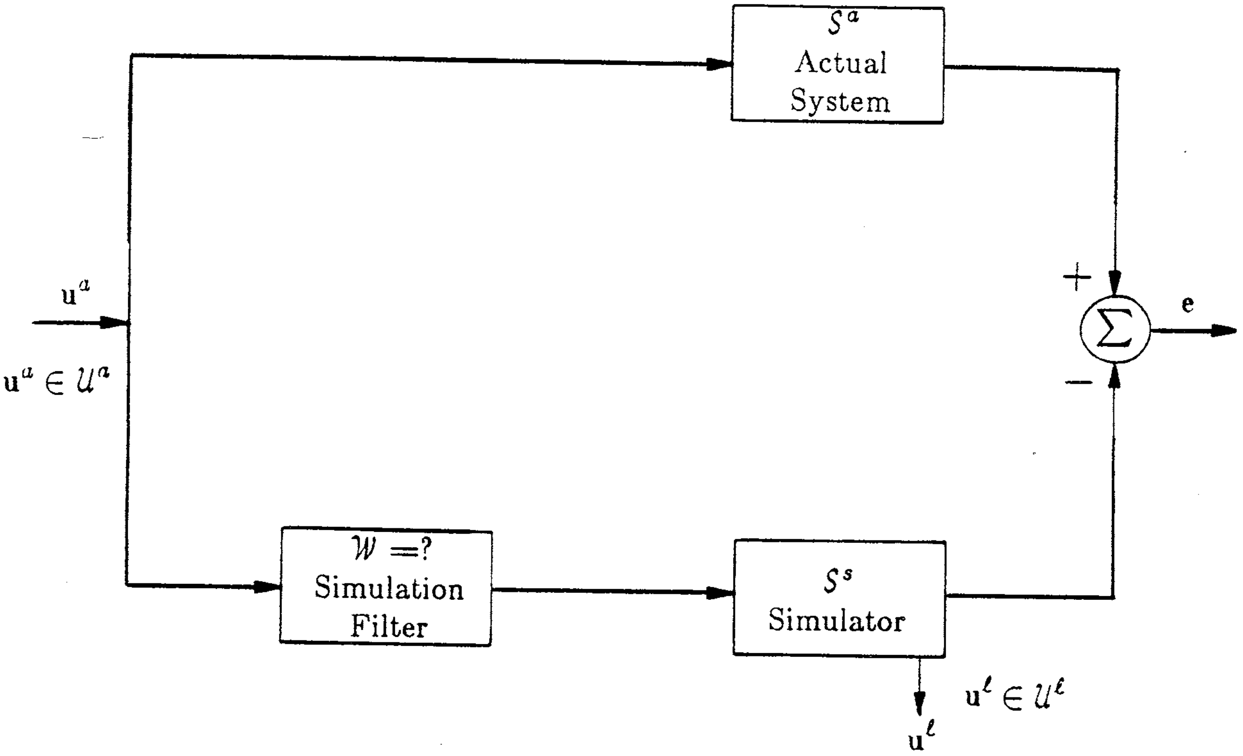


Figure 15. The ABSTRACT OPTIMAL SIMULATOR DESIGN PROBLEM.

10. Abstract Optimal Simulator Problem Statement

We now take the specific case of the optimal flight simulator motion design problem and formulate the general ABSTRACT OPTIMAL SIMULATOR PROBLEM shown in Figure 15. This general formulation has the advantage of being a condensed representation of our flight simulator motion design problem and includes also other optimal design problems.

Let us go through the transformations applied to the flight simulator design problem representation in Figure 6 in order to arrive at the abstract problem formulation of Figure 15. First we replace the anticipated airplane motion input $\mathbf{u}^{a\prime}(t)$ by $\mathbf{u}^a(t)$ defined by:

$$\mathbf{u}^a(t) = \mathbf{u}^{a\prime}(t) - \mathbf{g} \tag{14}$$

thus "absorbing" the summation of the one $\mathbf{g}$.

Second we add back one $\mathbf{g}$ to $\mathbf{u}^a(t)$ to form the correct input to $\mathcal{V}^a$. The combination of the added one g and system $\mathcal{V}^a$ (Pilot's INERTIAL MOTION

SENSOR Model) are defined to be the system $\mathcal{S}^a$. Similarly, we define $\mathcal{S}^s$ to include the one **g** addition and the cascade of system $\mathcal{M}$ (Simulator MOTION BASE DYNAMICS model) and $\mathcal{V}^s$ (Pilot's INERTIAL MOTION SENSOR Model). The system $\mathcal{S}^s$ has for an input $\mathbf{u}^s(t)$, the commands to the flight simulator motion base, and has for outputs both $\mathbf{y}^s(t)$ (physiological output of the pilot's sensory system) and $\mathbf{u}^\ell(t)$ the limiting variables. $\mathbf{u}^\ell(t)$ also includes the states of the simulator that have limitations associated with them. It is required that

$$\mathbf{u}^\ell(t) \in \mathcal{U}^\ell \tag{15}$$

which is to say that $\mathbf{u}^\ell(t)$ belongs to the constraint class $\mathcal{U}^\ell$ of possible $\mathbf{u}^\ell(t)$ that the simulator motion-base can operate within.

Then we replace the system $\mathcal{N}$ (anticipated airplane motion generation) by:

$$\mathbf{u}^a(t) \in \mathcal{U}^a \tag{16}$$

that is to say, $\mathbf{u}^a(t)$ belongs to some class (or set) of possible inputs $\mathcal{U}^a$ (a more generalized representation of the possible inputs $\mathbf{u}^a(t)$ than represented by $\mathcal{N}$).

Now, according to Figure 15, we can define:

ABSTRACT OPTIMAL SIMULATOR PROBLEM:
Given the set $\mathcal{U}^a$ and the constraint set $\mathcal{U}^\ell$, find a mapping $\mathbf{u}^a \rightarrow \mathbf{u}^s$ for all $\mathbf{u}^a \in \mathcal{U}^a$, so that $\mathbf{u}^\ell \in \mathcal{U}^\ell$ and such that a given norm of **e** is minimized.

In some cases, we will require a causal mapping from $\mathbf{u}^a$ to $\mathbf{u}^s$, that is a transformation $\mathcal{W}$ such that

$$\mathbf{u}^s(t) = \mathcal{W}(\mathbf{u}^a(\tau)\,;\ \text{for } -\infty \leq \tau \leq t), \tag{17}$$

so that $\mathbf{u}^s$ at time t does not depend on future values of $\mathbf{u}^a$.

In the next chapter, we solve a Linear Quadratic (L.Q.) version of the ABSTRACT SIMULATOR DESIGN PROBLEM.

References

[Baron76] Baron, S. and Sheldon, "Model of Human Control and Monitoring Based on Modern Control Theory," *Journal of Cybernetics and Information Sciences*, **1**, no. 1, spring 1976.

[Borah82] Borah, J., Young, L.R. and Curry, R.E., "Optimal Estimator Model For Human Spatial Orientation," *IEEE Transactions on Systems, Man and Cybernetics*, in press, 1983.

[Correia78] Correia, M.J. and Guedry, Jr., F.E., "The Vestibular System: Basic Biophysical and Physiological Mechanisms," Chapter 8 in *Handbook of Behavioral Neurobiology*, **1**, Sensory Integration, Masterton, R.B., Plenum, N.Y., 1978.

[Dechesne80] Dechesne, C. and Sans, A., "Control of the Vestibular Nerve Activity by the Efferent System in the Cat," *Acta Otolaryngol.*, **90**, pp. 82–85, 1980.

[Dieudonne72] Dieudonne, J.E. Parrish, R.V. and Bardusch, R.E., "An Actuator Extension Transformation for a Motion Simulator and an Inverse Transformation Applying Newton-Raphson's Method," NASA TN D-7067, 1972.

[Etkin72] Etkin, B., *Dynamics of Atmospheric Flight*, Wiley and Sons Inc., pp. 104–112, 1972.

[FCDL80] U.S. Air Force, "Flight Control Development Laboratory," Technical Brochure, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

[Friedland73] Friedland, B., Ling, C.-K. and Hutton, M.F., "Quasi-optimum Design of a Six-Degree-of-Freedom Moving Base Simulator Control System," NASA CR-23l2, p. 2, 1973.

[Guedry74] Guedry, F.E., "Psychophysics of Vestibular Sensation," in *Handbook of Sensory Physiology*, vol. **VI/2**, Koruhuber, H.H., Editor, Springer-Verlag, Berlin, 1974.

[Hosman78] Hosman, R.J.A.W. and van der Vaart, J.C., "Vestibular Models and Thresholds of Motion Perception. Results of Tests in a Flight Simulator," LR-265, Delft University Technical Department of Aerospace Engineering, Delft, The Netherlands, April, 1978.

[Hosman79] Hosman, R.J.A.W., van der Vaart, J.C. and van de Moesdijk, G.A.J., "Optimalization and Evaluation of Linear Motion Filters," *Annual Conference on Manual Control*, March, 1979.

[Huang79] Huang, J.-K., "Visual Field Influence on Motion Sensation in Yaw and on Manual Roll Stabilization," M.Sc. Thesis M.I.T., June, 1979.

[Ish-Shalom79] Ish-Shalom, J., "Head Rotation Monitoring Device," report, Man-Vehicle Lab., Dep. of Aeronautics and Astronautics, M.I.T., Cambridge, Ma. 02139, June, 1979.

[Kleinman71] Kleinman, D.L., Baron, S. and Levinson, W.H., "A Control Theoretic Approach to Manned-Vehicle Systems Analysis," *IEEE Transactions on Automatic Control*, **16-AC**, no. 6, pp. 824–832, Dec., 1971.

[Kosut79] Kosut, R.L., "Nonlinear Optimal Cue-Shaping Filters for Motion Base Simulators," *Journal of Guidance and Control*, 2, pp. 486–490, Dec., 1979.

[Martin80] Martin, E.A., "Motion and Force Simulation Systems," Simulator Course at Wright-Patterson Air Force Base, Ohio, 1980.

[McRuer65] McRuer, D., Graham, D., Krendel, E. and Reiseuer, W.,Jr., *Human Pilot Dynamics in Compensatory System—Theory Models and Experiments with controlled Elements and Forcing Function Variations*, AFFDL-TR-65-15, Wright-Patterson AFB, 1965.

[Oman71] Oman, C.M., "Vestibular Models and Flight Simulator Motion Washout Techniques," NASA Proposal prepared in the Man Vehicle Laboratory, MIT, 1971.

[Oman80] Oman, C.M., "A Heuristic Mathematical Model for the Dynamics of Sensory Conflict and Motion Sickness," MVT-80-1, Man Vehicle Laboratory Report, MIT, Cambridge, MA., 1980.

[Oman82] Oman, C.M., "Space Motion Sickness and Vestibular Experiments in Spacelab," SAE Technical Paper Series 820233, Twelfth Intersociety Conference on Environmental Systems, San Diego, California, July, 1982.

[Ormsby74] Ormsby, C.C., "Model of Human Dynamics Orientation," Ph.D. Thesis, M.I.T., Cambridge, Ma., Jan., 1974.

[Parrish73] Parrish, R.V., Dieudonne, J.E., Bowles, R.L. and Martin, Jr. D.J.,, "Coordinated adaptive Washout for Motion Simulators," presented at AIAA Conference, Palo Alto, California, 1973.

[Parrish78] Parrish, R.V., "Platform Motion for Flight Simulators—Let's Be Realistic," AIAA Flight Simulator Technologies Conference, Arlington, Texas, 1978.

[Peters69] Peters, R.A., "Dynamics of the Vestibular System and Their Relation to Motion Perception, Spatial Disorientation and Illusions," NASA CR-1309, 1969.

[Puig78] Puig, J.A., Harris, W.T. and Ricard, G.L., "Motion in Flight Simulation: An Annotated Bibliography," Technical Report: NAVTRAEQUIPCEN IH-298, Naval Training Equipment Center, Orlando, Florida, 1978.

[Ramnath80] Ramnath, R.V. and Tao, Y-C, "Nonlinear Attitude Dynamics of Satellites," presented at the Asilomar conference on New Approaches to Nonlinear Problems, Dec., 1979, monterey, California, US, Charles Stark Draper Laboratory report P-992, Jan., 1980.

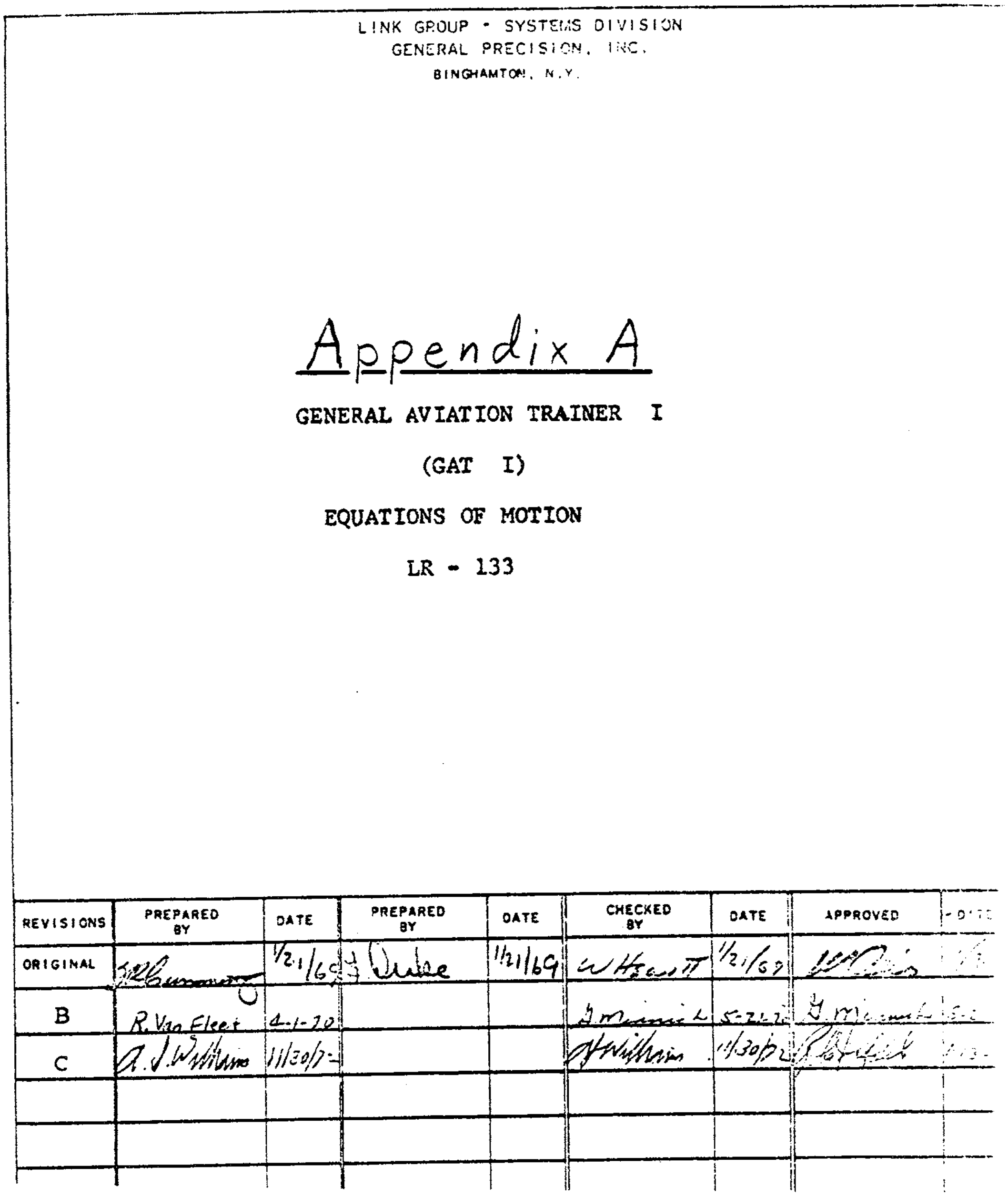
[Sinacori77] Sinacori, J.B., Stapleford, R.L., Jewell, W.F. and Lebman, J.M., Researcher's Guide to the FSAA Ames Flight Simulator for Advanced Aircraft (FSAA)," NASA CR-2875.

[Sturgeon81] Sturgeon, W. R., "Controllers for Aircraft Motion Simulators," *Journal Guidance and Control*, 4, no. 2, pp. 184–191, April, 1981.

[Van-Trees68] Van-Trees, H.L., *Detection, Estimation and Modulation Theory*, part I, Wiley, 1968.

[Von-Gierke61] Von Gierke, H.E. and Steinmetz, E., *Motion Devices for Linear and Angular Oscillation and for Abrupt Acceleration Studies on Human Subjects (Impact),* Publication 903, National Academy of Sciences—National Research Council, Washington, D.C, 1961.

[Young82] Young, L.R., "Human Orientation in Space," AIAA Dryden lecture in research, in AIAA 20th Aerospace Sciences Meeting, Orlando, Florida, Jan., 1982.



LINK GROUP • SYSTEMS DIVISION
GENERAL PRECISION, INC.
BINGHAMTON, N.Y.

Appendix A

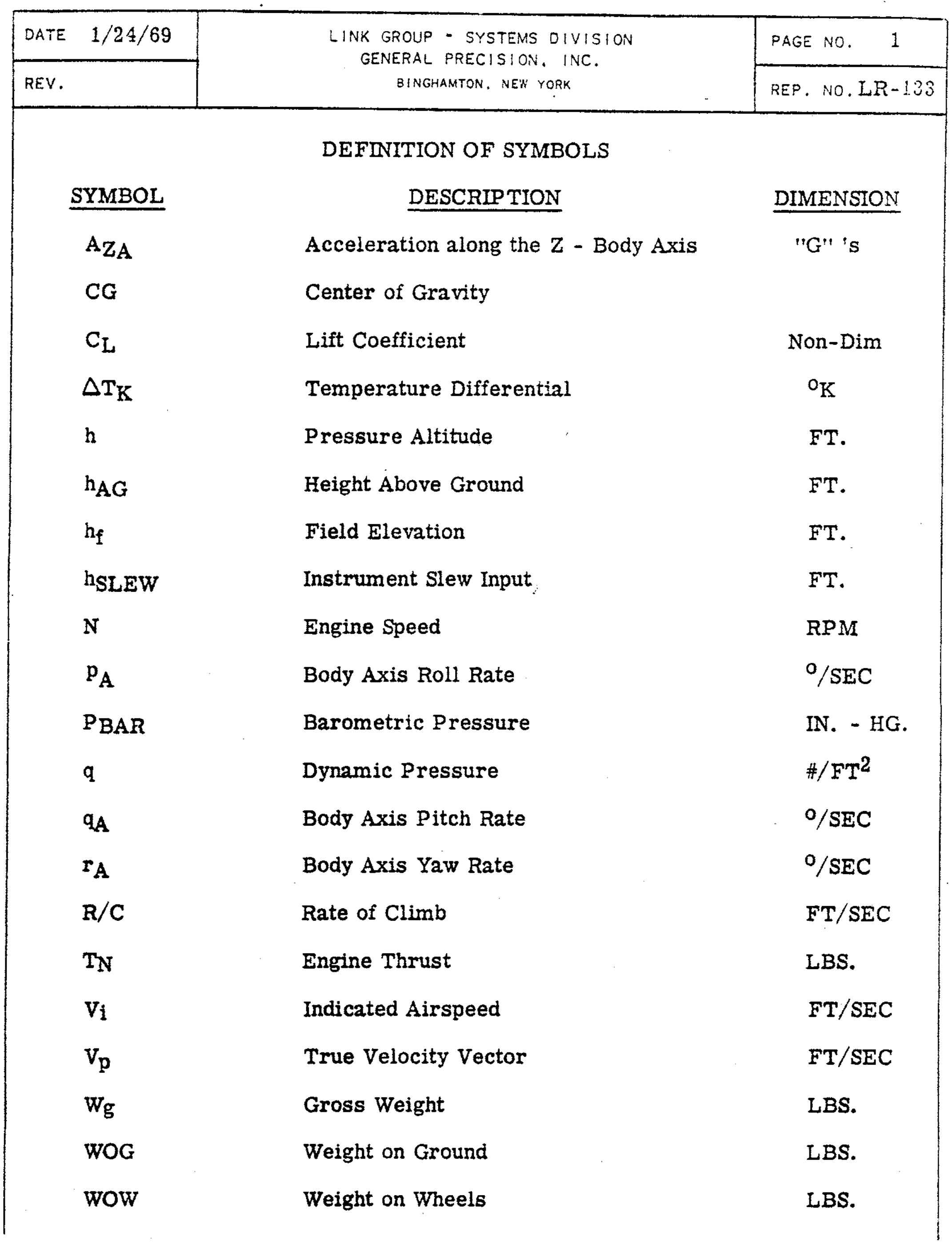
GENERAL AVIATION TRAINER I

(GAT I)

EQUATIONS OF MOTION

LR - 133

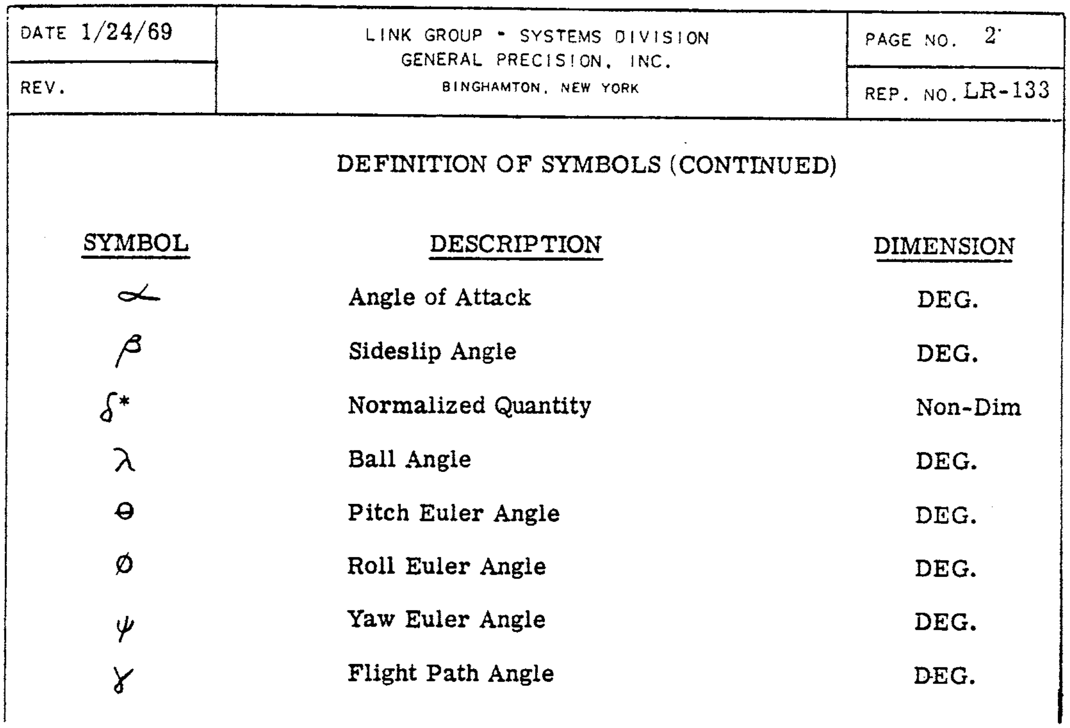
REVISIONS	PREPARED BY	DATE	PREPARED BY	DATE	CHECKED BY	DATE	APPROVED	DATE
ORIGINAL		1/21/69	J. Duke	11/21/69	W Hewitt	1/21/69		
B	R. Van Fleet	4-1-70			G. Minnick	5-21-7[illegible]	G. Minnick	[illegible]
C	A. J. Williams	11/30/7-			Williams	11/30/72		[illegible]



DATE 1/24/69	LINK GROUP - SYSTEMS DIVISION GENERAL PRECISION, INC. BINGHAMTON, NEW YORK	PAGE NO. 1
REV.		REP. NO. LR-133

DEFINITION OF SYMBOLS

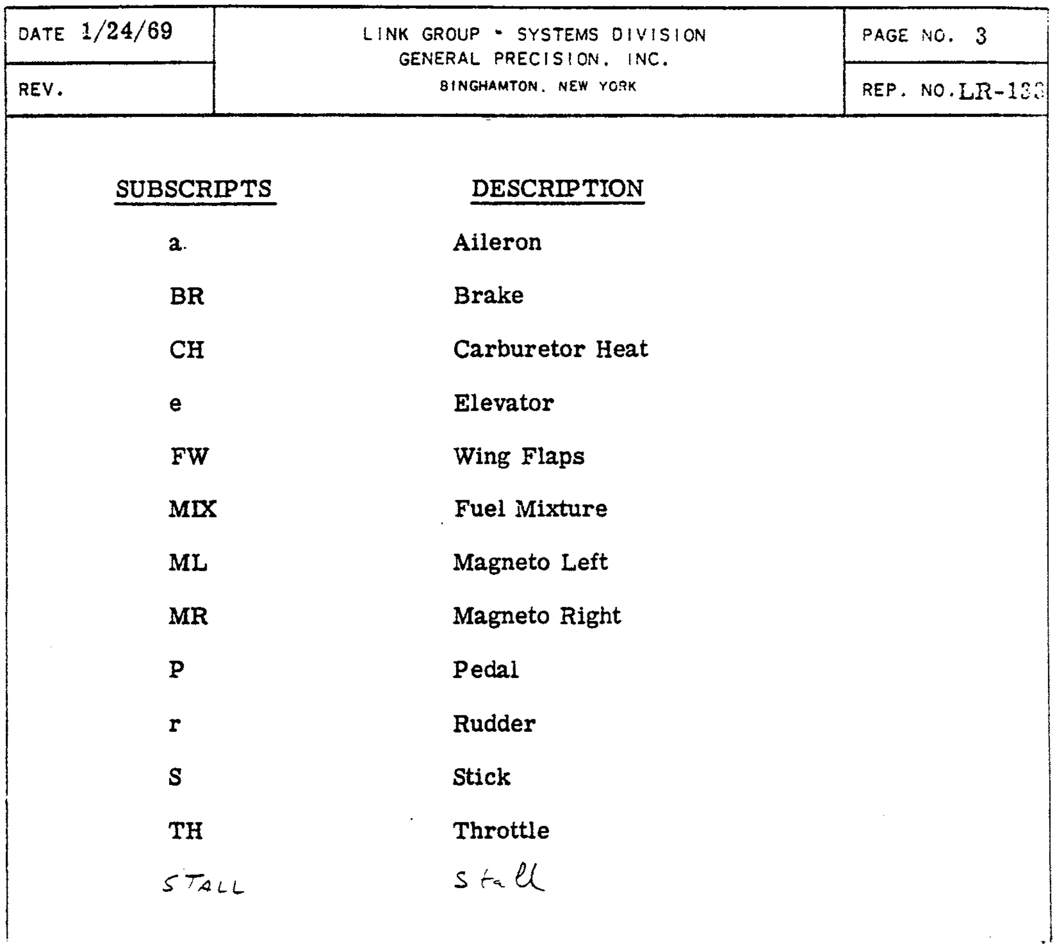
SYMBOL	DESCRIPTION	DIMENSION
A_{ZA}	Acceleration along the Z - Body Axis	"G" 's
CG	Center of Gravity	
C_L	Lift Coefficient	Non-Dim
ΔT_K	Temperature Differential	°K
h	Pressure Altitude	FT.
h_{AG}	Height Above Ground	FT.
h_f	Field Elevation	FT.
h_{SLEW}	Instrument Slew Input	FT.
N	Engine Speed	RPM
p_A	Body Axis Roll Rate	°/SEC
P_{BAR}	Barometric Pressure	IN. - HG.
q	Dynamic Pressure	#/FT^2
q_A	Body Axis Pitch Rate	°/SEC
r_A	Body Axis Yaw Rate	°/SEC
R/C	Rate of Climb	FT/SEC
T_N	Engine Thrust	LBS.
V_i	Indicated Airspeed	FT/SEC
V_p	True Velocity Vector	FT/SEC
W_g	Gross Weight	LBS.
WOG	Weight on Ground	LBS.
WOW	Weight on Wheels	LBS.



DATE 1/24/69	LINK GROUP • SYSTEMS DIVISION GENERAL PRECISION, INC. BINGHAMTON, NEW YORK	PAGE NO. 2
REV.		REP. NO. LR-133

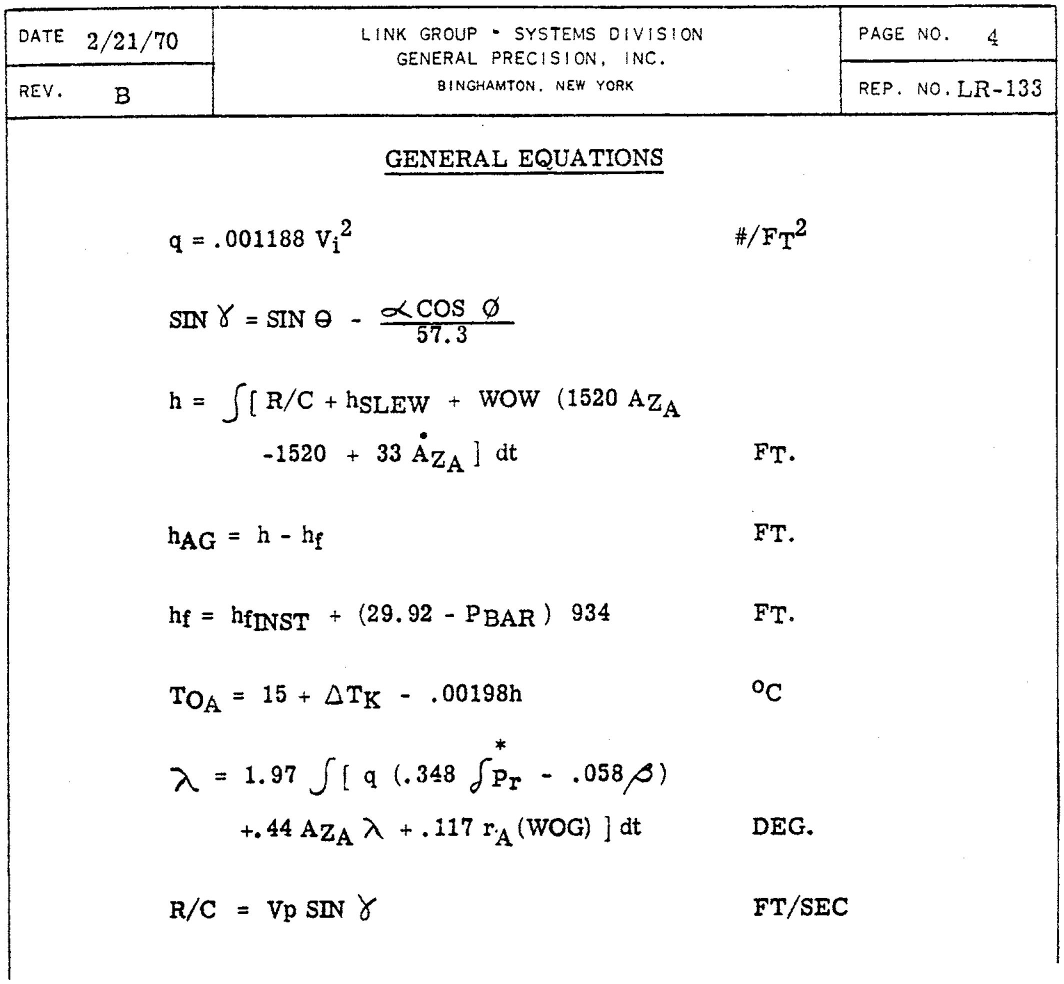
DEFINITION OF SYMBOLS (CONTINUED)

SYMBOL	DESCRIPTION	DIMENSION
α	Angle of Attack	DEG.
β	Sideslip Angle	DEG.
δ^*	Normalized Quantity	Non-Dim
λ	Ball Angle	DEG.
θ	Pitch Euler Angle	DEG.
ϕ	Roll Euler Angle	DEG.
ψ	Yaw Euler Angle	DEG.
γ	Flight Path Angle	DEG.



DATE 1/24/69	LINK GROUP • SYSTEMS DIVISION GENERAL PRECISION, INC. BINGHAMTON, NEW YORK	PAGE NO. 3
REV.		REP. NO. LR-133

SUBSCRIPTS	DESCRIPTION
a	Aileron
BR	Brake
CH	Carburetor Heat
e	Elevator
FW	Wing Flaps
MIX	Fuel Mixture
ML	Magneto Left
MR	Magneto Right
P	Pedal
r	Rudder
S	Stick
TH	Throttle
STALL	Stall



DATE 2/21/70	LINK GROUP • SYSTEMS DIVISION GENERAL PRECISION, INC. BINGHAMTON, NEW YORK	PAGE NO. 4
REV. B		REP. NO. LR-133

GENERAL EQUATIONS

$$q = .001188\ V_i^2 \qquad \#/FT^2$$

$$\text{SIN}\ \gamma = \text{SIN}\ \Theta\ -\ \frac{\alpha\ \text{COS}\ \varnothing}{57.3}$$

$$h = \int [\ R/C + h_{SLEW} + WOW\ (1520\ A_{Z_A} - 1520 + 33\ \dot{A}_{Z_A}\ ]\ dt \qquad \text{FT.}$$

$$h_{AG} = h - h_f \qquad \text{FT.}$$

$$h_f = h_{f_{INST}} + (29.92 - P_{BAR})\ 934 \qquad \text{FT.}$$

$$T_{OA} = 15 + \Delta T_K - .00198h \qquad {}^{o}C$$

$$\lambda = 1.97 \int [\ q\ (.348 \overset{*}{\int P_r} - .058\beta) + .44\ A_{Z_A}\ \lambda + .117\ r_A\ (WOG)\ ]\ dt \qquad \text{DEG.}$$

$$R/C = Vp\ \text{SIN}\ \gamma \qquad \text{FT/SEC}$$

LIFT COEFFICIENT

$$C_L = .077 + .077\alpha + .62\, \delta^*_{FW} + .445 \text{ X } 10^{-3}\ T_N$$

$$A_{Z_A} = \frac{157\, q\, C_L}{W_g} \left(1 - \delta^*_{STALL}\right) \qquad \text{G's}$$

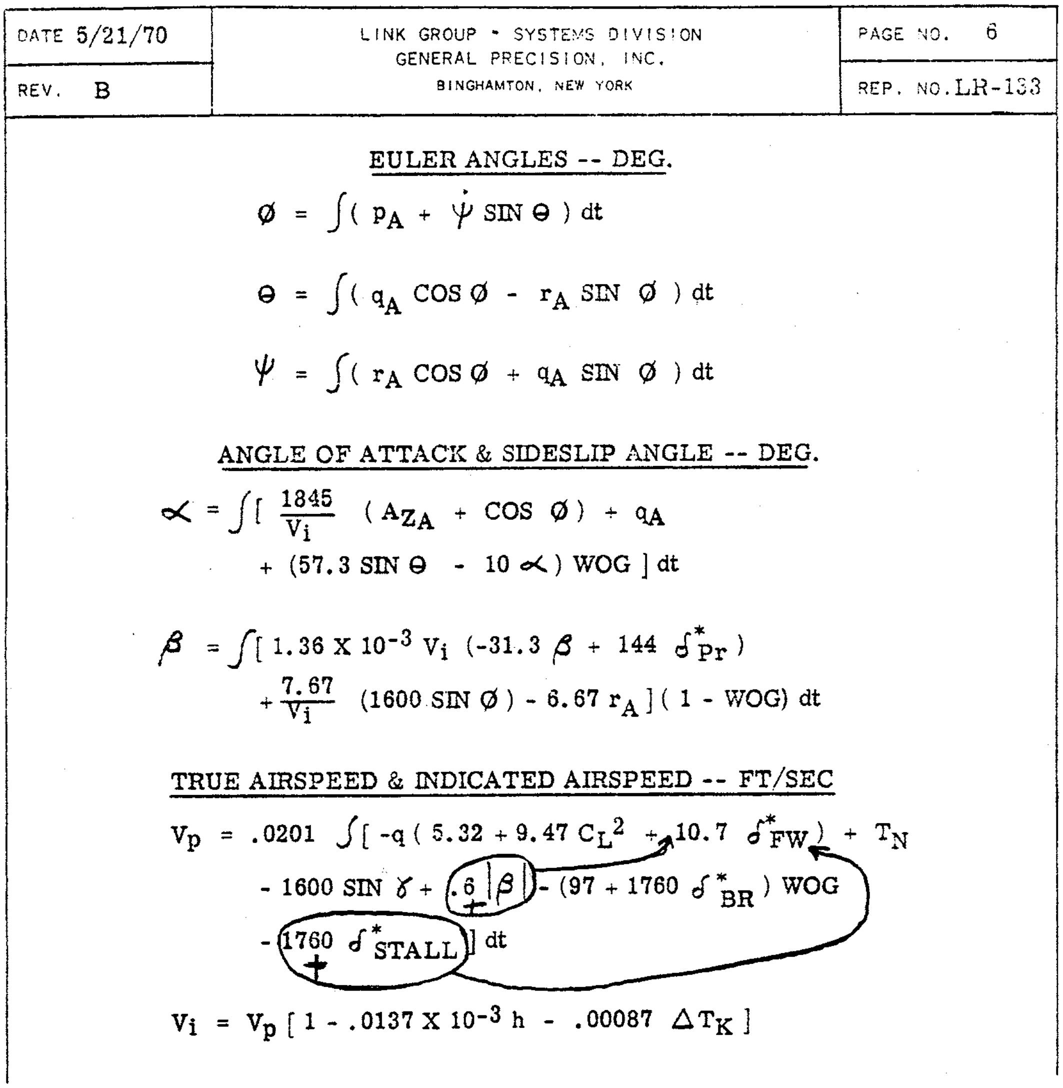
BODY AXES ANGULAR RATES -- DEG/SEC

$$\begin{aligned} p_A = 338 \int [\, & q\,(-.00177\beta - .0260\, \delta^*_{Sa} + .00148\,) \\ & + V_i\,(.057 \text{ X } 10^{-3} r_A - .084 \text{ X } 10^{-3}\ p_A\,) - .223 \text{ X } 10^{-3}\ T_N \\ & - (5.53 \text{ SIN } \emptyset + .0205\, p_A\,) \text{ WOG }]\, dt \end{aligned}$$

$$\begin{aligned} q_A = 37.7 \int [\, & q\,(-.3\, \delta^*_{Se} - .00288 - .314\, C_L - .55\, \delta^*_{STALL} \\ & - .0183\, \delta^*_{FW} + .0933 \text{ X } 10^{-3}\ T_N + .31\ \delta^*_{TRIM}\,) \\ & -1.62 \text{ X } 10^{-3}\ q_A\ V_i + 0.141 \text{ X } 10^{-3}\ T_N - \text{ KCG} \\ & + (-.\overbrace{402\, q_A - 78 \text{ SIN } \Theta + 3.66}^{\textcircled{1}} - 1.33 \text{ WOW) WOG}]\, dt \end{aligned}$$

$$\begin{aligned} r_A = 173 \int [\, & q\,(-.0292\, \delta^*_{Pr} + .00131\beta + .00240\, C_L\ \delta^*_{Sa}\,) \\ & - .0346 \text{ X } 10^{-3}\ r_A\ V_i - (\,.00575\, r_A + 1.2\, \delta^*_{Pr}\,) \text{ WOW }]\, dt \end{aligned}$$

$\overbrace{\qquad\qquad}^{\textcircled{1}}$ NEVER LESS THAN ZERO



DATE 5/21/70	LINK GROUP • SYSTEMS DIVISION GENERAL PRECISION, INC. BINGHAMTON, NEW YORK	PAGE NO. 6
REV. B		REP. NO. LR-133

EULER ANGLES -- DEG.

$$\emptyset = \int (p_A + \dot{\psi} \text{ SIN } \Theta) \, dt$$

$$\Theta = \int (q_A \text{ COS } \emptyset - r_A \text{ SIN } \emptyset) \, dt$$

$$\psi = \int (r_A \text{ COS } \emptyset + q_A \text{ SIN } \emptyset) \, dt$$

ANGLE OF ATTACK & SIDESLIP ANGLE -- DEG.

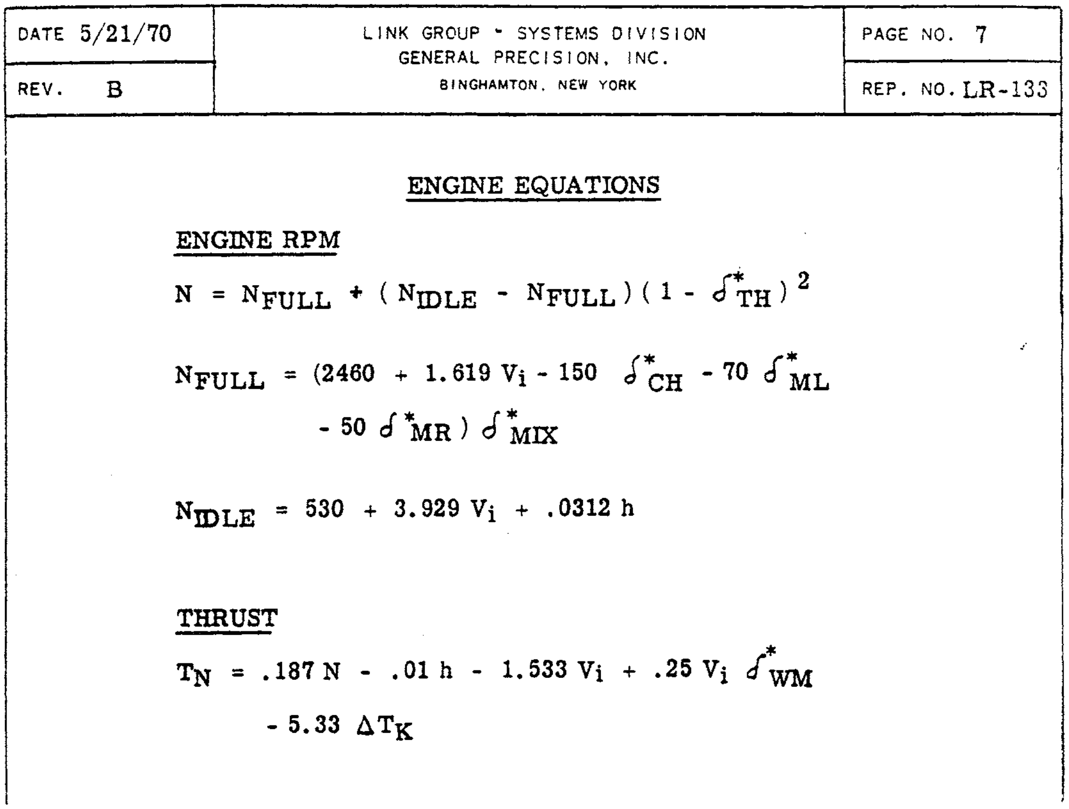
$$\alpha = \int [\frac{1845}{V_i} (A_{ZA} + \text{COS } \emptyset) + q_A + (57.3 \text{ SIN } \Theta - 10 \alpha) \text{ WOG }] \, dt$$

$$\beta = \int [1.36 \text{ X } 10^{-3} V_i (-31.3 \beta + 144 \delta^*_{Pr}) + \frac{7.67}{V_i} (1600 \text{ SIN } \emptyset) - 6.67 r_A] (1 - \text{WOG}) \, dt$$

TRUE AIRSPEED & INDICATED AIRSPEED -- FT/SEC

$$V_p = .0201 \int [-q (5.32 + 9.47 C_L^2 + 10.7 \delta^*_{FW}) + T_N - 1600 \text{ SIN } \gamma + \boxed{\underset{+}{.6} |\beta|} - (97 + 1760 \delta^*_{BR}) \text{ WOG } - \boxed{\underset{+}{1760} \delta^*_{STALL}}] \, dt$$

$$V_i = V_p [1 - .0137 \text{ X } 10^{-3} h - .00087 \Delta T_K]$$



DATE 5/21/70	LINK GROUP - SYSTEMS DIVISION GENERAL PRECISION, INC. BINGHAMTON, NEW YORK	PAGE NO. 7
REV. B		REP. NO. LR-133

ENGINE EQUATIONS

ENGINE RPM

$$N = N_{FULL} + (N_{IDLE} - N_{FULL})(1 - \delta^*_{TH})^2$$

$$N_{FULL} = (2460 + 1.619\ V_i - 150\ \delta^*_{CH} - 70\ \delta^*_{ML} - 50\ \delta^*_{MR})\ \delta^*_{MIX}$$

$$N_{IDLE} = 530 + 3.929\ V_i + .0312\ h$$

THRUST

$$T_N = .187\ N - .01\ h - 1.533\ V_i + .25\ V_i\ \delta^*_{WM} - 5.33\ \Delta T_K$$

Chapter III

Solution of the Linear Quadratic Case

We start by stating the *Linear Quadratic* (L.Q.) case. It is shown that the L.Q. optimal washout solution is a combination of the solutions to two subproblems: a *deterministic* and a *stochastic* one. The deterministic washout is the solution to the problem of how to move the flight simulator cab, given the past and future motions of the real airplane, without violating the flight simulator motion constraints while producing the least possible vestibular error between a pilot flying an imaginary reference airplane and a pilot flying the simulator. The L.Q. optimization problem is relatively simple and can be solved in the frequency domain using the calculus of variations. The transfer matrix $\overline{W}(s)$ obtained, transforms the computed airplane motion inputs, to the controls of the flight simulator motion-base as outputs. $\overline{W}(s)$ is obtained in an analytic closed form, such that many of its properties are quite transparent. These properties serve as a guide for the study of the properties of the solution of the stochastic problem which are somewhat similar, but are harder to see. The final form of the deterministic solution is a set of time functions, computed in advance, which are used as part of the control signals applied to the flight simulator motion-base.

The stochastic part of the problem answers the question of how to modify the previously computed controls, given by the deterministic washout, to account for variations of the airplane computed motions away from those used in the design of the deterministic washout. Thus we try to accomplish our two goals—limit the simulator cab motion to the given constraints, while sacrificing as little as possible in the way of motion-quality provided to the simulator pilot. This problem can be solved independently, as if the computed deterministic control signals were zero. The class of unknown airplane motions is modeled as colored Gaussian noise, generated by a white Gaussian noise process filtered through a pilot control model cascaded to a model of the airplane dynamics, linearized about the ensemble expected value of the considered airplane motions. The ensemble expected value was already considered in the deterministic problem solution. This overall model is named $\mathcal{N}$, which stands for the noise shaping filter. The solution of the stochastic problem is obtained by combining the equations of $\mathcal{N}$ and two other models: the *Vestibular system*, $\mathcal{V}$, and the *Motion base*, $\mathcal{M}$. As a result we obtain the augmented linear system $\tilde{\mathcal{S}}$. By using a quadratic optimization criterion J (cost function), we formulated a standard L.Q. stochastic optimization problem, whose solution is linear feedback of the augmented system $\tilde{\mathcal{S}}$ states. The steady state feedback gains are obtained by calculating the solution to an *Algebraic Riccati Equation*, (ARE) (which is a matrix quadratic equation).

The number of states in $\tilde{\mathcal{S}}$ is prohibitively large, even for the lowest possible dimension modeling of $\mathcal{N}$, $\mathcal{V}$ and $\mathcal{M}$. Thus it is important to be able to solve numerically a smaller dimension ARE. Beyond that, in order to best handle the design of such a large system we need to learn some of the solution's general p roperties that reflect the problem's structure. The properties of the solution that we obtain here can be used to find the ARE solution by solving several much smaller matrix equations. An even more important consequence is that the solution is decoupled in such a way that only two physical dimensions are coupled at the same time. This enables us to implement the optimal washout easily. Furthermore this result makes it reasonable to think of complicating the problem and its solution by some consideration of nonlinearities as described later in chapter IV.

Another important observation is that if the cost matrices are constructed appropriately (plausible to our design) the poles of the optimal washout are independent of $\mathcal{N}$. In other words, they do not depend on the details of the class of airplane maneuvers that are simulated, but depend only on a gross measure of their amplitude. Furthermore the "motion" of the poles with a change in the amplitude of the maneuvers can be analyzed and "predicted". These two observations are used in chapter VI to compare the optimal washout to other experimentally optimized washouts and to confirm the empirical design rules [Sinacori77] used by simulator practitioners to set the initial values of the *washout* parameters. Several other properties useful in the implementation of the Optimal Washout System (OWS) are also derived.

The next development is a time-varying washout system which is formulated and solved. The time variation emerges from the expected time changes in the model $\mathcal{N}$ that represents the class of expected airplane motions. In the case where the changes in the parameters of $\mathcal{N}$ are "slow", this new *Time-Varying Optimal Washout System* can also be viewed as an adaptive OWS. In chapter VII, we describe an implementation of the OWS on a GAT-1 flight simulator which enables the use of such a time-varying/adaptive OWS.

Consideration of the OWS implementation is discussed in Chapter VI.

1. Statement of the Linear Quadratic Case

We assume that $\mathcal{S}^s$ and $\mathcal{S}^a$ are both *linear, time invariant, finite dimensional* systems, whose state equations are:

$$\mathcal{S}^s: \quad \begin{aligned} \dot{\mathbf{x}}^s(t) &= A^s\mathbf{x}^s(t) + B^s\mathbf{u}^s(t) && (1)\\ \mathbf{y}^s(t) &= C^s\mathbf{x}^s(t) + D^s\mathbf{u}^s(t) && (2)\\ \mathbf{u}^\ell(t) &= C^\ell\mathbf{x}^s(t) + D^\ell\mathbf{u}^s(t) && (3) \end{aligned}$$

$$\mathcal{S}^a: \quad \begin{aligned} \dot{\mathbf{x}}^a(t) &= A^a\mathbf{x}^a(t) + B^a\mathbf{u}^a(t) && (4)\\ \mathbf{y}^a(t) &= C^a\mathbf{x}^a(t) + D^a\mathbf{u}^a(t) && (5) \end{aligned}$$

where A^s, B^s, C^s, D^s, C^ℓ, D^ℓ and A^a, B^a, C^a, D^a are the parameters and $\mathbf{x}^s(t)$, $\mathbf{x}^a(t)$ are the state vectors of the simulated and actual systems, respectively. The error vector is defined by:

$$\mathbf{e}(t) = \mathbf{y}^a(t) - \mathbf{y}^s(t)\,. \tag{6}$$

Next, we select a quadratic cost function:

$$J = \mathfrak{E}\{\lim_{T\to\infty} \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} \mathbf{e}^T(t)Q\mathbf{e}(t) + \rho\mathbf{u}^{\ell^T}(t)R\mathbf{u}^\ell(t)\,dt\} \tag{7}$$

which minimizes a quadratic norm of the error, $\mathbf{e}$, and penalizes quadratically the limitation vector, $\mathbf{u}^{\ell}$. The expected value, $\mathfrak{E}\{\ \}$, is taken over the ensemble of all inputs $\mathbf{u}^a(t) \in \mathcal{U}^a$. It is assumed that $\rho > 0$, that $D^{\ell^T} R D^{\ell}$ is a positive definite matrix and that Q is a positive semidefinite matrix.

Our next step is to separate this L.Q. problem into two subproblems: one deterministic and one stochastic. These problems are solved independently to obtain the solution to our L.Q. optimal simulator problem which defines the operator $\mathcal{W}$ which computes $\mathbf{u}^s(t)$ from $\mathbf{u}^a(\tau)$:

$$\mathbf{u}^s(t) = \mathcal{W}(\mathbf{u}^a(\tau)\,; \quad \text{for } -\infty < \tau < \infty)\,. \tag{8}$$

2. Separation Into Deterministic And Stochastic Problems

To obtain the deterministic subproblem we take the expected values of equations (1)–(5):

$$\mathfrak{S}^s: \qquad \begin{aligned} \dot{\bar{\mathbf{x}}}^s(t) &= A^s \bar{\mathbf{x}}^s(t) + B^s \bar{\mathbf{u}}^s(t) && (9) \\ \bar{\mathbf{y}}^s(t) &= C^s \bar{\mathbf{x}}^s(t) + D^s \bar{\mathbf{u}}^s(t) && (10) \\ \bar{\mathbf{u}}^{\ell}(t) &= C^{\ell} \bar{\mathbf{x}}^s(t) + D^{\ell} \bar{\mathbf{u}}^s(t) && (11) \end{aligned}$$

$$\mathfrak{S}^a: \qquad \begin{aligned} \dot{\bar{\mathbf{x}}}^a(t) &= A^a \bar{\mathbf{x}}^a(t) + B^a \bar{\mathbf{u}}^a(t) && (12) \\ \bar{\mathbf{y}}^a(t) &= C^a \bar{\mathbf{x}}^a(t) + D^a \bar{\mathbf{u}}^a(t) && (13) \end{aligned}$$

where we used the over-bar to denote the expected value of a variable over the input ensemble $\mathbf{u}^a(t) \in \mathcal{U}^a$ (e.g. $\bar{\mathbf{x}}^a(t) = \mathfrak{E}\{\mathbf{x}^a(t)\}$). It is **assumed** that the following order of operations can be interchanged so that:

$$\dot{\bar{\mathbf{x}}}^a(t) \triangleq \frac{d}{dt}\mathfrak{E}\{\mathbf{x}^a(t)\} = \mathfrak{E}\{\,\frac{d}{dt}\mathbf{x}^a(t)\,\} \tag{14}$$

where a similar relation holds for $\dot{\bar{\mathbf{x}}}^s(t)$. Now we can define the stochastic part, as:

$$\tilde{\mathbf{x}}^a(t) \triangleq \mathbf{x}^a(t) - \bar{\mathbf{x}}^a(t) \tag{15}$$

for the actual states and similarly define $\tilde{\mathbf{x}}^s(t)$, $\tilde{\mathbf{u}}^a(t)$, $\tilde{\mathbf{u}}^s(t)$, $\tilde{\mathbf{y}}^a(t)$, $\tilde{\mathbf{y}}^s(t)$, $\tilde{\mathbf{e}}(t)$, and $\tilde{\mathbf{u}}^{\ell}(t)$. Thus, we substitute the expressions:

$$\mathbf{e}(t) = \bar{\mathbf{e}}(t) + \tilde{\mathbf{e}}(t) \tag{16}$$

and

$$\mathbf{u}^{\ell}(t) = \bar{\mathbf{u}}^{\ell}(t) + \tilde{\mathbf{u}}^{\ell}(t) \tag{17}$$

into (7) and obtain:

$$J = \bar{J} + \tilde{J} \tag{18}$$

where

$$\bar{J} = \lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \bar{\mathbf{e}}^{T}(t)Q\bar{\mathbf{e}}(t) + \rho\bar{\mathbf{u}}^{\ell^{T}}(t)R\bar{\mathbf{u}}^{\ell}(t)\,dt \tag{19}$$

and

$$\tilde{J} = \mathfrak{E}\{\lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \tilde{\mathbf{e}}^{T}(t)Q\tilde{\mathbf{e}}(t) + \rho\tilde{\mathbf{u}}^{\ell^{T}}(t)R\tilde{\mathbf{u}}^{\ell}(t)\,dt\} \tag{20}$$

It is assumed that the integration and the expected value operations can be interchanged, to arrive at (18)–(20).

Thus the cost, J, is decomposed into two independent parts $\bar{J}$ and $\tilde{J}$. The equations (9)–(13) and (19) constitute the deterministic problem. The stochastic problem consists of the following equations, obtained by subtracting the corresponding equations (9)–(13) from (1)–(5), and from (6) and (16):

$\tilde{S}^{s}$:

$$\dot{\tilde{\mathbf{x}}}^{s}(t) = A^{s}\tilde{\mathbf{x}}^{s}(t) + B^{s}\tilde{\mathbf{u}}^{s}(t) \tag{21}$$

$$\tilde{\mathbf{y}}^{s}(t) = C^{s}\tilde{\mathbf{x}}^{s}(t) + D^{s}\tilde{\mathbf{u}}^{s}(t) \tag{22}$$

$$\tilde{\mathbf{u}}^{\ell}(t) = C^{\ell}\tilde{\mathbf{x}}^{s}(t) + D^{\ell}\tilde{\mathbf{u}}^{s}(t) \tag{23}$$

$\tilde{S}^{a}$:

$$\dot{\tilde{\mathbf{x}}}^{a}(t) = A^{a}\tilde{\mathbf{x}}^{a}(t) + B^{a}\tilde{\mathbf{u}}^{a}(t) \tag{24}$$

$$\tilde{\mathbf{y}}^{a}(t) = C^{a}\tilde{\mathbf{x}}^{a}(t) + D^{a}\tilde{\mathbf{u}}^{a}(t) \tag{25}$$

and

$$\tilde{\mathbf{e}}(t) = \tilde{\mathbf{y}}^{a}(t) - \tilde{\mathbf{y}}^{s}(t) \tag{26}$$

while minimizing the cost, $\tilde{J}$, (20). Each of these problems involves only its own variables and thus the two subproblems can be solved independently. The overall solution is (Figure 1):

$$\mathbf{u}^{s}(t) = \bar{\mathbf{u}}^{s}(t) + \tilde{\mathbf{u}}^{s}(t). \tag{27}$$

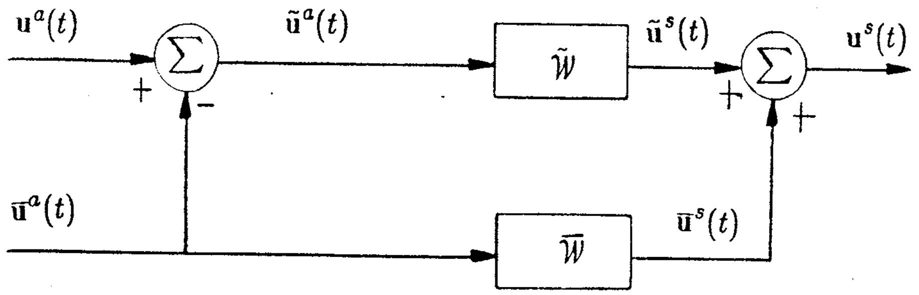

$\mathbf{u}^a(t)$
Σ
+
−
$\tilde{\mathbf{u}}^a(t)$
$\tilde{W}$
$\tilde{\mathbf{u}}^s(t)$
Σ
+
+
$\mathbf{u}^s(t)$
$\overline{\mathbf{u}}^a(t)$
$\overline{W}$
$\overline{\mathbf{u}}^s(t)$

Figure 1. The Combination of the deterministic, $\overline{W}$, and stochastic $\tilde{W}$, solutions for the L.Q. optimal simulator.

This deterministic-stochastic decomposition is based on the L.Q. assumption, but can be extended, in some sense to a nonlinear system with a quadratic cost function, as shown in chapter IV.

3. Deterministic Problem Solution

In this section we find the optimal value of $\overline{\mathbf{u}}^s(t)$ which minimizes $\overline{J}$ (19) for the dynamic system (9)–(13). The solution of the deterministic problem is obtained in a closed analytical form. This enables us to characterizes the properties of this solution, and beyond that to gain insight into the properties of the solution of the stochastic problem which is much harder to obtain. The deterministic problem solution is solved by use of the calculus of variations on the Fourier transform of our original time domain formulation and by the use of Parseval's theorem.

The frequency domain system functions of $\overline{\mathcal{S}}^s$ and $\overline{\mathcal{S}}^a$ are obtained from (1)–(5):

$$\overline{\mathbf{Y}}^s(j\omega) = \overline{S}^s(j\omega)\overline{\mathbf{U}}^s(j\omega) \tag{28}$$
$$\overline{\mathbf{U}}^\ell(j\omega) = \overline{S}^\ell(j\omega)\overline{\mathbf{U}}^s(j\omega) \tag{29}$$
$$\overline{\mathbf{Y}}^a(j\omega) = \overline{S}^a(j\omega)\overline{\mathbf{U}}^a(j\omega) \tag{30}$$

where $\overline{\mathbf{Y}}^s(j\omega)$, $\overline{\mathbf{U}}^s(j\omega)$, $\overline{\mathbf{U}}^\ell(j\omega)$, $\overline{\mathbf{Y}}^a(j\omega)$, $\overline{\mathbf{U}}^a(j\omega)$ denote the *Fourier transform* of each of the time dependent vectors $\overline{\mathbf{y}}^s(t)$, $\overline{\mathbf{u}}^s(t)$, $\overline{\mathbf{u}}^\ell(t)$, $\overline{\mathbf{y}}^a(t)$ and $\overline{\mathbf{u}}^a(t)$ respectively.

The system functions of the simulated, limiting and actual systems are represented by $\overline{S}^s(j\omega)$, $\overline{S}^\ell(j\omega)$ and $\overline{S}^a(j\omega)$ respectively where it is assumed that

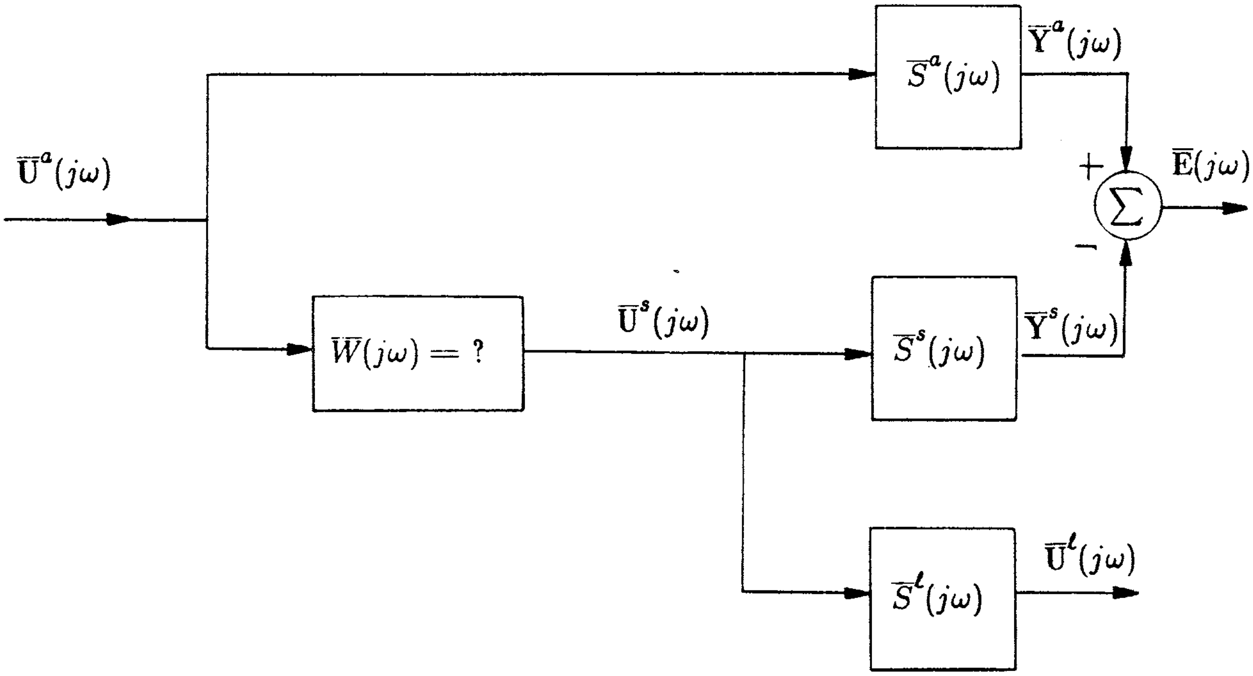

$\overline{\mathbf{U}}^a(j\omega)$
$\overline{S}^a(j\omega)$
$\overline{\mathbf{Y}}^a(j\omega)$
+
Σ
−
$\overline{\mathbf{E}}(j\omega)$
$\overline{W}(j\omega) = ?$
$\overline{\mathbf{U}}^s(j\omega)$
$\overline{S}^s(j\omega)$
$\overline{\mathbf{Y}}^s(j\omega)$
$\overline{S}^\ell(j\omega)$
$\overline{\mathbf{U}}^\ell(j\omega)$

Figure 2. Deterministic problem formulation. The systems $\overline{S}^s(j\omega)$, $\overline{S}^\ell(j\omega)$, $\overline{S}^a(j\omega)$ and $\overline{W}(j\omega)$ correspond respectively to the simulator, limiting, actual and optimal deterministic washout solution.

these Fourier transform exists. The problem formulation including the solution $\overline{W}(j\omega)$ is shown in Figure 2.

Next, Parseval's theorem is used to transform the cost $\overline{J}$ (19) to the frequency domain, assuming that $\overline{J}$ is finite (recall that ρ is real and Q and R are real symmetric matrices):

$$2\pi\overline{J} = \int_{-\infty}^{\infty} \overline{\mathbf{E}}^H(j\omega)Q\overline{\mathbf{E}}(j\omega) + \rho\overline{\mathbf{U}}^{\ell^H}(j\omega)R\overline{\mathbf{U}}^\ell(j\omega)\,dw \tag{31}$$

$$= \int_{-\infty}^{\infty} J_e(j\omega) + J_\ell(j\omega)\,dw \tag{32}$$

where the integrand of $2\pi\overline{J}$ is separated into two parts:

$$J_e(j\omega) \triangleq \overline{\mathbf{E}}^H(j\omega)Q\overline{\mathbf{E}}(j\omega) \tag{33}$$

and

$$J_\ell(j\omega) \triangleq \rho\overline{\mathbf{U}}^{\ell^H}(j\omega)R\overline{\mathbf{U}}^\ell(j\omega) \tag{34}$$

and where the superscript H denotes the conjugate transpose operator. From the definition of $\overline{\mathbf{E}}(j\omega)$, (28), (30) and Figure 2, we obtain:

$$\overline{\mathbf{E}}(j\omega) = \big(S^a(j\omega) - S^s(j\omega)\overline{W}(j\omega)\big)\overline{\mathbf{U}}^a(j\omega) \tag{35}$$

and similarly

$$\overline{\mathbf{U}}^\ell(j\omega) = S^\ell(j\omega)\overline{W}(j\omega)\overline{\mathbf{U}}^a(j\omega)\,. \tag{36}$$

For clarity, the independent variable, $j\omega$, is not printed from here on but only shown in the final answer. Now, substituting (35) and (36) into J_e (33) and J_ℓ (34) respectively, we obtain:

$$J_e = \overline{\mathbf{U}}^{aH} J_e \overline{\mathbf{U}}^a \tag{37}$$
$$J_\ell = \overline{\mathbf{U}}^{aH} J_\ell \overline{\mathbf{U}}^a \tag{38}$$

where

$$J_e \triangleq (S^a - S^s\overline{W})^H Q (S^a - S^s\overline{W}) \tag{39}$$
$$J_\ell \triangleq \rho(S^\ell\overline{W})^H R S^\ell\overline{W}\,. \tag{40}$$

In (39) and (40) let us substitute for $\overline{W}(j\omega)$ the optimal solution $\overline{W}(j\omega)$ plus a perturbation function $\epsilon\eta(j\omega)$, and then take the derivative of J_e and J_ℓ, with respect to the parameter ϵ. This gives the minimum and maximum points (we need the minimum cost $\overline{J}$) by solving the equation when $\epsilon = 0$. Let us compute:

$$\left.\frac{dJ_e}{d\epsilon}\right|_{\epsilon=0} = J_{ee} + J_{ee}^H \tag{41}$$
$$\left.\frac{dJ_\ell}{d\epsilon}\right|_{\epsilon=0} = J_{\ell\ell} + J_{\ell\ell}^H \tag{42}$$

where

$$J_{ee} \triangleq -\eta^H S^{sH} Q (S^a - S^s\overline{W}) \tag{43}$$
$$J_{\ell\ell} \triangleq \rho\eta^H S^{\ell H} R S^\ell \overline{W}\,. \tag{44}$$

Then if we combine J_e and J_ℓ using (32), (37), (38), (41)–(44) we obtain:

$$\left.\frac{dJ_{e+\ell}}{d\epsilon}\right|_{\epsilon=0} = \overline{\mathbf{U}}^{aH}\big((J_{ee} + J_{\ell\ell}) + (J_{ee} + J_{\ell\ell})^H\big)\overline{\mathbf{U}}^a \tag{45}$$

where by (43), (44)

$$J_{ee} + J_{\ell\ell} = \eta^H J_{e+\ell}\,, \tag{46}$$

with

$$J_{e+\ell} \triangleq -S^{a^H}QS^s + \overline{W}^H(S^{s^H}QS^s + \rho S^{\ell^H}RS^\ell)\,. \tag{47}$$

A minimum (or maximum) is obtained for:

$$\left.\frac{dJ_{e+\ell}}{d\epsilon}\right|_{\epsilon=0} = 0 \tag{48}$$

and since $\eta(j\omega)$ is arbitrary, this means that we must have $J_{e+\ell} = 0$. Thus, from (47) the optimal solution for $\overline{W}$ is given by:

$$\overline{W}(j\omega) = \Big(S^{s^H}(j\omega)QS^s(j\omega) + \rho S^{\ell^H}(j\omega)RS^\ell(j\omega)\Big)^{-1}\Big(S^{s^H}(j\omega)QS^a(j\omega)\Big)\,. \tag{49}$$

Note that this solution is in closed analytical form, it is noncausal and by construction is stable. Nevertheless, it is a nontrivial numerical task to calculate $\overline{\mathrm{U}}^s(t)$ from $\overline{W}$ and $\overline{\mathrm{U}}^a(t)$, since in general, $\overline{\mathrm{U}}^a(t)$ has no simple analytic form such that a numerical Fourier transform of $\overline{\mathrm{U}}^a(t)$ can be avoided (numerical convolution usually requires even more computations). One simplification can be done by using a windowed ("smooth" truncation in the time domain) $\overline{W}(t)$ which makes the calculation of $\overline{\mathrm{U}}^s(t)$ practical, using an "overlap-add method" or an "overlap-save method" [Oppenheim75].

4. Stochastic Problem Solution

In the solution of the stochastic problem we use a further assumption: ergodicity i.e. that the ensemble of inputs $\tilde{\mathrm{u}}^a(t) \in \tilde{\mathcal{U}}^a$ can be represented by a *zero mean colored Gaussian noise process.* The solution method is to formulate our problem as a standard Linear Quadratic Gaussian (L.Q.G.) optimal control problem and solve it using state space formulation in the time domain. Recently, frequency domain solutions were developed by Youla [Youla76], but are not used in the current solution, although they could possibly lead to an easier development of some of the solution properties. The additional "Gaussian" assumption used here, means that $\tilde{\mathrm{u}}^a$ is considered in a class of random processes generated by filtering white

Gaussian noise through a linear system $\mathcal{N}$, possibly time varying, i.e.:

$$\mathcal{N}: \quad \begin{aligned} \dot{\tilde{\mathbf{x}}}^n(t) &= A^n(t)\tilde{\mathbf{x}}^n(t) + B^n(t)\mathbf{n}(t) && (50) \\ \tilde{\mathbf{u}}^a(t) &= C^n(t)\tilde{\mathbf{x}}^n(t) && (51) \end{aligned}$$

where the matrices $A^n(t)$, $B^n(t)$, $C^n(t)$ specify the noise shaping filter $\mathcal{N}$, $\tilde{\mathbf{x}}^n$ is the state of this filter, and $\mathbf{n}(t)$ is white Gaussian noise. The optimal stochastic simulator problem thus reduces to the following:

$$\tilde{\mathcal{S}}^s: \quad \begin{aligned} \dot{\tilde{\mathbf{x}}}^s(t) &= A^s\tilde{\mathbf{x}}^s(t) + B^s\tilde{\mathbf{u}}^s(t) && (21) \\ \tilde{\mathbf{y}}^s(t) &= C^s\tilde{\mathbf{x}}^s(t) + D^s\tilde{\mathbf{u}}^s(t) && (22) \\ \tilde{\mathbf{u}}^\ell(t) &= C^\ell\tilde{\mathbf{x}}^s(t) + D^\ell\tilde{\mathbf{u}}^s(t) && (23) \end{aligned}$$

$$\tilde{\mathcal{S}}^a: \quad \begin{aligned} \dot{\tilde{\mathbf{x}}}^a(t) &= A^a\tilde{\mathbf{x}}^a(t) + B^a\tilde{\mathbf{u}}^a(t) && (24) \\ \tilde{\mathbf{y}}^a(t) &= C^a\tilde{\mathbf{x}}^a(t) + D^a\tilde{\mathbf{u}}^a(t) && (25) \end{aligned}$$

Linear Quadratic Gaussian Optimal Simulator Design Problem[1]:

Given the simulator $\tilde{\mathcal{S}}^s$ (21)–(23), the actual system $\tilde{\mathcal{S}}^a$ (24)–(25), and the noise shaping filter $\mathcal{N}$, (50)–(51), find an operator $\tilde{\mathcal{W}}$ which generates $\tilde{\mathbf{u}}^s(t)$, on the basis of $\tilde{\mathbf{u}}^a(\tau), \{\tau\colon -\infty < \tau \le t\}$, for all t, so that $\tilde{J}$ is minimized, where under the ergodicity assumption $\tilde{J}$ (20) reduces to:

$$\tilde{J} = \mathfrak{E}\{\tilde{\mathbf{e}}^T(t)Qe(t) + \rho\tilde{\mathbf{u}}^{\ell^T}(t)R\tilde{\mathbf{u}}^\ell(t)\} \qquad (52)$$

where $\tilde{\mathbf{e}}(t)$ is given by

$$\tilde{\mathbf{e}}(t) = \tilde{\mathbf{y}}^a(t) - \tilde{\mathbf{y}}^s(t)\,. \qquad (26)$$

It is assumed that $\rho > 0$, $D^{\ell^T}RD^\ell$ is a positive definite matrix and Q is a positive semidefinite matrix.

4.1. Solution of the L.Q.G. Optimal Simulator

Let us combine the three linear systems (21)–(23), (24)–(25) and (50)–(51), so as to obtain the following augmented linear system, $\tilde{\mathcal{S}}$,

$$\tilde{\mathcal{S}}: \quad \begin{aligned} \dot{\tilde{\mathbf{x}}}(t) &= \tilde{A}(t)\tilde{\mathbf{x}}(t) + \tilde{B}\tilde{\mathbf{u}}^s(t) + \tilde{H}(t)\mathbf{n}(t) && (53) \\ \tilde{\mathbf{y}}(t) &= \tilde{C}(t)\tilde{\mathbf{x}}(t) + \tilde{D}\tilde{\mathbf{u}}^s(t) && (54) \end{aligned}$$

[1] A less general problem was solved and published by the author [Sivan82].

where

$$\tilde{\mathbf{x}}(t) = \begin{pmatrix} \tilde{\mathbf{x}}^s(t) \\ \tilde{\mathbf{x}}^a(t) \\ \tilde{\mathbf{x}}^n(t) \end{pmatrix}, \quad \tilde{\mathbf{y}}(t) = \begin{pmatrix} \tilde{\mathbf{e}}(t) \\ \tilde{\mathbf{u}}^\ell(t) \end{pmatrix} \tag{55}$$

and

$$\tilde{A}(t) = \begin{pmatrix} A^s & 0 & 0 \\ 0 & A^a & B^a C^n(t) \\ 0 & 0 & A^n(t) \end{pmatrix}, \quad \tilde{B} = \begin{pmatrix} B^s \\ 0 \\ 0 \end{pmatrix}, \quad \tilde{H}(t) = \begin{pmatrix} 0 \\ 0 \\ B^n(t) \end{pmatrix}, \tag{56}$$

$$\tilde{C}(t) = \begin{pmatrix} -C^s & C^a & D^a C^n(t) \\ C^\ell & 0 & 0 \end{pmatrix}, \quad \tilde{D} = \begin{pmatrix} -D^s \\ D^\ell \end{pmatrix}. \tag{57}$$

The optimization criterion $\tilde{J}$ (52) can be written as:

$$\tilde{J} = \mathfrak{E}\{\tilde{\mathbf{x}}^T(t) R_1(t) \tilde{\mathbf{x}}(t) + 2\tilde{\mathbf{x}}^T(t) R_{12}(t) \tilde{\mathbf{u}}^s(t) + \tilde{\mathbf{u}}^{sT}(t) R_2 \tilde{\mathbf{u}}^s(t)\} \tag{58}$$

where

$$R_1(t) = \tilde{C}^T(t) \tilde{Q} \tilde{C}(t) \tag{59}$$

$$R_{12}(t) = \tilde{C}^T(t) \tilde{Q} \tilde{D} \tag{60}$$

$$R_2 = \tilde{D}^T \tilde{Q} \tilde{D} \tag{61}$$

and

$$\tilde{Q} = \begin{pmatrix} Q & 0 \\ 0 & \rho R \end{pmatrix}. \tag{62}$$

The problem of minimizing (58), subject to the system constraint (53)–(54), is a standard stochastic state feedback optimization problem, the solution of which is [Kwakernaak72]

$$\tilde{\mathbf{u}}^s(t) = -F(t) \tilde{\mathbf{x}}(t) \tag{63}$$

where F(t) is given by

$$F(t) = R_2^{-1} \big(\tilde{B}^T P(t) + R_{12}^T(t) \big) \tag{64}$$

and P(t) is the solution of the matrix Riccati equation (condition of existence, see [Kwakernaak72]):

$$\begin{aligned} -\dot{P}(t) = & -P(t) \tilde{B} R_2^{-1} \tilde{B}^T P(t) + P(t) \big(\tilde{A}(t) - \tilde{B} R_2^{-1} R_{12}^T(t) \big) \\ & + \big(\tilde{A}(t) - \tilde{B} R_2^{-1} R_{12}^T(t) \big)^T P(t) + R_1(t) - R_{12}(t) R_2^{-1} R_{12}^T(t) \end{aligned} \tag{65}$$

with the boundary condition

$$P(\infty) = P_\infty \,. \tag{66}$$

In the special case that the noise shaping filter, $\mathcal{N}$, is time-invariant, then our system $\tilde{S}$ (53)-(54) and the cost matrices R_1, R_{12}, R_2 (59)-(61) in (58) are also time-invariant. In the time-invariant case, the steady state optimal gain matrix F (64) is also time-invariant where the steady state P is given by the Algebraic matrix Riccati Equation (ARE):

$$\begin{aligned} 0 = &-P\tilde{B}R_2^{-1}\tilde{B}^T P + P(\tilde{A} - \tilde{B}R_2^{-1}R_{12}^T) + (\tilde{A} - \tilde{B}R_2^{-1}R_{12}^T)^T P \\ &+ R_1 - R_{12}R_2^{-1}R_{12}^T \end{aligned} \tag{67}$$

which is obtained by setting $\dot{P} = 0$ in (65), and where P is the unique positive semidefinite solution to (67).

In the next section we discuss both the time-varying and the time-invariant cases in conjunction with the properties of the optimal simulator solution. Note that the L.Q.G. optimal simulator input $\tilde{\mathbf{u}}^s(t)$ is given in (63) as a function of the whole state vector, whereas we are actually looking for an expression for $\tilde{\mathbf{u}}^s(t)$ as a function of $\tilde{\mathbf{u}}^a$. This difficulty is resolved in the next subsection.

4.2. Derivation of the L.Q.G. Optimal Simulator Filter

In some cases, we would like to have the solution in the form of an optimal filter, i.e. an *Open Loop Optimal System*, rather than the feedback form given in (63). This derivation is especially useful for studying the closed-loop characteristics of the optimal simulation system, as discussed in the next section.

Let us write F in block form, corresponding to the blocks of $\mathbf{x}(t)$ (55),

$$F(t) = (F^s,\ F^a,\ F^n(t)) \tag{68}$$

Note that F^s and F^a are time-invariant (Section 6). From (63) we obtain:

$$\tilde{\mathbf{u}}^s(t) = -F^s\tilde{\mathbf{x}}^s(t) - F^a\tilde{\mathbf{x}}^a(t) - F^n(t)\,\tilde{\mathbf{x}}^n(t)\,. \tag{69}$$

Note that only F^s is a "real" feedback gain, while F^a and F^n are really feed-forward gains. Now assume that $C^n(t)$ is chosen[2] so that there exists a matrix $\mathcal{C}^n(t)^{-1}$ such that,

$$\mathcal{C}^n(t)^{-1}C^n(t) = I \qquad \text{for all } t\,. \tag{70}$$

By using (51) we obtain,

$$\tilde{\mathbf{x}}^n(t) = \mathcal{C}^n(t)^{-1}\tilde{\mathbf{u}}^a(t)\,. \tag{71}$$

Now substituting (71) into (69) we find,

$$\tilde{\mathbf{u}}^s(t) = -F^s\tilde{\mathbf{x}}^s(t) - F^a\tilde{\mathbf{x}}^a(t) - F^n(t)\mathcal{C}^n(t)^{-1}\tilde{\mathbf{u}}^a(t)\,. \tag{72}$$

Using this result (72), we can now regroup the two state equations (21), (24) of systems $\tilde{\mathcal{S}}^s$ and $\tilde{\mathcal{S}}^a$ with the output equation (72) to form the optimal washout filter $\tilde{\mathcal{W}}$,

$$\tilde{\mathcal{W}}: \qquad \begin{aligned} \dot{\tilde{\mathbf{x}}}^{\tilde{w}}(t) &= A^{\tilde{w}}\tilde{\mathbf{x}}^{\tilde{w}}(t) + B^{\tilde{w}}(t)\tilde{\mathbf{u}}^a(t) && (73)\\ \tilde{\mathbf{u}}^s(t) &= C^{\tilde{w}}\tilde{\mathbf{x}}^{\tilde{w}}(t) + D^{\tilde{w}}(t)\tilde{\mathbf{u}}^a(t) && (74) \end{aligned}$$

where

$$\mathbf{x}^{\tilde{w}}(t) \triangleq \begin{pmatrix} \tilde{\mathbf{x}}^s(t) \\ \tilde{\mathbf{x}}^a(t) \end{pmatrix} \tag{75}$$

and

$$A^{\tilde{w}} = \begin{pmatrix} A^s - B^sF^s & -B^sF^a \\ 0 & A^a \end{pmatrix}, \quad B^{\tilde{w}}(t) = \begin{pmatrix} -B^sF^n(t)\mathcal{C}^n(t)^{-1} \\ B^a \end{pmatrix}, \tag{76}$$

$$C^{\tilde{w}} = -(F^s, F^a), \quad D^{\tilde{w}}(t) = -F^n(t)\mathcal{C}^n(t)^{-1}\,. \tag{77}$$

Note that the filter is of order $\dim(A^s)+\dim(A^a)$.

In the case that the noise shaping filter $\mathcal{N}$ is time-invariant, then the gain F_3 is also time-invariant and our overall optimal simulation filter $\tilde{\mathcal{W}}$ is time-invariant, which can also be put in the form of a transfer matrix $\tilde{W}(s)$, where s is the

[2] To make this general, $C^n(t)$ is chosen as follows:
1. Original Dimension($C^n(t)$) > Dimension($\tilde{\mathbf{x}}^n(t)$) ⇒ need only a subset of $\tilde{\mathbf{u}}^a(t)$ to compute $\tilde{\mathbf{x}}^n(t)$.
2. Dimension($C^n(t)$) < Dimension($\tilde{\mathbf{x}}^n(t)$) ⇒ augment $\tilde{\mathbf{u}}^a(t)$ by a dummy vector (adding independent rows to $C^n(t)$) such that the new dimension of $C^n(t)$ = dimension($\tilde{\mathbf{x}}^n(t)$). Further augment B^a and D^a with zero columns so that the overall system $\tilde{\mathcal{S}}$ is not changed.
3. If some states of $\tilde{\mathbf{x}}^n(t)$ are not directly available, an estimation of these states is required, e.g. a Kalman filter.

$\mathcal{L}$-transform *Laplace* variable. Let us use the following notation:

$$\mathcal{L}\{\tilde{\mathbf{x}}^{w}(t)\} \triangleq \tilde{\mathrm{X}}^{w}(s)\,, \quad \mathcal{L}\{\tilde{\mathbf{u}}^{s}(t)\} \triangleq \tilde{\mathrm{U}}^{s}(s)\,, \quad \mathcal{L}\{\tilde{\mathbf{u}}^{a}(t)\} \triangleq \tilde{\mathrm{U}}^{a}(s)\,. \tag{78}$$

Since we deal only with the steady state situation, assuming that $A^{\tilde{w}}$ is stable, we can derive the optimal simulation transfer matrix $\tilde{W}$ from (76) and (77):

$$\tilde{W}(s) = -\Big(I - F^{a}(sI - A^{s} + B^{s}F^{s})^{-1}B^{s}\Big)\Big(F^{a}(sI - A^{a})^{-1}B^{a} + F^{n}\mathcal{C}^{n-1}\Big) \tag{79}$$

where

$$\tilde{\mathrm{U}}^{s}(s) = \tilde{W}(s)\tilde{\mathrm{U}}^{a}(s)\,. \tag{80}$$

Note that the eigenvalues of $A^{\tilde{w}}$ (the poles of the transfer matrix $\tilde{W}(s)$) are the eigenvalues of the matrix $(A^{s} - B^{s}F^{s})$ and those of the matrix A^{a}.

5. Properties of the time-invariant L.Q.G. Solution

The number of states in $\tilde{\mathcal{S}}$ is prohibitively large, even for the lowest possible dimension modeling of $\mathcal{N}$, $\mathcal{V}$ and $\mathcal{M}$ ((6 states for the pilot + 12 for the airplane) states for $\mathcal{N}$,6 states for $\mathcal{V}$, 18 for $\mathcal{M}$ = total of 42 states), it is important to be able to solve numerically a smaller dimension ARE. Beyond that, in order to best handle the design of such a large system we need to learn some of the solution's general properties that reflect the problem structure. Such solution properties were found and enable us to find the ARE solution by solving several much smaller matrix equations, where the maximum dimension ARE that has to be solved has only 8 states. It also requires the solution of an 18 × 8 *Sylvester* equation (a set of linear equations with a special structure, for 18 × 8 = 144 unknowns). An even more important consequence is that the solution decouples in such a way that only two physical dimensions are coupled at the same time. This enables us to implement the optimal washout in easy way.

Another important observation is that under plausible conditions of constructing the relative cost matrices, the poles of the optimal washout are independent of $\mathcal{N}$, or in other words they do not depend on the details of the class of airplane maneuvers that are simulated, but depend only on a gross measure of their amplitude. Furthermore the "motion" of the poles with a change in the amplitude of the maneuvers can be analyzed and "predicted". These two observations enable the comparison in chapter VI of the optimal washout to other experimentally optimized washouts and furthermore confirm the design rules [Sinacori77] used in the "field" to set the initial values of the *washout* parameters by simulator experimentalists who have the engineering knowledge for setting those parameters. Several other properties of the OWS which are useful in the implementation are also derived.

5.1. Algebraic Structure of the Algebraic Riccati Equation

Let us partition the matrix P of (67) into blocks which correspond to the blocks of $\tilde{x}(t)$ (55). Furthermore, it can be seen that P, the solution to the Algebraic Riccati Equation, has to be symmetric, so that we can write it as:

$$P = \begin{pmatrix} P^s & P^a & P^n \\ P^{aT} & P^{a,a} & P^{a,n} \\ P^{nT} & P^{a,nT} & P^{n,n} \end{pmatrix} \tag{81}$$

where P^s, $P^{a,a}$ and $P^{n,n}$ are square symmetric matrices of the same dimensions as the matrices A^s, A^a and A^n respectively. Similarly let us partition the matrices $\tilde{A}$, $\tilde{B}$, $\tilde{C}$, $\tilde{D}$, R_1, R_2 and $R_{1,2}$, so that we obtain the following six matrix equations derived in appendix A using MACSYMA, [MACSYMA77] (note that in the following equations the superscripts s,a,n and ℓ of the system matrices A, B, C and D are subscripts and the symbol rho is used for ρ):

Block 1 , 1

$$\begin{aligned}
&(A_s^T - C_s^T . Q . D_s . R_2^{<-1>} . B_s^T - \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T)) . P_s \\
&+ P_s .(A_s - B_s . R_2^{<-1>} . D_s^T . Q . C_s - \mathrm{rho}\ (B_s . R_2^{<-1>} . D_1^T . R . C_1)) \\
&- P_s . B_s . R_2^{<-1>} . B_s^T . P_s - C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . C_s \\
&- \mathrm{rho}\ (C_s^T . Q . D_s . R_2^{<-1>} . D_1^T . R . C_1) + C_s^T . Q . C_s \\
&- \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . C_s) \\
&- \mathrm{rho}^2\ (C_1^T . R . D_1 . R_2^{<-1>} . D_1^T . R . C_1) + \mathrm{rho}\ (C_1^T . R . C_1) = 0
\end{aligned} \tag{82}$$

Block 1 , 2

$$\begin{aligned}
&(A_s^T - C_s^T . Q . D_s . R_2^{<-1>} . B_s^T - \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T)) . P_a \\
&+ P_s . B_s . R_2^{<-1>} . D_s^T . Q . C_a - P_s . B_s . R_2^{<-1>} . B_s^T . P_a \\
&+ C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . C_a - C_s^T . Q . C_a \\
&+ \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . C_a) + P_a . A_a = 0
\end{aligned} \tag{83}$$

Block 1 , 3

$$\begin{aligned}
&(A_s^T - C_s^T . Q . D_s . R_2^{<-1>} . B_s^T - \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T)) . P_n \\
&+ P_s . B_s . R_2^{<-1>} . D_s^T . Q . D_a . C_n \\
&- P_s . B_s . R_2^{<-1>} . B_s^T . P_n + C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . D_a . C_n \\
&- C_s^T . Q . D_a . C_n + P_n . A_n \\
&+ \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . D_a . C_n) + P_a . B_a . C_n = 0
\end{aligned} \tag{84}$$

Block 2 , 2

$$P_{a,\ a} \, . \, \bar{A}_a + P_a^T \, . \, B_s \, . \, R_2^{<-\ 1>} \, . \, D_s^T \, . \, Q \, . \, C_a$$

$$- P_a^T \, . \, B_s \, . \, R_2^{<-\ 1>} \, . \, B_s^T \, . \, P_a - C_a^T \, . \, Q \, . \, D_s \, . \, R_2^{<-\ 1>} \, . \, D_s^T \, . \, Q \, . \, C_a$$

$$+ C_a^T \, . \, Q \, . \, D_s \, . \, R_2^{<-\ 1>} \, . \, B_s^T \, . \, P_a + C_a^T \, . \, Q \, . \, C_a + A_a^T \, . \, P_{a,\ a} = 0 \qquad (85)$$

Block 2 , 3

$$P_{a,\ n} \, . \, A_n + P_{a,\ a} \, . \, B_a \, . \, C_n + P_a^T \, . \, B_s \, . \, R_2^{<-\ 1>} \, . \, D_s^T \, . \, Q \, . \, D_a \, . \, C_n$$

$$- P_a^T \, . \, B_s \, . \, R_2^{<-\ 1>} \, . \, B_s^T \, . \, P_n - C_a^T \, . \, Q \, . \, D_s \, . \, R_2^{<-\ 1>} \, . \, D_s^T \, . \, Q \, . \, D_a \, . \, C_n$$

$$+ C_a^T \, . \, Q \, . \, D_s \, . \, R_2^{<-\ 1>} \, . \, B_s^T \, . \, P_n + C_a^T \, . \, Q \, . \, D_a \, . \, C_n + A_a^T \, . \, P_{a,\ n} = 0 \qquad (86)$$

Block 3 , 3

$$P_{n,\ n} \, . \, A_n + P_n^T \, . \, B_s \, . \, R_2^{<-\ 1>} \, . \, D_s^T \, . \, Q \, . \, D_a \, . \, C_n$$

$$- P_n^T \, . \, B_s \, . \, R_2^{<-\ 1>} . \, B_s^T \, . \, P_n - C_n^T \, . \, D_a^T \, . \, Q \, . \, D_s \, . \, R_2^{<-\ 1>} . \, D_s^T \, . \, Q \, . \, D_a \, . \, C_n$$

$$+ C_n^T \, . \, D_a^T \, . \, Q \, . \, D_s \, . \, R_2^{<-\ 1>} \, . \, B_s^T \, . \, P_n$$

$$+ C_n^T \, . \, D_a^T \, . \, Q \, . \, D_a \, . \, C_n + C_n^T \, . \, B_a^T \, . \, P_{a,\ n} + A_n^T \, . \, P_{n,\ n}$$

$$+ P_{a,\ n}^T \, . \, B_a \, . \, C_n = 0 \qquad (87)$$

Let us analyze these cumbersome matrix equations in order to gain some insight into the role of the design parameters in the Optimal Washout System and simplify its numerical computation. Looking carefully at the matrix equations (82)-(87) we can make the following observations:

(i) *Block* 1,1 has only one matrix unknown: P^s.

(ii) The equations can be solved one at a time if solved in the following order: *block* 1,1, *block* 1,2, *block* 1,3, *block* 2,2, *block* 2,3 and *block* 3,3.

(iii) If the equations are solved in the above order then only the equation of *block* 1,1 is quadratic, all the other equations are linear in the unknown matrices P^i and are of the form:

$$P^i G^i + H^i P^i = K^i \tag{88}$$

for $i = \{(a), (n), (a,a), (a,n), (n,n)\}$

(iv) The equation in *block* 1,1 is independent of the parameters of the systems $\tilde{\mathcal{S}}^a$ and $\mathcal{N}$ and thus the solution for P^s depends only on A^s, B^s, C^s, D^s, C^ℓ, D^ℓ, Q, R, and ρ.

(v) The quadratic equation in *block* 1,1 is an Algebraic Riccati Equation with a similar form as the one we started with (67).

(vi) The equations in *blocks* 1,1, 1,2 and 2,2 are independent of the parameters of the system $\mathcal{N}$ (representing the class of input signals) and the parameters, B^a and D^a, of the system $\tilde{\mathcal{S}}^a$.

(vii) All the equations are independent of B^n.

(viii) The equations in *blocks* 1,2, 1,3 can be simplified to (93) and (94) where their parameters are given in (95)-(99). Note that the dependence of the equations on the parameters of system $\tilde{\mathcal{S}}^a$ (A^a and C^a, B^a and D^a) is shown explicitly for P^a in (93) and for P^n in $\mathcal{K}$ (98). Similarly the dependence of P^n on the parameters of the system $\mathcal{N}$ (A^n and C^n) is shown explicitly in (94).

Returning now to our problem, we can evaluate the following *block* partitioned optimal gains from (64) and (68):

$$F^s = R_2^{-1}(B^{s^T}P^s + D^{s^T}QC^s + \rho D^{\ell^T}RC^\ell) \tag{89}$$
$$F^a = R_2^{-1}(B^{s^T}P^a - D^{s^T}QC^a) \tag{90}$$
$$F^n = R_2^{-1}(B^{s^T}P^n - D^{s^T}QD^aC^n). \tag{91}$$

In order to find the required optimal gains, F^s, F^a and F^n, we need only solutions for three *block* matrices: P^s, P^a and P^n. The solutions for these matrices can be obtained from the following simplified equations corresponding to *blocks* 1,1, 1,2 and 1,3, derived from (82)-(84):

Block 1,1

$$-P^s B^s R_2^{-1} B^{s^T} P^s + P^s \mathcal{A}^s + \mathcal{A}^{s^T} P^s + \mathcal{Q} = 0 \tag{92}$$

Block 1,2

$$P^a A^a + (\mathcal{A}^s - B^s R_2^{-1} B^{s^T} P^s)^T P^a + \mathcal{H} C^a = 0 \tag{93}$$

Block 1,3

$$P^n A^n + (\mathcal{A}^s - B^s R_2^{-1} B^{s^T} P^s)^T P^n + \mathcal{K} C^n = 0 \tag{94}$$

where

$$\mathcal{A}^s = A^s - B^s R_2^{-1}(D^{s^T} Q C^s + \rho D^{\ell^T} R C^\ell) \tag{95}$$

$$\begin{aligned}\mathcal{Q} = {} & C^{s^T} Q C^s - C^{s^T} Q D^s R_2^{-1} D^{s^T} Q C^s + \rho C^{\ell^T} R C^\ell - \rho C^{\ell^T} R D^\ell R_2^{-1} D^{s^T} Q C^s \\ & - \rho (C^{\ell^T} R D^\ell R_2^{-1} D^{s^T} Q C^s)^T - \rho^2 C^{\ell^T} R D^\ell R_2^{-1} D^{\ell^T} R C^\ell\end{aligned} \tag{96}$$

$$\mathcal{H} = -C^{s^T} Q + (P^s B^s + C^{s^T} Q D^s + \rho C^\ell R D^\ell) R_2^{-1} D^{s^T} Q \tag{97}$$

$$\mathcal{K} = P^a B^a + \mathcal{K}_s D^a \tag{98}$$

$$\mathcal{K}_s = (-C^{s^T} + (P^s B^s + C^{s^T} Q D^s + \rho C^{\ell^T} R D^\ell) R_2^{-1} D^{s^T}) Q\,. \tag{99}$$

These equations are presented in a way that shows their algebraic structure and dependence on "elementary" design matrices, as discussed in our observations before. From observations (iv) and (vi), we gather that only F^n depends on the parameters of the system $\mathcal{N}$ (it also depends on parameters B^a and D^a of system $\tilde{S}^a$). Furthermore F^s depends only on A^s, B^s, C^s, D^s, C^ℓ, D^ℓ, Q, R and ρ. Therefore, from this and from (76) it can be concluded that the eigenvalues of the washout matrix, $A^{\tilde{w}}$, or the poles of the washout transfer matrix $\tilde{W}(s)$ in (79), are independent of the parameters of the system $\mathcal{N}$. Thus, for a given set of systems $\tilde{S}^s$ and $\tilde{S}^a$, the only parameters that "control" the placement of the eigenvalues (**poles**) in our solution, are the weighting matrices Q and R, and the scalar ρ. In the discussion later on, we reason for a particular choice of matrices Q and R for some given simulator limitations, so that the only remaining design parameter which effects the placement of the eigenvalues (poles) is the scalar ρ. Now recall that in our approach, the parameters of the system $\mathcal{N}$, represent the class of inputs for which the OWS is designed. Thus, changing the system $\mathcal{N}$ changes only the zeros of the washout filter transfer matrix $\tilde{W}(s)$ and has only a second order effect on the poles through the adjustment of the scalar ρ. Furthermore, a change of $\mathcal{N}$ affects only F^n (91), which uses the solution P^n. From our previous observation (viii), we know that P^n is a solution of a linear equation (94), which is "relatively" simple to solve. All this may allow us, if we wish, to design an "Adapting Optimal Washout System", which will adapt to the "current" class of input signals $\tilde{\mathbf{u}}^a(t)$. This is a reasonable approximation for the solution for a very "slow" time-varying system $\mathcal{N}$. An exact solution for a problem formulation that includes a time-varying system $\mathcal{N}$ is developed in the next section, 6.

Another property that can be inferred, is that scaling of all the inputs to the *"actual"* system $\tilde{S}^a$ by a scalar α (i.e. substituting αB^a and αD^a for B^a and D^a respectively), we obtain an Optimal Washout System which is linear in α (i.e. $\tilde{W}(s)$ is α times our previous $\tilde{W}(s)$).

Proof: From observation (viii) equations (94) and (98) we conclude that the solution, P^n, of (94) is linear in α. By substituting this solution into (91) we find that F^n is also linear in α. Finally substituting F^n into (79) we observe that $\tilde{W}(s)$ is also linear in α ▮

The Optimal Washout System $\tilde{W}(s)$ is invariant to scaling the matrix C^n by a scalar α.

Proof: From (94), we see that P^n is linear in α, thus from (91), F^n is linear in α too. Therefore from (76)-(77) (or (79)), we find that $\tilde{W}(s)$ is invariant to the value of α. This is an obvious result due to the linear nature of our problem formulation. In chapter IV a superior nonlinear washout is developed which does not have this property i.e. depends on the input amplitude $\mathbf{u}^a(t)$ ▮

In the next section further simplifications of the "solution" (design) are performed by reducing the problem to a sequential solution of a series of smaller dimension problems.

5.2. Dimension Reduction

In this subsection, we explore the conditions under which the equations (92)-(94) for P^s, P^a and P^n can be further subdivided into sub-blocks which can be solved separately or sequentially. Beyond the gains in the computation, this subdivision enables us to further understand the structure of the Optimal Washout System and take advantage of this structure in its implementation. First, *block* 1,1 (92) is considered; it is also an Algebraic Riccati Equation of the same form as our original Algebraic Riccati Equation (67). Next we look at the linear equation (94) (for P^n) and skip a discussion about equation (93) (for P^a) since P^a is a sub-block of P^s (in the class of problems we have interest in) and these two equations are very similar. Finally, we will discuss the coupling between the physical dimension groups which arises from our further subdivision of the problem.

5.2.1. Riccati equation (P^s =?)

The optimal feedback gains F^s are computed from (89) using the solution for P^s, given by (100) (a rewritten version of (92)):

$$F^s = R_2^{-1}(B^{s^T}P^s + D^{s^T}QC^s + \rho D^{\ell^T}RC^\ell) \tag{89}$$

$$-P^s\mathcal{B}P^s + P^s\mathcal{A}^s + \mathcal{A}^{s^T}P^s + \mathcal{Q} = 0 \tag{100}$$

where

$$\mathcal{B} = B^s R_2^{-1} B^{s^T} \tag{101}$$

and $\mathcal{A}^s$ and $\mathcal{Q}$ are defined in (95) and (96) respectively. In the special case, where the square matrices $\mathcal{A}^s$, $\mathcal{B}$ and $\mathcal{Q}$ are block diagonal of the form:

$$G = \begin{pmatrix} G^1 & 0 & 0\,0 \ldots & 0 \\ 0 & G^2 & 0\,0 \ldots & 0 \\ 0 & 0 & \cdot\, 0 \ldots & 0 \\ \vdots & \vdots & \cdot & \vdots \\ 0 & 0 & 0\,0 \ldots & G^m \end{pmatrix} \tag{102}$$

and they all have the same number of square blocks m with dimension m_i; then the solution for the matrix P^s also has block diagonal form, with the corresponding block dimension. This can be seen by plugging in a matrix P^s, which has block variables instead of the zero blocks; then computing these blocks by substituting the block structure of $\mathcal{A}^s$, $\mathcal{B}$ and $\mathcal{Q}$ into (100) and expanding the matrix equation into its block equations.

This special case is of interest because it occurs in our flight simulator motion design problem. In the L.Q. formulation of a cascaded motion-base system we obtain four "independent" diagonal blocks in the system $\tilde{\mathcal{S}}^s$, which correspond to the following four physical dimension groups[3]:

(i) *Longitudinal*: *Surge* linear and *Pitch* angular.
(ii) *Lateral*: *Sway* linear and *Roll* angular.
(iii) *Heave* linear.
(iv) *Yaw* angular.

These groups correspond to a block diagonal structure of the matrices A^s, B^s, C^s, D^s, C^ℓ, D^ℓ, Q and R; which are shown to generate a corresponding square block diagonal structure in the matrices $\mathcal{A}^s$, $\mathcal{B}$ and $\mathcal{Q}$. This is why the designers of *washout systems* for *cascaded motion-bases* can get away with designs that consider these four physical dimension groups independently[4].

Recall R_2's definition (61) and then expand it to obtain:

$$R_2 = D^{s^T} Q D^s + \rho D^{\ell^T} R D^\ell . \tag{103}$$

Thus, dim(R_2)= number columns(D^s)= number columns(D^ℓ)= number of control inputs in $\tilde{\mathbf{u}}^s$; Recall that it was required that $D^{\ell^T} R D^\ell > 0$. By plugging in the block diagonal structure of the matrices D^s, Q, D^ℓ and R into (103) and expanding, we find that R_2 has also a block diagonal structure with square blocks of dimension equal to the number of control inputs to each block = number columns(D^{s^i}) = number columns (D^{ℓ^i}), that is when the following dimension equalities hold:

$$\text{number rows}(D^{s^i}) = \text{number rows}(Q) \tag{104}$$

[3] These four groups are fundamental to an L.Q. formulation. In more complex motion-base systems, the motion-base dynamics and/or limitations may cause some coupling between these dimension groups.

[4] It is shown in the next subsection that this result does not necessarily hold for the feed-forward gains F^n, in the case of sophisticated modeling of the anticipated airplane motions. More details are discussed in the implementation Chapter VI, in the examples of chapter V.

$$\text{number rows}(D^{\ell^i}) = \text{number rows}(R). \tag{105}$$

Thus the number of diagonal blocks in R_2 is m, and each diagonal block has dimension:

$$m_i^{R_2} = \text{number control input of block } i. \tag{106}$$

It follows that R_2^{-1} has the same block diagonal structure and dimensions of R_2, which further matches that of the columns of B^s. Thus $\mathcal{B}$ (101) is also block diagonal with m diagonal blocks of dimension m_i, where:

$$m_i = \text{number rows}(\mathcal{B}^i) = \text{number rows}(B^{si}) \tag{107}$$

and

$$m_i = \text{number rows}(A^{si}). \tag{108}$$

Recall equation (95) for $\mathcal{A}^s$ which can verify that $\mathcal{A}^s$ has the same square block diagonal structure as A^s. Finally, one can verify by direct substitution and expansion of (96), that $\mathcal{Q}$ has also the same square block diagonal structure as $\mathcal{A}^s$ and $\mathcal{B}$. Now from (100), it can be verified that P^s also has the same square block diagonal structure with the same m and m_i's and each P^{si} can be solved independently from its own Algebraic Riccati Equation (similar to (100)):

$$-P^{si}\mathcal{B}^iP^{si} + P^{si}\mathcal{A}^{si} + \mathcal{A}^{si^T}P^{si} + \mathcal{Q}^i = 0 \;\blacksquare \tag{109}$$

Furthermore, from (89), F^s has a block diagonal structure with the following dimensions:

$$m^{F^s} = m \tag{110}$$

$$\text{number columns}(F^{si}) = m_i \tag{111}$$

$$\text{number rows}(F^{si}) = \text{number control inputs in block } i \tag{112}$$

and block F^{si} can be computed using only the i^{th} blocks as follows:

$$F^{si} = (R_2^{-1})^i(B^{si^T}P^{si} - C^{si^T}Q^iD^{si}). \tag{113}$$

In summary, if the system $\tilde{\mathcal{S}}^s$ has m diagonal blocks with correctly matching dimensions, then the Algebraic Riccati Equation can be separated into m smaller Algebraic Riccati Equation of the same form which can be solved independently. Furthermore the feedback optimal gains, F^s, also separate into m diagonal blocks (not necessarily square) corresponding to the original block diagonal structure of the system $\tilde{\mathcal{S}}^s$and thus we end up with m independent parallel closed-loop systems. This result does not necessarily hold for the feed-forward gains, as discussed in the next subsection.

5.2.2. Linear equation ($P^n =?$)

The feed-forward gains, F^n (91), use the solution of the matrix block P^n computed from (114) (a rewritten version of (94)).

$$F^n = R_2^{-1}(B^{s^T}P^n - D^{s^T}QD^aC^n) \tag{91}$$

$$P^nA^n + \mathcal{A}^TP^n + \mathcal{C} = 0 \tag{114}$$

$$\mathcal{A} = \mathcal{A}^s - B^sR_2^{-1}B^{s^T}P^s \tag{115}$$
$$\mathcal{A}^s = A^s - B^sR_2^{-1}(D^{s^T}QC^s + \rho D^{\ell^T}RC^\ell) \tag{95}$$
$$\mathcal{C} = \mathcal{K}C^n \tag{116}$$
$$\mathcal{K} = P^aB^a + \mathcal{K}^sD^a \tag{98}$$
$$\mathcal{K}^s = (-C^{s^T} + (P^sB^s + C^{s^T}QD^s + \rho C^{\ell^T}RD^\ell)R_2^{-1}D^{s^T})Q. \tag{99}$$

Equation (114) for P^n is linear and is known as a *Sylvester equation.* There are three interesting cases in which this equation can be partitioned into blocks that can be solved sequentially or independently:

(i) Diagonal—matrices A^n and $\mathcal{A}$ are block diagonal.

(ii) Block triangular—matrices A^n or $\mathcal{A}$ are upper (lower) block triangular matrices.

(iii) Triangular—matrices A^n or $\mathcal{A}$ are upper (lower) triangular matrices.

The *diagonal* case is a special case of the *triangular* case and is the most frequently encountered in practice. Note that $\mathcal{C}$ does not need to have any special structure. Such a general $\mathcal{C}$ results from a general C^n (116), which arises when modeling the expected input to the simulated system $\tilde{\mathrm{u}}^a$ with common modes to several physical dimension groups. An example is the coupling between the roll and yaw motions of an airplane; turning the airplane (yawing) is achieved by banking (rolling) the airplane, thus using its lift to perform the desired turn. This coupling is manifested by a gain matrix F^n which is not block diagonal and has coupling gains between the four physical dimension groups discussed before. Let us first show the equation separation achieved in the *diagonal* case and then in the more general *Triangular* case.

Diagonal case

Let matrices A^n and $\mathcal{A}$ of (114) be block diagonal of the form given in (102):

$$A^n = \begin{pmatrix} A^{n1} & 0 & 0\,0\ldots & 0 \\ 0 & A^{n2} & 0\,0\ldots & 0 \\ 0 & 0 & \cdot\,0\ldots & 0 \\ \vdots & \vdots & . & \vdots \\ 0 & 0 & 0\,0\ldots & A^{nm^{A^n}} \end{pmatrix}, \tag{117}$$

$$\mathcal{A} = \begin{pmatrix} \mathcal{A}^1 & 0 & 0\,0\ldots & 0 \\ 0 & \mathcal{A}^2 & 0\,0\ldots & 0 \\ 0 & 0 & \cdot\,0\ldots & 0 \\ \vdots & \vdots & . & \vdots \\ 0 & 0 & 0\,0\ldots & \mathcal{A}^{m^{\mathcal{A}}} \end{pmatrix}. \tag{118}$$

A^n has m^{A^n} diagonal blocks which correspond to the number of dimension groups in the model $\mathcal{N}$ that generates the expected class of inputs $\tilde{\mathrm{u}}^a$. Similarly $\mathcal{A}$ has $m^{\mathcal{A}}$ diagonal blocks which correspond to the number of dimension groups in the system $\tilde{\mathcal{S}}^s$. In general, $m^{A^n} \neq m^{\mathcal{A}}$. According to these definitions, the matrix P^n and $\mathcal{C}$ have $m^{\mathcal{A}}$ rows and m^{A^n} columns of block matrices, that is:

$$P^n = \begin{pmatrix} P^{n1,1} & \ldots & P^{n1,m^{A^n}} \\ \vdots & & \vdots \\ P^{nm^{\mathcal{A}},1} & \ldots & P^{nm^{\mathcal{A}},m^{A^n}} \end{pmatrix}, \quad \mathcal{C} = \begin{pmatrix} \mathcal{C}^{1,1} & \ldots & \mathcal{C}^{1,m^{A^n}} \\ \vdots & & \vdots \\ \mathcal{C}^{m^{\mathcal{A}},1} & \ldots & \mathcal{C}^{m^{\mathcal{A}},m^{A^n}} \end{pmatrix}. \tag{119}$$

By direct substitution of the special block structure of A^n and $\mathcal{A}$ (117)-(118), and a general block structure for $\mathcal{C}$ and the unknown matrix P^n (119), we obtain

$m^{A^n} \times m^{\mathcal{A}}$ matrix equations of the form:

$$P^{ni,j}A^{nj} + \mathcal{A}^{i^T}P^{ni,j} = \mathcal{C}^{i,j} \tag{120}$$

$$\text{for:} \qquad 1 \leq i \leq m^{\mathcal{A}} \text{ and } 1 \leq j \leq m^{A^n}$$

which can be solved independently (a formal proof can be done using induction). ∎

Use of this result (120) clearly requires much less computation to find the solution P^n (this can be a few orders of magnitude for the flight simulator motion design problem). Furthermore, this result shows where coupling between dimension groups in the feed-forward gains comes from and which design parameters influence its existence.

Triangular case

This case is solved by recursion. Let us partition the matrix $\mathcal{A}$ as shown in (121). The result of partitioning matrix A^n is the same when we consider the transpose of (114). Let us assume that $\mathcal{A}$ is a block triangular matrix which can be partitioned into the following four blocks (not a unique partition), where block 2,1 = 0 matrix:

$$\mathcal{A} = \begin{pmatrix} \mathcal{A}^1 & 0 \\ \mathcal{A}^3 & \mathcal{A}^2 \end{pmatrix} \tag{121}$$

where blocks $\mathcal{A}^1$ and $\mathcal{A}^2$ are square. The dimension of $\mathcal{A}^1$ is $m_1^{\mathcal{A}}$. Similarly $\mathcal{C}$ and P^n are partitioned into two general blocks as follows:

$$\mathcal{C} = \begin{pmatrix} \mathcal{C}^1 \\ \mathcal{C}^2 \end{pmatrix}, \qquad P^n = \begin{pmatrix} P^{n1} \\ P^{n2} \end{pmatrix}. \tag{122}$$

where their dimensions should be:

$$\text{number rows}(\mathcal{C}^1) = \text{ number rows}(P^{n1}) = m^{\mathcal{A}} \tag{123}$$

$$\text{number columns}(\mathcal{C}^1) = \text{ number columns}(P^{n1}) = m^{A^n}. \tag{124}$$

Plugging these partitions into (114), we obtain the following two matrix equations (they are written in the order they should be solved in):

Block 2

$$P^{n2}A^n + \mathcal{A}^{2^T}P^{n2} = -\mathcal{C}^2 \tag{125}$$

Block 1

$$P^{n1}A^n + \mathcal{A}^{1^T}P^{n1} = -\mathcal{C}^{1'} \triangleq -(\mathcal{C}^1 - \mathcal{A}^{3^T}P^{n2}) \tag{126}$$

If one or more of the matrices A^n, $\mathcal{A}^1$, $\mathcal{A}^2$, is block triangular, then we can apply the above partitioning again and thus solve simultaneously a smaller set of linear equations. This procedure can be applied recursively as many times as possible to obtain a smaller and smaller set of equations to be solved simultaneously. ▮

Note that the *diagonal case* is just a special case where $\mathcal{A}^3 = 0$ in (121), (126) and it can also be solved recursively.

Special triangular case

For any system $\mathcal{N}$, A^n can be put into the following triangular matrix form (a Jordan canonical form is even more restrictive):

$$A^n = \begin{pmatrix} a^n_{1,1} & a^n_{1,2} & \dots & a^n_{1,n^{\mathcal{N}}} \\ 0 & a^n_{2,2} & \dots & a^n_{2,n^{\mathcal{N}}} \\ \vdots & & . & \vdots \\ 0 & 0 & \dots & a^n_{n^{\mathcal{N}},n^{\mathcal{N}}} \end{pmatrix}. \tag{127}$$

Let us consider the partition of matrices P^n and $\mathcal{C}$ into $n^{\mathcal{N}}$ column vectors:

$$P^n = (\mathbf{p}^n_1 \dots \mathbf{p}^n_{n^{\mathcal{N}}}) \tag{128}$$

$$\mathcal{C} = (\mathbf{c}_1 \dots \mathbf{c}_{n^{\mathcal{N}}}) \tag{129}$$

then the solution for block 1,1 of (114) is given by:

$$\mathbf{p}^n_1 a^n_{1,1} + \mathcal{A}^T\mathbf{p}^n_1 = -\mathbf{c}_1 \tag{130}$$

which can be rewritten as:

$$(a^n_{1,1}I + \mathcal{A}^T)\mathbf{p}^n_1 = -\mathbf{c}_1 \tag{131}$$

where I is the identity matrix and $a_{1,1}^n$ is a scalar. If the matrix $(a_{1,1}^n I + \mathcal{A}^T)$ is invertible then the solution for the vector $\mathbf{p}_1^n$ exists[5] and is given by:

$$\mathbf{p}_1^n = -(a_{1,1}^n I + \mathcal{A}^T)^{-1}\mathbf{c}_1 \,. \tag{132}$$

In general, we have:

$$\mathbf{p}_i^n = -\big((a_{i,i}^n I + \mathcal{A})^{-1}\big)^T \mathbf{c}'_i \tag{133}$$

where the order of the matrix transpose and inverse where interchanged, and $\mathbf{c}'_i$ depends on the previous $\mathbf{p}_j^n$ $(j < i)$ through:

$$\mathbf{c}'_i = \mathbf{c}_i + \sum_{j=1}^{i-1} a_{j,i}^n \, \mathbf{p}_j^n \tag{134}$$

Finally, using (91) we can find the solution for the feed-forward vectors of gains $\mathbf{f}_i^n$ as:

$$\mathbf{f}_i^n = R_2^{-1}(B^{s^T}\mathbf{p}_i^n - D^{s^T} Q D^a \mathbf{c}_i^n) \tag{135}$$

where

$$F^n = (\mathbf{f}_1^n \ldots \mathbf{f}_{n\mathcal{N}}^n) \tag{136}$$

$$C^n = (\mathbf{c}_1^n \ldots \mathbf{d}_{n\mathcal{N}}^n) \tag{137}$$

Recall from (115) and (95) that $\mathcal{A}$ depends only on the parameters of system $\tilde{\mathcal{S}}^s$ and the ARE solution for that system P^s (92), (95), (96) (superscripts s and ℓ), therefore we can use the solution to the following general inverse matrix, $\mathcal{A}^\alpha(\alpha)^{-1}$, as a function of a single variable, α,

$$\mathcal{A}^\alpha(\alpha)^{-1} = (\alpha I + \mathcal{A})^{-1} \tag{138}$$

and use this to find the required inverses, by evaluating $\mathcal{A}^\alpha(\alpha = a_{i,i}^n)$ for our given $a_{i,i}^n$'s. This provides us with an algorithm to compute the "adaptive" feed-forward

[5] if $\mathcal{A}$ does not have any eigenvalues $= -a_{1,1}^n$.

gains F^n in real time, which changes according to "slow" changes in the system $\mathcal{N}$. "Slow" here refers to the approximation $\dot{\mathbf{p}}_i^n \approx 0$. A more complex solution which does not include this "slow" approximation is derived in the next section.

A solution $\mathbf{p}_i^n$ exists if the inverse matrix of (138) exists for all $\alpha = a_{i,i}^n$, which exist if and only if the

$$\text{eigenvalues}(\mathcal{A}) \neq -a_{i,i}^n\,. \tag{139}$$

This condition is usually satisfied, for the following reasons. The eigenvalues of $\mathcal{A}$ are always negative, since $\mathcal{A}$ is the system matrix of the closed-loop system, A^s (79), (89), (115), (95), which is asymptotically stable, due to our optimal control design method. Next, the system $\mathcal{N}$ is <u>usually</u> modeled as a stable system, which implies that the eigenvalues of $\mathcal{N}$, $a_{i,i}^n < 0$. Thus, condition (139) is always satisfied. Note that if the design includes an unstable system $\mathcal{N}$, with eigenvalues which only approximately violate (139), then severe problems are expected in implementation, since even a small perturbation, ϵ, from the eigenvalues of $\mathcal{A}$ or A^n will cause a discontinuity of $\mathbf{p}_i^n(\epsilon)$ from $+\infty$ to $-\infty$, or vice versa.

5.3. Simulation Filter Simplification in the Symmetric Case

In this case, the *actual* and the *simulated* systems ($\mathcal{S}^a$ and $\mathcal{S}^s$) have the same dynamics (but not the same constraints!), so that:

$$A^a = A^s \triangleq A\,, \quad B^a = B^s \triangleq B\,, \quad C^a = C^s \triangleq C\,, \quad D^a = D^s \triangleq D\,, \tag{140}$$

and also,

$$C^\ell \equiv 0 \tag{141}$$

then the simulation filter (WASHOUT FILTER) can be simplified and its order reduced as follows.

Substituting the symmetric case into (92), (93), (95)–(97) we observe that:

$$P^a = -P^s\,. \tag{142}$$

Substituting (140)–(142) into (89), (90) we obtain,

$$F^a = -F^s \tag{143}$$

and

$$\begin{aligned} F^s &= R_2^{-1}(B^T P^s + D^T QC)\,, && (144)\\ F^n &= R_2^{-1}(B^T P^n - D^T QDC^n)\,. && (145) \end{aligned}$$

Let us define a new error state vector:

$$\tilde{\mathrm{x}}^e(t) \triangleq \tilde{\mathrm{x}}^a(t) - \tilde{\mathrm{x}}^s(t)\,. \qquad (146)$$

Using this new state, $\tilde{\mathrm{x}}^e(t)$, and result (143) we can write the system $\tilde{\mathcal{W}}$, (73)–(74) in a new form:

$$\tilde{\mathcal{W}}: \quad \begin{aligned} \dot{\tilde{\mathrm{x}}}(t) &= \tilde{A}\tilde{\mathrm{x}}(t) + \tilde{B}(t)\tilde{\mathrm{u}}^a(t) && (147)\\ \tilde{\mathrm{u}}^s(t) &= \tilde{C}\tilde{\mathrm{x}}(t) + \tilde{D}(t)\tilde{\mathrm{u}}^a(t) && (148) \end{aligned}$$

where

$$\tilde{\mathrm{x}}(t) \triangleq \begin{pmatrix} \tilde{\mathrm{x}}^e(t) \\ \tilde{\mathrm{x}}^a(t) \end{pmatrix} \qquad (149)$$

and

$$\tilde{A} = \begin{pmatrix} A - BF^s & 0 \\ 0 & A \end{pmatrix}, \quad \tilde{B}(t) = \begin{pmatrix} B + BF^n(t)\mathcal{C}^n(t)^{-1} \\ B \end{pmatrix}, \qquad (150)$$

$$\tilde{C} = -(F^s,\, 0)\,, \quad \tilde{D}(t) = -F^n(t)\mathcal{C}^n(t)^{-1}\,. \qquad (151)$$

We see that this choice of state vector decoupled $\tilde{A}$ into two systems, where the output of the system $\tilde{\mathcal{W}}$, $\tilde{u}^s(t)$, is only dependent on $\tilde{x}^e(t)$. Therefore, in the *symmetric case*, the system $\tilde{\mathcal{W}}$ can be reduced into a smaller system $\mathcal{W}^S$, of $\dim(A)$, instead of $\dim(A^s)+\dim(A^a)$:

$$\tilde{\mathcal{W}}^S: \quad \begin{aligned} \dot{\tilde{\mathrm{x}}}^e(t) &= A^{\tilde{w}^S}\tilde{\mathrm{x}}^e(t) + B^{\tilde{w}^S}(t)\tilde{\mathrm{u}}^a(t) && (152)\\ \tilde{\mathrm{u}}^s(t) &= C^{\tilde{w}^S}\tilde{\mathrm{x}}^e(t) + D^{\tilde{w}^S}(t)\tilde{\mathrm{u}}^a(t) && (153) \end{aligned}$$

where

$$A^{\tilde{w}^S} = A - BF^s\,, \quad B^{\tilde{w}^S}(t) = B + BF^n(t)\mathcal{C}^n(t)^{-1}\,, \qquad (154)$$

$$C^{\tilde{w}^S} = -F^s\,, \quad D^{\tilde{w}^S}(t) = -F^n(t)\mathcal{C}^n(t)^{-1}\,. \qquad (155)$$

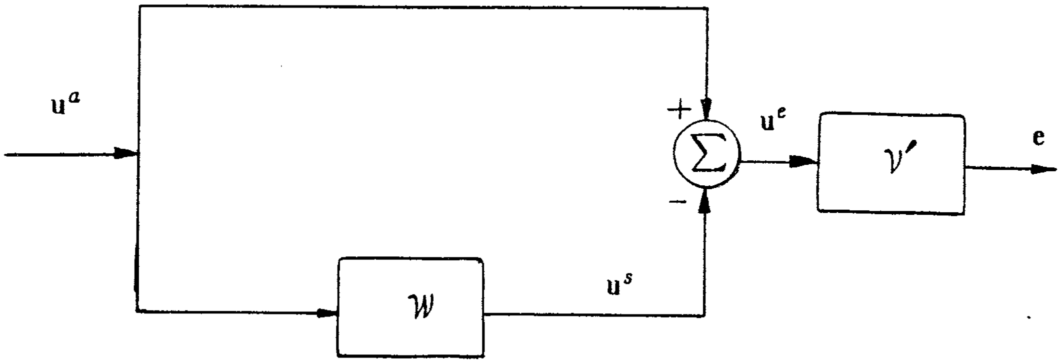


Figure 3. Symmetric case simplification (see Figure II.11).

The transfer matrix form of $\tilde{\mathcal{W}}^S$ for a time-invariant C^n can be obtained using the $\mathcal{L}$ transform (compare to (79)):

$$\tilde{W}(s) = -F^n C^{n-1} + F^s(sI - A + BF^s)^{-1}B(F^n C^{n-1} + I)\,. \qquad (156)$$

To summarize, we see that the order of the WASHOUT FILTER obtained (156) is of dim(A) rather than dim(A^s)+dim(A^a) in (79). This result generalizes such that if the system $\mathcal{S}^s$ is a cascade of some system (usually the motion-base dynamics) and of the system $\mathcal{S}^a$, then the order of the WASHOUT FILTER, $\tilde{\mathcal{W}}^S$, obtained is that of dim(A^s). Furthermore, we only need to solve for P^s and P^n while the result needed for P^a is obtained as a sub-matrix of P^s.

The result derived here can also be derived by combining $\mathcal{S}^a$ and $\mathcal{S}^s$ in such a way that their input, $\mathbf{u}^e$, is the difference between $\mathbf{u}^a$ and $\mathbf{u}^{s'}$ (Figure 3). In general, we can combine common cascaded parts of $\mathcal{S}^a$ and $\mathcal{S}^s$ (Figure II.11).

6. Time-Varying Stochastic Problem Solution

In the statement of the L.Q. case, we assumed that the systems $\mathcal{S}^s$ and $\mathcal{S}^a$, and cost matrices A and R, were time-invariant (1)–(7). Furthermore, we need a system $\mathcal{N}$ which was time-invariant to obtain the time-invariant solution to (67). We now solve for the case of a time-varying $\mathcal{N}$, which beyond its generality, allows us to use a lower dimensional system $\mathcal{N}$, to model the expected input $\mathbf{u}^a$. The same method can be used to extend the solution to a time-varying system $\mathcal{S}^a$. The cost matrices

Q and R can also be generalized to some classes of $Q(t)$ and $R(t)$ which can be handled by system augmentation, called *Frequency-Shaping of Cost Functions* [Gupta80].

Repeating the steps in Subsection 5.1 for (65) with the boundary condition (66), we obtain similar observations, with the exception of having $-\dot{\mathrm{p}}^{i,j}(t)$ on the right hand side of equations (82)–(87), (92)–(94) instead of zero. Now due to observations (i), (ii), (iv) and (vi), we have that (92) and (93), are independent of $\mathcal{N}$ and thus we obtain a steady state time-invariant solution for P^s, P^a, F^s and F^a as before. The only time varying solution we obtain (even in the steady state) is for $P^n(t)$:

$$-\dot{P}^n(t) = \mathcal{A}^T P^n(t) + P^n(t) A^n(t) + \mathcal{C}(t) \tag{157}$$

with the boundary conditions (from (66)):

$$P^n(\infty) = P^n_\infty \tag{158}$$

where

$$\mathcal{C}(t) = \mathcal{K} C^n(t) \tag{159}$$

and with $\mathcal{A}$ (115) time-invariant,

$$\mathcal{A} = \mathcal{A}^s - B^s R_2^{-1} B^{s^T} P^s \tag{115}$$

$$\mathcal{A}^s = A^s - B^s R_2^{-1}(D^{s^T} Q C^s + \rho D^{\ell^T} R C^{\ell}) . \tag{95}$$

The solution to (157) is:

$$P^n(t) = e^{-\mathcal{A}^T t} P^n(0) \Phi(t, o) + \int_0^t e^{-\mathcal{A}^T (t-\tau)} \mathcal{C}(\tau) \Phi(\tau, t)\, d\tau \tag{160}$$

where $\Phi(t_1, t_2)$ is the state transition matrix for the system

$$\dot{\mathrm{x}}(t) = A^n(t) \mathrm{x}(t) \tag{161}$$

from time t_1 to time t_2. The initial condition $P^n(0)$ is computed backwards to fulfill the boundary condition (158). In fact, it is better to obtain a solution similar to (160) by backward integration of (157). This is due to the following facts: (i) the matrix $\mathcal{A}$ is the closed-loop system matrix of system $\tilde{S}^s$, which is asymptotically stable according to our design procedure; (ii) A^n is usually an asymptotically stable matrix. Therefore the state transition matrices are best integrated in their stable direction, which is backward in time for (157).

In the special triangular case for A^n (127):

$$\mathbf{p}_i^n(t) = e^{-\int_0^t a_{i,i}^n(\gamma)\,d\gamma} e^{-\mathcal{A}^T t} \mathbf{p}_i(0) + e^{-\int_\tau^t a_{i,i}^n(\gamma)\,d\gamma} e^{-\mathcal{A}^T(t-\tau)} \mathbf{c}_i'(\tau)\,d\tau \tag{162}$$

where $\mathbf{c}_i'$ depends on previous $\mathbf{c}_j'$ (134) (for $j < i$)

$$\mathbf{c}_i'(t) = \mathbf{c}_i(t) + \sum_{j=1}^{i-1} a_{j,i}^n\, \mathbf{p}_j^n(t)\,. \tag{163}$$

Note that the computation of the state transition matrix $\Phi(t_1, t_2)$ is reduced to only computing $n^{\mathcal{N}}$ scalar exponent integrals of $a_{i,i}^n(t)$, which makes this computation feasible to implement in real time. Note also that $-a_{i,i}^n$ and $-\mathcal{A}^T$ are unstable, which may cause numerical computation problems for large t.

References

[Gupta80] Gupta, N.K., "Frequency-Shaped Cost Functionals: An Extension of L.Q.G. Design Methods," *Journal of Guidance and Control*, 1980.

[Kwakernaak72] Kwakernaak, H. and Sivan, R., *Linear Optimal Control Systems*, p. 231, 253, Wiley, 1972.

[MACSYMA77] M.I.T. Mathlab group, MACSYMA *Reference Manual*, version nine, Dec., 1977.

[Oppenheim75] Oppenheim, A.V., Schafer, R.W., *Digital Signal Processing*, p. 113, Prentice-Hall, 1975.

[Sinacori77] Sinacori, J.B., *The Determination of Some Requirements for a Helicopter Flight Research Simulation Facility*, Technical Report no. 1097-1, Systems Technology, Inc., Contract NAS2-9421 (CR-152066), pp. 2–12, Sep., 1977.

[Sivan82] Sivan, R., Ish-Shalom, J. and Huang, J.-K., "On the Design of Optimal Simulators with Application to Flight Simulators", *IEEE Transactions on Man Systems and Cybernetics*, SMC, Nov., 1982.

[Youla76] Youla, D.C., Jabr, H.A. and Bongiorno, J.J., "Modern Wiener-Hopf Design of Optimal Controllers—Part II: The Multivariable Case," *IEEE Transactions on Automatic Control*, **AC-21**, no. 3, pp. 319–338, June, 1976.

Appendix III.A: Derivation of Block Equations

--------------- (Macsyma output edited a bit)

```
(D4)                   Sunday, Jul 11, 1982  6:09am
(C5) ALIAS(T,TRANSPOSE)$
(C6) BOTHCASES:TRUE$
(C7) MATRIX_ELEMENT_MULT:"."$
(C8) MATRIX_ELEMENT_TRANSPOSE:NONSCALARS$
(C9) DOTSCRULES:TRUE$
(C10) DECLARE([q,rho],SCALAR)$
(C11) DECLARE([Aa,Ba,Ca,Da,As,Bs,Cs,Ds,An,Bn,Cn,A1,B1,C1,D1],NONSCALAR)$
(C12) DECLARE([Paa,Pa,Pan,Ps,Pn,Pnn],NONSCALAR)$
(C13) /* DECLARE([Q,R,R2])$ */
/* not used so that T(Q)=Q, T(R)=R ... (all are symatric matrices) */
DECLARE(T,ADDITIVE)$
(C14) PR(MVAR):=SUBST('A[s],As,SUBST('B[s],Bs,SUBST('C[s],Cs,SUBST('D[s],Ds,
SUBST('R[d],Rd,SUBST('A[a],Aa,SUBST('B[a],Ba,SUBST('C[a],Ca,SUBST('D[a],Da,
SUBST('A[1],A1,SUBST('B[1],B1,SUBST('C[1],C1,SUBST('D[1],D1,SUBST('R[2],R2,
SUBST('A[n],An,SUBST('C[n],Cn,SUBST('P[a,a],Paa,SUBST('P[a],Pa,
SUBST('P[a,n],Pan,SUBST('P[s],Ps,SUBST('P[n],Pn,SUBST('P[n,n],Pnn,
SUBST(Aa^T,T(Aa),SUBST(Ba^T,T(Ba),SUBST(Ca^T,T(Ca),SUBST(Da^T,T(Da),
SUBST(As^T,T(As),SUBST(Bs^T,T(Bs),SUBST(Cs^T,T(Cs),SUBST(Ds^T,T(Ds),
SUBST(An^T,T(An),SUBST(Cn^T,T(Cn),SUBST(C1^T,T(C1),SUBST(D1^T,T(D1),
SUBST(Pa^T,T(Pa),SUBST(Pan^T,T(Pan),SUBST(Pn^T,T(Pn),
MVAR)))))))))))))))))))))))))))))))))))))$
(C15) TRANSLATE(PR):    (D15)   [PR]
(C16) A%:MATRIX([As,0,0],[0,Aa,Ba.Cn],[0,0,An])$
(C17) B%:MATRIX([Bs],[0],[0])$
(C18) C%:MATRIX([-Cs,Ca,Da.Cn],[C1,0,0])$
(C19) D%:MATRIX([-Ds],[D1])$
(C20) P:MATRIX([Ps,Pa,Pn],[T(Pa),Paa,Pan],[T(Pn),T(Pan),Pnn])$
(C21) Q%:MATRIX([Q,0],[0,rho*R])$
(C22) R1:T('C%) . 'Q% . 'C%$
(C23) R12:T('C%) . 'Q% . 'D%$
(C24) RR2:T('D%).'Q%.'D%$
(C25) RICCATI:-'P.'RR%.'P+'P.'AR%+T('AR%).'P+'QR%$
(C26) RR%:'B%.[R2^^(-1)].T('B%)$
(C27) AR%:'A%-'B%.[R2^^(-1)].T('R12)$
(C28) QR%:'R1-'R12.[R2^^(-1)].T('R12)$
(C29) BLOCK(
PRINT(" A% =",PR(A%),"          ","B% =",PR(B%),"
","C% =",PR(C%),"          ","D% =",PR(D%),"
"," P =",PR(P),"
","Q% =",Q%,"
","The Algebraic Riccati Equation: ",SUBST('AR%^T,T('AR%),RICCATI),"
","RR% =",SUBST('B%^T,T('B%),SUBST(R[2]^^(-1),[R2^^(-1)],RR%)),"
","AR% =",SUBST('R[1,2]^T,T('R12),SUBST(R[2]^^(-1),[R2^^(-1)],AR%)),"
"," QR% =",SUBST('R[1,2],'R12,SUBST(R[2]^^(-1),[R2^^(-1)],
                 SUBST('R[1],R1,SUBST('R[1,2]^T,T('R12),QR%)))),"
",'R[1],"=",SUBST('C%^T,T('C%),R1),"
",'R[1,2],"=",SUBST('C%^T,T('C%),R12),"
",'R[2],"=",SUBST('D%^T,T('D%),RR2)),
R1:EV(R1,NOUNS),
R12:EV(R12,NOUNS),
RR2:EV(RR2,NOUNS))$
```

$$A\% = \begin{bmatrix} A_s & 0 & 0 \\ 0 & A_a & B_a . C_n \\ 0 & 0 & A_n \end{bmatrix} \qquad B\% = \begin{bmatrix} B_s \\ 0 \\ 0 \end{bmatrix}$$

$$C\% = \begin{bmatrix} - C_s & C_a & D_a . C_n \\ C_1 & 0 & 0 \end{bmatrix} \qquad D\% = \begin{bmatrix} - D_s \\ D_1 \end{bmatrix}$$

$$P = \begin{bmatrix} P_s & P_a & P_n \\ P_a^T & P_{a, a} & P_{a, n} \\ P_n^T & P_{a, n}^T & P_{n, n} \end{bmatrix}$$

$$Q\% = \begin{bmatrix} Q & 0 \\ 0 & \mathrm{rho}\ R \end{bmatrix}$$

The Algebraic Riccati Equation: $QR\% - P . RR\% . P + P . AR\% + AR\%^T . P$

$$RR\% = B\% . R_2^{<- 1>} . B\%^T$$

$$AR\% = A\% - B\% . R_2^{<- 1>} . R_{1, 2}^T$$

$$QR\% = R1 - R_{1, 2} . R_2^{<- 1>} . R_{1, 2}^T$$

$$R_1 = C\%^T . Q\% . C\%$$

$$R_{1, 2} = C\%^T . Q\% . D\%$$

$$R_2 = D\%^T . Q\% . D\%$$

```
(C30) BLOCK(
PRINT('R[1],"=",PR(R1),"
",'R[1,2],"=",PR(R12),"
",'R[2],"=",PR(RR2)))$
              [  T                  T           ]
              [ C  . Q . C  + rho (C  . R . C ) ]
              [  s        s         1        1  ]
              [                                 ]
              [            T                    ]
R  =  Col 1 = [          - C  . Q . C           ]
 1            [             a        s          ]
              [                                 ]
              [          T    T                 ]
              [       - C  . D  . Q . C         ]
              [          n    a        s        ]

         [     T            ]         [     T                 ]
         [  - C  . Q . C    ]         [  - C  . Q . D  . C    ]
         [     s        a   ]         [     s        a    n   ]
         [                  ]         [                       ]
         [    T             ]         [    T                  ]
 Col 2 = [   C  . Q . C     ] Col 3 = [   C  . Q . D  . C     ]
         [    a        a    ]         [    a        a    n    ]
         [                  ]         [                       ]
         [  T    T          ]         [  T    T               ]
         [ C  . D  . Q . C  ]         [ C  . D  . Q . D  . C  ]
         [  n    a        a ]         [  n    a        a    n ]

         [  T                  T           ]
         [ C  . Q . D  + rho (C  . R . D ) ]
         [  s        s         1        1  ]
         [                                 ]
         [             T                   ]
 R     = [          - C  . Q . D           ]
  1, 2   [             a        s          ]
         [                                 ]
         [          T    T                 ]
         [       - C  . D  . Q . D         ]
         [          n    a        s        ]

       T                  T
 R  = D  . Q . D  + rho (D  . R . D )
  2    s        s         1        1
```

```
(C31) RR%:FACTOR(EV(RR%,NOUNS,EXPAND))$
(C32) AR%:A%+FACTOR(SUBST(R,T(R),SUBST(Q,T(Q),SUBST(R2^^(-1),T(R2^^(-1)),EV(AR%-'
A%,NOUNS,EXPAND)))))$
(C33) QR%:FACTOR(SUBST(R,T(R),SUBST(Q,T(Q),SUBST(R2^^(-1),T(R2^^(-1)),
EV(QR%-'R1,NOUNS,EXPAND)))))+FACTOR(R1)$
(C34) EQ:EV(RICCATI,NOUNS)$
(C35) EQ:SUBST(Q,T(Q),EQ)$
(C36) EQ:SUBST(R,T(R),EQ)$
(C37) EQ:SUBST(R2^^(-1),T(R2^^(-1)),EQ)$
(C38) EQ11:ISOLATE(EQ[1,1],Ps)$
(C41) EQ12:ISOLATE(EQ[1,2],Pa)$
(C42) EQ13:ISOLATE(EQ[1,3],Pn)$
(C43) (FOR ii:1 THRU 3 DO
     FOR ij:ii THRU 3 DO
        PRINT("

","Block ",ii,",",ij,"
",PR(EQ[ii,ij]))),
PRINT("

","Block 1 , 1
",PR(EQ[1,1]),"

","Block 1 , 2
",PR(EQ[1,2]),"

","Block 1 , 3
",PR(EQ[1,3]),"

","Block 1 , 1
",PR(EQ11),"

","Block 1 , 2
",PR(EQ12),"

","Block 1 , 3
",PR(EQ13),"

","Subexpreations

"),
(FOR i:LENGTH(LABELS(E)) STEP -1 THRU 1 DO PRINT
(LABELS(E)[i],"=",PR(EV(LABELS(E)[i])),"
"))$
```

Block 1 , 1

```
    T               <- 1>     T     T          T               <- 1>     T
(- C   . Q . D   . R       . B   + A   - rho (C   . R . D   . R       . B  ))
    s         s     2         s     s          1         1     2         s

                        <- 1>     T
. P   + P   . (- B   . R       . D   . Q . C
   s     s        s     2         s         s

              <- 1>     T                                   <- 1>     T
- rho (B   . R       . D   . R . C  ) + A  ) - P   . B   . R       . B   . P
        s     2         1         1      s      s     s     2         s     s

   T               <- 1>     T
- C   . Q . D   . R       . D   . Q . C
   s         s     2         s         s

        T               <- 1>     T                T
- rho (C   . Q . D   . R       . D   . R . C  ) + C   . Q . C
        s         s     2         1         1      s         s

        T               <- 1>     T
- rho (C   . R . D   . R       . D   . Q . C  )
        1         1     2         s         s

     2   T               <- 1>     T                     T
- rho  (C   . R . D   . R       . D   . R . C  ) + rho (C   . R . C  )
         1         1     2         1         1           1         1
```

Block 1 , 2

```
    T               <- 1>     T     T          T               <- 1>     T
(- C   . Q . D   . R       . B   + A   - rho (C   . R . D   . R       . B  ))
    s         s     2         s     s          1         1     2         s

                     <- 1>     T                           <- 1>     T
. P   + P   . B   . R       . D   . Q . C   - P   . B   . R       . B   . P
   a     s     s     2         s         a     s     s     2         s     a

   T               <- 1>     T               T
+ C   . Q . D   . R       . D   . Q . C   - C   . Q . C
   s         s     2         s         a     s         a

        T               <- 1>     T
+ rho (C   . R . D   . R       . D   . Q . C  ) + P   . A
        1         1     2         s         a      a     a
```

```
Block 1 , 3

       T                     <- 1>          T       T               T                     <- 1>          T
(- C   .  Q   .  D    .  R          .   B   +   A   -  rho    ( C    .  R  .  D    .  R          .   B  ))
       s             s       2              s       s               1             1       2              s

                             <- 1>          T
.   P    +  P    .  B    .  R          .  D    .   Q  .  D    .  C
     n       s       s       2              s             a       n

                     <- 1>          T               T                     <- 1>          T
-   P    .  B    .  R          .  B    .  P    +  C    .  Q  .   D    .  R          .  D    .  Q  .  D
     s       s       2              s       n       s             s       2              s             a

             T
.   C   -   C   .   Q  .  D    .  C    +  P    .  A    +   rho
     n       s              a       n       n       n

    T                    <- 1>          T
( C    .  R  .  D    .  R          .   D    .  Q  .  D    .  C   )  +  P    .  B    .  C
   1              1       2             s             a        n        a       a       n
```

Block 1 , 1

$$- P_{s} . B_{s} . R_{2}^{<-1>} . B_{s}^{T} . P_{s} + P_{s} . E39 + E40 + E38 . P_{s}$$

Block 1 , 2

$$- P_{s} . B_{s} . R_{2}^{<-1>} . B_{s}^{T} . P_{a} + P_{a} . A_{a} + E41 + E38 . P_{a}$$

Block 1 , 3

$$- P_{s} . B_{s} . R_{2}^{<-1>} . B_{s}^{T} . P_{n} + P_{n} . A_{n} + E42 + E38 . P_{n}$$

Subexpreations

$$E38 = - C_s^T . Q . D_s . R_2^{<-1>} . B_s^T + A_s^T - \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T)$$

$$E39 = - B_s . R_2^{<-1>} . D_s^T . Q . C_s - \mathrm{rho}\ (B_s . R_2^{<-1>} . D_1^T . R . C_1) + A_s$$

$$E40 = - C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . C_s$$

$$- \mathrm{rho}\ (C_s^T . Q . D_s . R_2^{<-1>} . D_1^T . R . C_1) + C_s^T . Q . C_s$$

$$- \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . C_s)$$

$$- \mathrm{rho}^2\ (C_1^T . R . D_1 . R_2^{<-1>} . D_1^T . R . C_1) + \mathrm{rho}\ (C_1^T . R . C_1)$$

$$E41 = P_s . B_s . R_2^{<-1>} . D_s^T . Q . C_a$$

$$+ C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . C_a - C_s^T . Q . C_a$$

$$+ \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . C_a)$$

$$E42 = P_s . B_s . R_2^{<-1>} . D_s^T . Q . D_a . C_n$$

$$+ C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . D_a . C_n - C_s^T . Q . D_a . C_n$$

$$+ \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . D_a . C_n) + P_a . B_a . C_n$$

Block 1 , 1

$$(- C_s^T . Q . D_s . R_2^{<-1>} . B_s^T + A_s^T - \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T))$$

$$. P_s + P_s . (- B_s . R_2^{<-1>} . D_s^T . Q . C_s$$

$$- \mathrm{rho}\ (B_s . R_2^{<-1>} . D_1^T . R . C_1) + A_s) - P_s . B_s . R_2^{<-1>} . B_s^T . P_s$$

$$- C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . C_s$$

$$- \mathrm{rho}\ (C_s^T . Q . D_s . R_2^{<-1>} . D_1^T . R . C_1) + C_s^T . Q . C_s$$

$$- \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . C_s)$$

$$- \mathrm{rho}^2\ (C_1^T . R . D_1 . R_2^{<-1>} . D_1^T . R . C_1) + \mathrm{rho}\ (C_1^T . R . C_1)$$

Block 1 , 2

$$(- C_s^T . Q . D_s . R_2^{<-1>} . B_s^T + A_s^T - \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T))$$

$$. P_a + P_s . B_s . R_2^{<-1>} . D_s^T . Q . C_a - P_s . B_s . R_2^{<-1>} . B_s^T . P_a$$

$$+ C_s^T . Q . D_s . R_2^{<-1>} . D_s^T . Q . C_a - C_s^T . Q . C_a$$

$$+ \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . Q . C_a) + P_a . A_a$$

Block 1 , 3

$$(- C_s^T . Q . D_s . R_2^{<- 1>} . B_s^T + A_s^T - rho (C_1^T . R . D_1 . R_2^{<- 1>} . B_s^T))$$

$$. P_n + P_s . B_s . R_2^{<- 1>} . D_s^T . Q . D_a . C_n$$

$$- P_s . B_s . R_2^{<- 1>} . B_s^T . P_n + C_s^T . Q . D_s . R_2^{<- 1>} . D_s^T . Q . D_a$$

$$. C_n - C_s^T . Q . D_a . C_n + P_n . A_n + rho$$

$$(C_1^T . R . D_1 . R_2^{<- 1>} . D_s^T . Q . D_a . C_n) + P_a . B_a . C_n$$

Block 2 , 2

$$P_{a, a} . A_a + P_a^T . B_s . R_2^{<- 1>} . D_s^T . Q . C_a$$

$$- P_a^T . B_s . R_2^{<- 1>} . B_s^T . P_a - C_a^T . Q . D_s . R_2^{<- 1>} . D_s^T . Q . C_a$$

$$+ C_a^T . Q . D_s . R_2^{<- 1>} . B_s^T . P_a + C_a^T . Q . C_a + A_a^T . P_{a, a}$$

Block 2 , 3

$$P_{a, n} . A_n + P_{a, a} . B_a . C_n + P_a^T . B_s . R_2^{<- 1>} . D_s^T . Q . D_a . C_n$$

$$- P_a^T . B_s . R_2^{<- 1>} . B_s^T . P_n - C_a^T . Q . D_s . R_2^{<- 1>} . D_s^T . Q . D_a$$

$$. C_n + C_a^T . Q . D_s . R_2^{<- 1>} . B_s^T . P_n + C_a^T . Q . D_a . C_n + A_a^T . P_{a, n}$$

```
Block  3 , 3

              T         <- 1>    T
P     . A  + P  . B  . R      . D  . Q . D  . C
 n, n    n    n    s    2        s        a    n

   T         <- 1>    T         T    T             <- 1>    T
- P  . B  . R      . B  . P  - C  . D  . Q . D  . R      . D  . Q
   n    s    2        s    n    n    a        s    2        s

             T    T             <- 1>    T
. D  . C  + C  . D  . Q . D  . R      . B  . P
   a    n    n    a        s    2        s    n

   T    T                  T    T            T
+ C  . D  . Q . D  . C  + C  . B  . P     + A  . P
   n    a        a    n    n    a    a, n    n    n, n

   T
+ P     . B  . C
   a, n    a    n
```

```
(C44) STATUS(RUNTIME);
(D44)                        102567 MSEC
```

```
/*    Appendix III.P:  Macsyma Program   */
/*    --------------                     */
LINEL:70$
WRITEFILE(DSK,JEHUDA)$
TIMEDATE();
ALIAS(T,TRANSPOSE)$
BOTHCASES:TRUE$
MATRIX_ELEMENT_MULT:"."$
MATRIX_ELEMENT_TRANSPOSE:NONSCALARS$
DOTSCRULES:TRUE$
DECLARE([q,rho],SCALAR)$
DECLARE([Aa,Ba,Ca,Da,As,Bs,Cs,Ds,An,Bn,Cn,Al,Bl,Cl,Dl],NONSCALAR)$
DECLARE([Paa,Pa,Pan,Ps,Pn,Pnn],NONSCALAR)$
/* DECLARE([Q,R,R2])$ */
/* not used so that T(Q)=Q, T(R)=R ... (all are symatric matrices) */
DECLARE(T,ADDITIVE)$
PR(MVAR):=SUBST('A[s],As,SUBST('B[s],Bs,SUBST('C[s],Cs.SUBST('D[s],Ds,
SUBST('R[d],Rd,SUBST('A[a],Aa,SUBST('B[a],Ba,SUBST('C[a],Ca,SUBST('D[a],Da,
SUBST('A[l],Al,SUBST('B[l],Bl,SUBST('C[l],Cl,SUBST('D[l],Dl,SUBST('R[2],R2,
SUBST('A[n],An,SUBST('C[n],Cn,SUBST('P[a,a],Paa,SUBST('P[a],Pa,
SUBST('P[a,n],Pan,SUBST('P[s],Ps,SUBST('P[n],Pn,SUBST('P[n,n],Pnn,
SUBST(Aa^T,T(Aa),SUBST(Ba^T,T(Ba),SUBST(Ca^T,T(Ca),SUBST(Da^T,T(Da),
SUBST(As^T,T(As),SUBST(Bs^T,T(Bs),SUBST(Cs^T,T(Cs),SUBST(Ds^T,T(Ds),
SUBST(An^T,T(An),SUBST(Cn^T,T(Cn),SUBST(Cl^T,T(Cl),SUBST(Dl^T,T(Dl),
SUBST(Pa^T,T(Pa),SUBST(Pan^T,T(Pan),SUBST(Pn^T,T(Pn),
MVAR))))))))))))))))))))))))))))))))))))$
TRANSLATE(PR);
A%:MATRIX([As,0,0],[0,Aa,Ba.Cn],[0,0,An])$
B%:MATRIX([Bs],[0],[0])$
C%:MATRIX([-Cs,Ca,Da.Cn],[Cl,0,0])$
D%:MATRIX([-Ds],[Dl])$
P:MATRIX([Ps,Pa,Pn],[T(Pa),Paa,Pan],[T(Pn),T(Pan),Pnn])$
Q%:MATRIX([Q,0],[0,rho*R])$
R1:T('C%) . 'Q% . 'C%$
R12:T('C%) . 'Q% . 'D%$
RR2:T('D%).'Q%.'D%$
RICCATI:-'P.'RR%.'P+'P.'AR%+T('AR%)..'P+'QR%$
RR%:'B%.[R2^^(-1)].T('B%)$
AR%:'A%-'B%.[R2^^(-1)].T('R12)$
QR%:'R1-'R12.[R2^^(-1)].T('R12)$
BLOCK(
PRINT(" A% =",PR(A%),"          ","B% =",PR(B%),"
","C% =",PR(C%),"        ","D% =",PR(D%),"
"," P =",PR(P),"
","Q% =",Q%,"
","The Algebraic Riccati Equation: ",SUBST('AR%^T,T('AR%),RICCATI),"
","RR% =",SUBST('B%^T,T('B%),SUBST(R[2]^^(-1),[R2^^(-1)],RR%)),"
","AR% =",SUBST('R[1,2]^T,T('R12),SUBST(R[2]^^(-1),[R2^^(-1)],AR%)),"
"," QR% =",SUBST('R[1,2],'R12,SUBST(R[2]^^(-1),[R2^^(-1)],
                 SUBST('R[1],R1,SUBST('R[1,2]^T,T('R12),QR%)))),"
",'R[1],"=",SUBST('C%^T,T('C%),R1),"
",'R[1,2],"=",SUBST('C%^T,T('C%),R12),"
",'R[2],"=",SUBST('D%^T,T('D%),RR2)),
R1:EV(R1,NOUNS),
R12:EV(R12,NOUNS),
RR2:EV(RR2,NOUNS))$
```

```
BLOCK(
PRINT('R[1],"=",PR(R1),"
",'R[1,2],"=",PR(R12),"
",'R[2],"=",PR(RR2)))$
RR%:FACTOR(EV(RR%,NOUNS,EXPAND))$
AR%:A%+FACTOR(SUBST(R,T(R),SUBST(Q,T(Q),SUBST(R2^^(-1),T(R2^^(-1)),EV(AR%-'A%,NOU
NS,EXPAND)))))$
QR%:FACTOR(SUBST(R,T(R),SUBST(Q,T(Q),SUBST(R2^^(-1),T(R2^^(-1)),
EV(QR%-'R1,NOUNS,EXPAND)))))+FACTOR(R1)$
EQ:EV(RICCATI,NOUNS)$
EQ:SUBST(Q,T(Q),EQ)$
EQ:SUBST(R,T(R),EQ)$
EQ:SUBST(R2^^(-1),T(R2^^(-1)),EQ)$
EQ11:ISOLATE(EQ[1,1],Ps)$
EQ12:ISOLATE(EQ[1,2],Pa)$
EQ13:ISOLATE(EQ[1,3],Pn)$
(FOR ii:1 THRU 3 DO
     FOR ij:ii THRU 3 DO
        PRINT("

","Block ",ii,",",ij,"
",PR(EQ[ii,ij]))),
PRINT("

","Block 1 , 1
",PR(EQ[1,1]),"

","Block 1 , 2
",PR(EQ[1,2]),"

","Block 1 , 3
",PR(EQ[1,3]),"

","Block 1 , 1
",PR(EQ11),"

","Block 1 , 2
",PR(EQ12),"

","Block 1 , 3
",PR(EQ13),"

","Subexpreations

"),
(FOR i:LENGTH(LABELS(E)) STEP -1 THRU 1 DO PRINT
(LABELS(E)[i],"=",PR(EV(LABELS(E)[i])),"
"))$
STATUS(RUNTIME);
CLOSEFILE(OUTPUT,>);
```

Chapter IV

Nonlinearities

In this chapter we discuss three topics that involve nonlinear effects:

1. A deterministic-stochastic decomposition for a nonlinear plant with a quadratic cost.
2. A Pseudo Linear Quadratic (PLQ) control design method that gives a better design for hard boundaries constraints and which provides a more flexible control design for linear and nonlinear plants.
3. An extended performance criteria for the simulator motion design problem is developed which leads to a *correlation cost.* Some special properties of this *sign sensitive cost* are derived.

We now give an introduction and summary of each of these topics.

The most common question asked in the design of flight simulator motion is: what is the minimum motion-base size needed to meet "specified motion quality" requirements? Given that we use the vestibular error as the motion quality measure, a lower bound on the required motion-base can be found by solving for the case where all the problem uncertainties are removed. Thus we prepare the simulator cab at the best initial position for the expected maneuver, e.g. before a take-off we would move the simulator cab to one end of the linear surge axis, so as to have a maximum linear surge motion for the take-off itself, which only consists of forward acceleration. In contrast, if we had complete uncertainty in the cab motion we would best position the cab initially at the center of the linear surge travel and thus lose half of the simulator size. From this example, we see that complete knowledge of the future enables us to use a simulator that has only half the size of the original simulator for the same motion quality. It therefore seems that if we can find a way to combine a solution to the expected future input with a solution that handles the uncertainties, then we can reduce the required simulator size. The previous example also shows that the solution to a deterministic input is fundamental to the overall solution (at least gives a lower bound). Therefore, we derive a general deterministic-stochastic decomposition which both defines the deterministic and the stochastic subproblems, and also suggest how to combine their solutions. This decomposition has some similarities to the *extended Kalman filter* and thus favorable simulation results done by Schwartz and Stear [Jazwinski70] can be used to **support** the **iteration** solution procedure suggested and the **superiority over** a **linear** system (we do not attempt to prove convergence and error properties). Beyond that we develop a *statistical linearization* method which is claimed to be even better then the *extended Kalman filter* method mentioned before [Gelb74]. Our main result is to suggest a decomposition procedure for a quadratic cost with nonlinear plant equations.

The second topic discussed in this chapter is Pseudo Linear Quadratic (PLQ) control. A Linear system with a Quadratic optimization criterion (L.Q. problem) leads to a linear feedback, similarly the minimization of integrals containing quartic or hexadic terms in the state variables leads, respectively, to cubic or quintic feedback. This idea was extended by Buss to the minimization of integrals including a finite or infinite sum of positive definite homogeneous multinomial forms of positive even degree of the state variables, which is desirable in order to impose inequality constraints upon the state variables [Bass66]. Such feedback laws are adaptive to actuator saturation and travel limitations in flight simulators and evolve from a desired *minimax criterion* of optimality:

$$\min_{\mathbf{u}} \max_{t} \phi(\mathbf{x}(t)), \tag{1}$$

where $\phi(\mathbf{x})$ denotes a positive definite scalar function, $\mathbf{x}$ the state vector, t time and $\mathbf{u}$ the control to be chosen. In practice, this criterion may be approximated by the criterion:

$$\min_{\mathbf{u}} \int_{0}^{T} (\phi(\mathbf{x}(t)))^{2\nu} dt, \quad (\nu = 1, 2, 3, \cdots) \tag{2}$$

for a large integral ν [Bass66].

This latter criterion was applied to the design of flight simulator motion by Kosut [Kosut79]. Kosut's problem formulation and solution are very nice, but the example he uses is oversimplified due to the cumbersome computation of the optimal control. Furthermore using a large ν (Kosut uses $\nu = 2$) results in a high order polynomial (a tensor actually) for the feedback control, the evaluation of which may not be feasible in real time. Therefore we derive a *sub-optimal* nonlinear feedback controller that requires less cumbersome computation both off-line and in real time. Furthermore, this approach allows us to use a non-symmetric control as is required, for example, for the *Link GAT-1* flight simulator as shown in chapter VII. This controller is a *Pseudo Linear Quadratic* controller (PLQ). It is derived from the standard *Linear Quadratic* optimal control solution by solving for a quasi-quadratic cost for each value of the state $\mathbf{x}$, i.e. using an algebraic Riccati equation that depends on $\mathbf{x}$. As one might expect, the PLQ solution has properties that are very similar to those obtained by Kosut for his example [Kosut79]. Furthermore, PLQ control also extends to the control of many nonlinear plants such as the ones obtained in robotic applications. In general, PLQ should be viewed as a compiler that translates design specification given as a *pseudo-quadratic* cost function in to a stable, easily implemented, feedback control law,

$$\mathbf{u}(t) = -F(\mathbf{x})\,\mathbf{x}(t)\,. \tag{3}$$

The feedback gain function $F(\mathbf{x})$ can be implemented using a small table lookup and interpolation. For example, in the *GAT-1* control system implementation the table lookup for $F(\mathbf{x})$ uses only four points. A prime candidate for PLQ is the design of a hexapod motion-base, where the plant is approximately linear but the cost function is nonquadratic due to the transformation from inertial axes to leg extension (see chapter II).

The third topic addressed in this chapter is the use of a *correlation cost*, or *sign sensitive cost*. In order to model some further characteristics of the vestibular organ we extend our previous vestibular model to include a memoryless function $f(y)$ cascaded to the output of our previous linear dynamic vestibular model. This function *normalizes* the vestibular output y into *"threshold units"*. Generally, $f(y)$ is an odd, compression type function, $\frac{df}{dy}(y_1) \geq \frac{df}{dy}(y_2)$ for $|y_2| \geq |y_1| > 0$. Such a function is motivated by Webber or Steven's perceptual laws (for the relative change in J.N.D. with increasing stimulation amplitude) and is also plausible if we consider the hair cell's input-output function (output firing rate versus input mechanical stimulation). It is proven that a general odd, compressive type function f(y) used to extend both the airplane pilot and the simulator pilot's linear vestibular models outputs (y^a and y^s, respectively) yields a sensitivity to the relative sign of the outputs y^a and y^s that should be included in the cost J_e in addition to the a quadratic form of the previous vestibular error $e = y^a - y^s$. Interestingly enough, this criterion was incorporated by Hosman in his design of motion washouts, with no explanation of its origin [Hosman79]. Another approach is to replace/augment our error $e \triangleq y^a - y^s$, comparison operation by the *correlation* operation $\mathfrak{E}\{y^a y^s\}$. Using *correlation* the relative sign of the y^a and y^s is also essential. Furthermore, the *correlation* operation suggested here is the optimal method to detect the "known" *reference signal*, y^a, in additive white Gaussian noise. The source of the *reference signal* used by the simulator pilot's brain is from the other sensors and the simulator pilot's expectation of what is about to happen.

A solution to the washout design problem is developed for the case when the cost J_e is augmented by a correlation term. The cost J_e is still quadratic. However, it is not necessarily positive anymore, since the state weighting matrix, Q, is no longer positive semidefinite as required in the standard L.Q. formulation. We show that a solution exists and is unique for the special case considered here. Furthermore, the solutions obtained seem to have the interesting characteristic of enhancing "motion" transitions, which is intuitively plausible. The difficulty with this solution is its "high" sensitivity to the relative weight parameter ρ of the control compared to the motion quality criteria.

1. Deterministic-Stochastic Problem Decomposition

Our objective is to find an optimal simulator, $\mathcal{W}$, defined in chapter III:

$$\mathbf{u}^s(t) = \mathcal{W}(\mathbf{u}^a(\tau); \quad \text{for } -\infty < \tau < \infty) . \qquad (\text{III}.8)$$

that minimizes a cost function J of arguments $\mathbf{y}^s$, $\mathbf{u}^\ell$ and $\mathbf{y}^a$. These arguments are solutions to the differential equations (4)–(8) of systems $\mathcal{S}^s$ and $\mathcal{S}^a$ with input $\mathbf{u}^a$. In this section we describe a *deterministic-stochastic decomposition* for nonlinear systems $\mathcal{S}^s$ and $\mathcal{S}^a$ where the cost function J is *quadratic*, defined by (chapter III):

$$\mathbf{e}(t) = \mathbf{y}^a(t) - \mathbf{y}^s(t)\,, \tag{III.6}$$

$$J = \mathfrak{E}\{\lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \mathbf{e}^T(t)Q\mathbf{e}(t) + \rho\mathbf{u}^{\ell^T}(t)R\mathbf{u}^\ell(t)\,dt\}\,. \tag{III.7}$$

Let us assume that $\mathcal{S}^s$ and $\mathcal{S}^a$ are *nonlinear, finite dimensional* systems which can be written in the following *state space form*:

$$\mathcal{S}^s: \quad \begin{aligned} \dot{\mathbf{x}}^s(t) &= \mathbf{f}^s(\mathbf{x}^s(t), \mathbf{u}^s(t), t) && (4)\\ \mathbf{y}^s(t) &= \mathbf{g}^s(\mathbf{x}^s(t), \mathbf{u}^s(t), t) && (5)\\ \mathbf{u}^\ell(t) &= \mathbf{g}^\ell(\mathbf{x}^s(t), \mathbf{u}^s(t), t) && (6)\end{aligned}$$

$$\mathcal{S}^a: \quad \begin{aligned} \dot{\mathbf{x}}^a(t) &= \mathbf{f}^a(\mathbf{x}^a(t), \mathbf{u}^a(t), t) && (7)\\ \mathbf{y}^a(t) &= \mathbf{g}^a(\mathbf{x}^a(t), \mathbf{u}^a(t), t) && (8)\end{aligned}$$

where $\mathbf{f}^s$, $\mathbf{g}^s$, $\mathbf{g}^l$, $\mathbf{f}^a$, $\mathbf{g}^a$ are vector functions of their arguments. The arguments $\mathbf{x}^s$, $\mathbf{x}^a$ are the system states and $\mathbf{u}^s$, $\mathbf{u}^a$ are the system inputs. For simplicity we assume that both systems have zero initial conditions. In our application of this decomposition we assume that the random input $\mathbf{u}^a(t) - \bar{\mathbf{u}}^a(t)$ defined in (10) is a Gaussian colored noise process.

The deterministic part of a variable is defined as its expected value, $\mathfrak{E}\{\ \}$, taken over the ensemble of all inputs $\mathbf{u}^a(t) \in \mathcal{U}^a$; it is denoted by an over-bar, e.g.:

$$\bar{\mathbf{x}}^a(t) = \mathfrak{E}\{\mathbf{x}^a(t)\}\,. \tag{9}$$

The stochastic part, denoted by the tilde variables is defined by:

$$\tilde{\mathbf{x}}^a(t) \triangleq \mathbf{x}^a(t) - \bar{\mathbf{x}}^a(t)\,, \tag{10}$$

for the actual system states and is similarly defined for $\tilde{\mathbf{x}}^s(t)$, $\tilde{\mathbf{u}}^a(t)$, $\tilde{\mathbf{u}}^s(t)$, $\tilde{\mathbf{y}}^a(t)$, $\tilde{\mathbf{y}}^s(t)$, $\tilde{\mathbf{u}}^\ell(t)$.

Let us write the function f( ) as:

$$\mathrm{f}(\mathrm{x}, \mathrm{u}, t) = \mathrm{f}(\bar{\mathrm{x}} + \tilde{\mathrm{x}}, \bar{\mathrm{u}} + \tilde{\mathrm{u}}, t). \tag{11}$$

Now, (11) can be expanded in a Taylor series around the expected values $\bar{\mathrm{x}}(t)$ and $\bar{\mathrm{u}}(t)$:

$$\mathrm{f}(\mathrm{x}, \mathrm{u}, t) = \mathrm{f}(\bar{\mathrm{x}}, \bar{\mathrm{u}}, t) + \left.\frac{\partial \mathrm{f}}{\partial \mathrm{x}}\right|_{\bar{\mathrm{x}},\bar{\mathrm{u}}} \tilde{\mathrm{x}} + \left.\frac{\partial \mathrm{f}}{\partial \mathrm{u}}\right|_{\bar{\mathrm{x}},\bar{\mathrm{u}}} \tilde{\mathrm{u}} + \cdots \tag{12}$$

where we **assume** that this series converges and that[1]:

$$|\mathrm{f}(\mathbf{x}, \mathbf{u}, t)| < \infty, \qquad \left|\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right| < \infty, \qquad \left|\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right| < \infty \qquad \text{for all} \qquad \mathbf{x},\, \mathbf{u},\, t, \tag{13}$$

where we have small error terms of order $\mathfrak{E}\{\tilde{\mathrm{x}}^2\}$, $\mathfrak{E}\{\tilde{\mathrm{u}}^2\}$, $\mathfrak{E}\{\tilde{\mathrm{x}}\tilde{\mathrm{u}}\}$ and higher (the form of the error terms is actually much more complicated). These error terms represent a **coupling** between solutions of the *stochastic* and the *deterministic* problems that does not exist for the linear case discussed in chapter III. By using (12) we obtain the approximation:

$$\bar{\mathrm{f}}(\mathbf{x}, \mathbf{u}, t) \approx \mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}, t). \tag{14}$$

Now, if we **assume** that the expectation and differentiation operations can be interchanged, so that[2]:

$$\dot{\bar{\mathbf{x}}}^a(t) \triangleq \frac{d}{dt}\mathfrak{E}\{\mathbf{x}^a(t)\} = \mathfrak{E}\{\, \frac{d}{dt}\mathbf{x}^a(t)\,\} \tag{15}$$

and similarly for $\underset{\cdot}{\mathrm{x}}^s(t)$, by using (14) for $\mathbf{f}^s$, $\mathbf{f}^a$, $\mathbf{g}^s$, $\mathbf{g}^\ell$, $\mathbf{g}^a$, the approximate system equations for the *deterministic* problem can be written as:

$$\bar{\mathcal{S}}^s: \qquad \begin{aligned} \dot{\bar{\mathbf{x}}}^s(t) &= \mathbf{f}^s(\bar{\mathbf{x}}^s(t), \bar{\mathbf{u}}^s(t), t) && (16)\\ \bar{\mathbf{y}}^s(t) &= \mathbf{g}^s(\bar{\mathbf{x}}^s(t), \bar{\mathbf{u}}^s(t), t) && (17)\\ \bar{\mathbf{u}}^\ell(t) &= \mathbf{g}^\ell(\bar{\mathbf{x}}^s(t), \bar{\mathbf{u}}^s(t), t) && (18) \end{aligned}$$

[1] (13) are sufficient conditions for a continuous map from **u** to **x** under **f**.

[2] A sufficient condition is that $|\mathbf{x}^a(t)| < \infty$ and $|\dot{\mathbf{x}}^a(t)| < \infty$.

$$\bar{S}^a: \quad \begin{aligned} \dot{\bar{\mathbf{x}}}^a(t) &= \mathbf{f}^a(\bar{\mathbf{x}}^a(t), \bar{\mathbf{u}}^a(t), t) && (19) \\ \bar{\mathbf{y}}^a(t) &= \mathbf{g}^a(\bar{\mathbf{x}}^a(t), \bar{\mathbf{u}}^a(t), t) && (20) \end{aligned}$$

and the approximate system equations for the *Stochastic* problem are:

$$\tilde{S}^s: \quad \begin{aligned} \dot{\tilde{\mathbf{x}}}^s(t) &= \left.\frac{\partial \mathbf{f}^s}{\partial \mathbf{x}^s}\right|_{\bar{\mathbf{x}}^s(t),\bar{\mathbf{u}}^s(t)} \tilde{\mathbf{x}}^s(t) + \left.\frac{\partial \mathbf{f}^s}{\partial \mathbf{u}^s}\right|_{\bar{\mathbf{x}}^s(t),\bar{\mathbf{u}}^s(t)} \tilde{\mathbf{u}}^s(t) && (21) \\ \tilde{\mathbf{y}}^s(t) &= \left.\frac{\partial \mathbf{g}^s}{\partial \mathbf{x}^s}\right|_{\bar{\mathbf{x}}^s(t),\bar{\mathbf{u}}^s(t)} \tilde{\mathbf{x}}^s(t) + \left.\frac{\partial \mathbf{g}^s}{\partial \mathbf{u}^s}\right|_{\bar{\mathbf{x}}^s(t),\bar{\mathbf{u}}^s(t)} \tilde{\mathbf{u}}^s(t) && (22) \\ \tilde{\mathbf{u}}^\ell(t) &= \left.\frac{\partial \mathbf{g}^\ell}{\partial \mathbf{x}^s}\right|_{\bar{\mathbf{x}}^s(t),\bar{\mathbf{u}}^s(t)} \tilde{\mathbf{x}}^s(t) + \left.\frac{\partial \mathbf{g}^\ell}{\partial \mathbf{u}^s}\right|_{\bar{\mathbf{x}}^s(t),\bar{\mathbf{u}}^s(t)} \tilde{\mathbf{u}}^s(t) && (23) \end{aligned}$$

$$\tilde{S}^a: \quad \begin{aligned} \dot{\tilde{\mathbf{x}}}^a(t) &= \left.\frac{\partial \mathbf{f}^a}{\partial \mathbf{x}^a}\right|_{\bar{\mathbf{x}}^a(t),\bar{\mathbf{u}}^a(t)} \tilde{\mathbf{x}}^a(t) + \left.\frac{\partial \mathbf{f}^a}{\partial \mathbf{u}^a}\right|_{\bar{\mathbf{x}}^a(t),\bar{\mathbf{u}}^a(t)} \tilde{\mathbf{u}}^a(t) && (24) \\ \tilde{\mathbf{y}}^a(t) &= \left.\frac{\partial \mathbf{g}^a}{\partial \mathbf{x}^a}\right|_{\bar{\mathbf{x}}^a(t),\bar{\mathbf{u}}^a(t)} \tilde{\mathbf{x}}^a(t) + \left.\frac{\partial \mathbf{g}^a}{\partial \mathbf{u}^a}\right|_{\bar{\mathbf{x}}^a(t),\bar{\mathbf{u}}^a(t)} \tilde{\mathbf{u}}^a(t) && (25) \end{aligned}$$

In the special case where we have the *quadratic* cost function:

$$\mathbf{e}(t) = \mathbf{y}^a(t) - \mathbf{y}^s(t)\,, \tag{III.6}$$

$$J = \mathfrak{E}\{\lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \mathbf{e}^T(t) Q \mathbf{e}(t) + \rho \mathbf{u}^{\ell^T}(t) R \mathbf{u}^\ell(t)\, dt\} \tag{III.7}$$

we can solve the deterministic and the stochastic problems separately. We first have to solve the deterministic problem to find the "operating point" $\bar{\mathbf{x}}(t)$ and $\bar{\mathbf{u}}(t)$ for the s, ℓ and a superscripts. The corresponding cost functions are given by:

$$\bar{J} = \lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \bar{\mathbf{e}}^T(t) Q \bar{\mathbf{e}}(t) + \rho \bar{\mathbf{u}}^{\ell^T}(t) R \bar{\mathbf{u}}^\ell(t)\, dt \tag{III.19}$$

and

$$\tilde{J} = \mathfrak{E}\{\lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \tilde{\mathbf{e}}^T(t) Q \tilde{\mathbf{e}}(t) + \rho \tilde{\mathbf{u}}^{\ell^T}(t) R \tilde{\mathbf{u}}^\ell(t)\, dt\}\,. \tag{III.20}$$

Note that the *stochastic* problem that we have just formulated is the same one that we solved in chapter III (see equations (III.21)–(III.25)), since (21)–(25) are also linear differential equations. This approximate separation method should work well if the stochastic variations are small enough compared to the expected values, i.e. if for all elements $\mathbf{u}_i^a$, $\mathbf{u}_j^a$, of the ensemble $\mathcal{U}^a$ we have for every time t:

$$\left|\mathbf{u}_i^a(t) - \mathbf{u}_j^a(t)\right| < \bar{\mathbf{u}}^a(t) . \tag{26}$$

This condition does not hold for many flight simulator motion design problems, but may hold for some robotic applications of this theory.

Iterative Solution

A better approximation for the overall solution can be obtained by solving the exact system equations for the deterministic problem, which are the expected values of (4)–(8):

$$\mathcal{S}^s: \qquad \begin{aligned} \dot{\bar{\mathbf{x}}}^s(t) &= \bar{\mathbf{f}}^s(\mathbf{x}^s(t), \mathbf{u}^s(t), t) && (27)\\ \bar{\mathbf{y}}^s(t) &= \bar{\mathbf{g}}^s(\mathbf{x}^s(t), \mathbf{u}^s(t), t) && (28)\\ \bar{\mathbf{u}}^\ell(t) &= \bar{\mathbf{g}}^\ell(\mathbf{x}^s(t), \mathbf{u}^s(t), t) && (29) \end{aligned}$$

$$\mathcal{S}^a: \qquad \begin{aligned} \dot{\bar{\mathbf{x}}}^a(t) &= \bar{\mathbf{f}}^a(\mathbf{x}^a(t), \mathbf{u}^a(t), t) && (30)\\ \bar{\mathbf{y}}^a(t) &= \bar{\mathbf{g}}^a(\mathbf{x}^a(t), \mathbf{u}^a(t), t) && (31) \end{aligned}$$

These equations are **coupled** to the solution of the stochastic problem since the evaluation of $\bar{\mathbf{f}}^s$, $\bar{\mathbf{g}}^s$ and $\bar{\mathbf{g}}^\ell$ depends on the stochastic solution for $\tilde{\mathbf{x}}^s$ and $\tilde{\mathbf{u}}^s$, which, in turn, depends on the "operating point", $(\bar{\mathbf{x}}^s(t)$, $\bar{\mathbf{u}}^s(t)$, $\bar{\mathbf{x}}^a(t)$, $\bar{\mathbf{u}}^a(t))$ calculated from (27)–(31), (III.19). One method to solve these equations is by iteration between the deterministic and the stochastic problems by using the following algorithm.

Iteration Algorithm

1. Initialization:
   (i) Calculate the initial "operating point" from the deterministic problem solution of (16)–(20), (III.19).

(ii) Calculate the stochastic problem solution of (21)–(25), (III.20).

2. Iteration:

   (i) Calculate a new "operating point" from the deterministic solution of (27)–(31), (III.19), using the previous stochastic solution.

   (ii) Calculate a new stochastic solution of (21)–(25), (III.20), using the new "operating point".

3. Termination: When a properly defined norm of the change in the "operating point" between iterations is small enough, then it is considered that the solution has converged.

In the case when the iterations converge (as is likely, **but not proved**) we obtain a solution to (4)–(8) and the cost J (III.7) by using the approximation:

$$\mathbf{f}(\mathbf{x},\mathbf{u},t) \approx \bar{\mathbf{f}}(\mathbf{x},\mathbf{u},t) + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\bigg|_{\bar{\mathbf{x}},\bar{\mathbf{u}}} \tilde{\mathbf{x}} + \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\bigg|_{\bar{\mathbf{x}},\bar{\mathbf{u}}} \tilde{\mathbf{u}} \tag{32}$$

rather than (12) and (14).

Improved Iterative Solution

A further solution improvement can be obtained by *statistical linearization* [Gelb74] which is the following "best" expected linear approximation of $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{u}}$ rather than (32):

$$\mathbf{f}(\mathbf{x},\mathbf{u},t) \approx \bar{\mathbf{f}}(\mathbf{x},\mathbf{u},t) + \overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{x}}}\tilde{\mathbf{x}} + \overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{u}}}\tilde{\mathbf{u}}\ . \tag{33}$$

We present here two possible "best" choices for the "parameters" $\overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{x}}}(t)$ and $\overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{u}}}(t)$ defined in the Least Square Error sense over the ensemble $\mathcal{U}^a$. The first choice is to find the "parameters" that minimize

$$\min\ \mathfrak{E}\{\left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}} - \overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{x}}}\right)^2 + \left(\frac{\partial \mathbf{f}}{\partial \mathbf{u}} - \overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{u}}}\right)^2\} \tag{34}$$

over the ensemble $\mathcal{U}^a$. i.e.,

$$\overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{x}}} = \mathfrak{E}\{\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\}, \tag{35}$$

$$\overline{\frac{\Delta \mathbf{f}}{\Delta \mathbf{u}}} = \mathfrak{E}\{\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\}. \tag{36}$$

The second choice is to find the "parameters" that minimize

$$\min \mathfrak{E}\{\left(\mathbf{f} - \bar{\mathbf{f}} - \frac{\overline{\Delta \mathbf{f}}}{\Delta \mathbf{x}}\tilde{\mathbf{x}} - \frac{\overline{\Delta \mathbf{f}}}{\Delta \mathbf{u}}\tilde{\mathbf{u}}\right)^2\} \tag{37}$$

over the ensemble $\mathcal{U}^a$. i.e. (Appendix A),

$$\frac{\overline{\Delta \mathbf{f}}}{\Delta \mathbf{x}} = \frac{\mathfrak{E}\{\mathbf{f}\tilde{\mathbf{x}}\}\mathfrak{E}\{\tilde{\mathbf{u}}^2\} - \mathfrak{E}\{\mathbf{f}\tilde{\mathbf{u}}\}\mathfrak{E}\{\tilde{\mathbf{x}}\tilde{\mathbf{u}}\}}{\mathfrak{E}\{\tilde{\mathbf{x}}^2\}\mathfrak{E}\{\tilde{\mathbf{u}}^2\} - \mathfrak{E}^2\{\tilde{\mathbf{x}}\tilde{\mathbf{u}}\}}, \tag{38}$$

$$\frac{\overline{\Delta \mathbf{f}}}{\Delta \mathbf{u}} = \frac{\mathfrak{E}\{\mathbf{f}\tilde{\mathbf{u}}\}\mathfrak{E}\{\tilde{\mathbf{x}}^2\} - \mathfrak{E}\{\mathbf{f}\tilde{\mathbf{x}}\}\mathfrak{E}\{\tilde{\mathbf{x}}\tilde{\mathbf{u}}\}}{\mathfrak{E}\{\tilde{\mathbf{x}}^2\}\mathfrak{E}\{\tilde{\mathbf{u}}^2\} - \mathfrak{E}^2\{\tilde{\mathbf{x}}\tilde{\mathbf{u}}\}}. \tag{39}$$

The first choice seems to be best suited to our problem since we are trying to find the best linear description for (40)–(44).

If we use (33) and the parameters (35)–(36) or (38)–(39) we can rewrite the stochastic system equations (21)–(25), as:

$$\dot{\tilde{\mathbf{x}}}^s(t) = \frac{\overline{\Delta \mathbf{f}^s}}{\Delta \mathbf{x}^s}(t)\,\tilde{\mathbf{x}}^s(t) + \frac{\overline{\Delta \mathbf{f}^s}}{\Delta \mathbf{u}^s}(t)\,\tilde{\mathbf{u}}^s(t) \tag{40}$$

$$\tilde{\mathcal{S}}^s: \qquad \tilde{\mathbf{y}}^s(t) = \frac{\overline{\Delta \mathbf{g}^s}}{\Delta \mathbf{x}^s}(t)\,\tilde{\mathbf{x}}^s(t) + \frac{\overline{\Delta \mathbf{g}^s}}{\Delta \mathbf{u}^s}(t)\,\tilde{\mathbf{u}}^s(t) \tag{41}$$

$$\tilde{\mathbf{u}}^\ell(t) = \frac{\overline{\Delta \mathbf{g}^\ell}}{\Delta \mathbf{x}^\ell}(t)\,\tilde{\mathbf{x}}^\ell(t) + \frac{\overline{\Delta \mathbf{g}^\ell}}{\Delta \mathbf{u}^\ell}(t)\,\tilde{\mathbf{u}}^\ell(t) \tag{42}$$

$$\tilde{\mathcal{S}}^a: \qquad \dot{\tilde{\mathbf{x}}}^a(t) = \frac{\overline{\Delta \mathbf{f}^a}}{\Delta \mathbf{x}^a}(t)\,\tilde{\mathbf{x}}^a(t) + \frac{\overline{\Delta \mathbf{f}^a}}{\Delta \mathbf{u}^a}(t)\,\tilde{\mathbf{u}}^a(t) \tag{43}$$

$$\tilde{\mathbf{y}}^a(t) = \frac{\overline{\Delta \mathbf{g}^a}}{\Delta \mathbf{x}^a}(t)\,\tilde{\mathbf{x}}^a(t) + \frac{\overline{\Delta \mathbf{g}^a}}{\Delta \mathbf{u}^a}(t)\,\tilde{\mathbf{u}}^a(t) \tag{44}$$

with the same cost J (III.20). Note that for a nonlinear $\mathbf{f}^s(\mathbf{x}^s)$, equations (40)–(44) are not linear since the system matrices like $\frac{\overline{\Delta \mathbf{f}^s}}{\Delta \mathbf{x}^s}(t)$ depend on the expected solution of $\tilde{\mathbf{x}}^s(t)$. Assuming "small" dependence of the system matrices on $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{u}}$, an approximate solution to the new stochastic problem (40)–(44) and cost J

(III.20) can be obtained by iterations (convergence has to proved!). Although a deterministic-stochastic solution separation does not exist here, a deterministic-stochastic iteration procedure (as before) seems likely to converge. In estimation theory it is claimed by Gelb [Gelb74] that using this *statistical linearization* is superior to the *extended Kalman filter* method (equivalent to our first approach (16)-(25)).

Although the deterministic-stochastic separation does not hold for a non-quadratic cost function we can use a more general cost function form due to the nonlinear output functions $\mathbf{g}^s$, $\mathbf{g}^a$ and $\mathbf{g}^\ell$. In the next section we obtain a design approach for nonquadratic cost functions.

2. Pseudo Linear Quadratic Control

PLQ control is a generalization of the standard L.Q. control, where we consider a *pseudo-quadratic*, *pseudo-linear* and a *pseudo-optimization* problem. First, the cost weighting matrices Q and R are made functions of the system state, $\mathbf{x}$, i.e.:

$$J \triangleq \int_0^T \mathbf{x}^T(t) Q(\mathbf{x}(t))\, \mathbf{x}(t) + \mathbf{u}^T(t) R(\mathbf{x}(t))\, \mathbf{u}(t)\, dt \,, \tag{45}$$

where (45) corresponds to a nonquadratic cost function (*pseudo-quadratic*). Second, our plant is allowed to be nonlinear (*pseudo-linear*). Thus PLQ control allows a simple, more flexible design procedure with specifications closely related to (45).

In the **special case** when (45) is equivalent to a positive definite form, where:

$$\mathbf{x}^T Q(\mathbf{x})\, \mathbf{x} = \sum_{i=1}^{\eta} \left(\mathbf{x}^T Q_i \mathbf{x}\right)^i , \quad Q_i\text{'s are constants} \quad \text{and} \quad R \text{ is a constant}\,, \tag{46}$$

then the optimal control for a **linear plant** results in a nonlinear feedback control law [Buss66, Kosut79, Sandor77]. By comparison, our solution also results in a nonlinear feedback control law but it **is not** the solution of an optimization problem (corresponds therefore to a *pseudo-optimization*), and the cost (45) is at best an approximation to the real cost being optimized; **if** such a cost exists. Our solution method is to compute the standard L.Q. steady state feedback gain F for each value of $Q(\mathrm{x})$ and $R(\mathrm{x})$ as if $Q(\mathrm{x})$ and $R(\mathrm{x})$ were constant, but then to use the gain F that corresponds to each measured state x, rather then a constant gain. We state the conditions on $Q(\mathrm{x})$, $R(\mathrm{x})$, $A(\mathrm{x})$ and $B(\mathrm{x})$ that guarantee that the closed-loop system is stable in the scalar case; for the matrix case we only conjecture such conditions.

Advantages of PLQ:

(i) The feedback implementation that we obtain is usually less cumbersome to evaluate in real time, see example in subsection 2.3. Furthermore, for physically measured states $\mathrm{x}(t)$ the feedback control can be computed using only fixed point arithmetic.

(ii) The *pseudo-cost* function used is more general than (46) [Kosut79], see equation (81) of the example in subsection 2.3.

(iii) The computation of the feedback function requires the solution of smaller dimension matrix equations.

(iv) PLQ control extends to the control of nonlinear plants as shown in subsection 2.4–2.5.

Disadvantages of PLQ:

(i) Conditions on $Q(\mathrm{x})$ and $R(\mathrm{x})$ that guarantee closed-loop stability in the **matrix** case have not yet been established (only a conjecture is available).

(ii) The feedback law is not a solution to an optimization problem for any given cost, although our *pseudo-quadratic cost* may be an approximation to such a cost.

The presentation is organized as follows:

1. Results for the linear *pseudo-quadratic* case.
2. Stability proof for the linear *pseudo-quadratic* case.
3. Example of PLQ control for a linear plant.
4. PLQ control of nonlinear plants.
5. Example of PLQ control for a nonlinear plant.

2.1. Result for the linear case

Definitions

We consider the following linear, time invariant, finite dimensional, stabilizable system:

$$\dot{\mathrm{x}}(t) = A\mathrm{x}(t) + B\mathrm{u}(t) \tag{47}$$

where $\mathrm{x}(t)$ is the state and $\mathrm{u}(t)$ is the input, with corresponding dimensions n and m. Then, the feedback control law is given by:

$$\mathrm{u}(t) = -F(\mathrm{x})\,\mathrm{x}(t) \tag{48}$$

where the gain $F(\mathrm{x})$ is computed by:

$$F(\mathrm{x}) = R^{-1}(\mathrm{x})\,B^T P(\mathrm{x}) \tag{49}$$

and where $P(\mathrm{x})$ is the unique positive definite $(P(\mathrm{x}) > 0)$ solution of the following Algebraic Riccati Equation (Are):

$$-P(\mathrm{x})BR^{-1}(\mathrm{x})B^T P(\mathrm{x}) + P(\mathrm{x})A + A^T P(\mathrm{x}) + Q(\mathrm{x}) = 0\,. \tag{50}$$

We assume that

$$Q(\mathrm{x}) \geq 0 \qquad \text{and} \qquad R(\mathrm{x}) > 0 \tag{51}$$

and that the pair $(A, \sqrt{Q(\mathrm{x})}\,)$ is detectable for all $\mathbf{x}$.

Scalar Result

In the scalar case the closed-loop system:

$$\dot{x}(t) = (a - bf(x))x(t)\,. \tag{52}$$

is globally stable when $q(x)$ and $r(x)$ are chosen such that:

$$(2q + q_x x) + \left(\frac{a}{b}\right)^2 \left(1 + \sqrt{1 + \frac{q}{r}\left(\frac{b}{a}\right)^2}\,\right)^2 (2r + r_x x) > 0 \qquad \text{for all } x \blacksquare \tag{53}$$

A sufficient condition for global stability of (52) is:

$$\frac{dq}{dx}x \geq 0 \qquad \text{and} \qquad \frac{dr}{dx}x \geq 0\,. \tag{54}$$

Matrix Results

Given the following closed-loop system:

$$\dot{\mathrm{x}}(t) = (A - BF(\mathrm{x}))\mathrm{x}(t)\,. \tag{55}$$

we prove that there is **only one equilibrium** point $\mathbf{x} = \mathbf{0}$ ▮

We further prove that the equilibrium point is **locally stable** for any $Q(\mathbf{x}) \geq 0$ and $R(\mathbf{x}) > 0$ that has finite partial derivatives with respect to all the states x_i ▮

Global stability is conjectured for the following condition which is similar to the scalar condition (53):

$$2Q(\mathbf{x}) + \sum_{i=1}^{n} \frac{\partial Q}{\partial x_i} x_i + P(\mathbf{x})B\left(2R^{-1}(\mathbf{x}) - \sum_{i=1}^{n} \frac{\partial R^{-1}}{\partial x_i} x_i\right)B^T P(\mathbf{x}) > 0 \; ▮ \tag{56}$$

A conjectured sufficient condition is:

$$\sum_{i=1}^{n} \frac{\partial Q}{\partial x_i} x_i \geq 0 \qquad \text{and} \qquad -\sum_{i=1}^{n} \frac{\partial R^{-1}}{\partial x_i} x_i \geq 0\,. \tag{57}$$

Generally this condition requires the *pseudo cost* (45) to increase for nonzero states $\mathbf{x}$, and thus increases the "attraction" of the only equilibrium point $\mathbf{x} = \mathbf{0}$. Thus the closed-loop system (55) becomes "more stable" for large $\mathbf{x}$.

2.2. Stability proof for the linear case

Let us define the Closed-Loop system matrix A_{CL} as:

$$A_{\mathrm{CL}} \triangleq A - BR^{-1}(\mathbf{x})B^T P(\mathbf{x}) = A - BF(\mathbf{x})\,. \tag{58}$$

Then, choose the scalar Lyapunov function:

$$v = \mathbf{x}^T P(\mathbf{x})\,\mathbf{x}\,. \tag{59}$$

By construction $v > 0$ for all $\mathbf{x}$ since $P(\mathbf{x})$ is the unique positive definite solution of the ARE (50). In order to prove the stability of (55) we have to show that for all trajectories $\mathbf{x}$,

$$\dot{v} < 0\,. \tag{60}$$

To compute $\dot{v}$ we use the chain rule, i.e.

$$\dot{v} = \dot{\mathbf{x}}^T P(\mathbf{x})\,\mathbf{x} + \mathbf{x}^T P(\mathbf{x})\,\dot{\mathbf{x}} + \mathbf{x}^T\left(\frac{\partial P}{\partial \mathbf{x}} \cdot \dot{\mathbf{x}}\right)\mathbf{x}\,. \tag{61}$$

Note that $\frac{\partial P}{\partial \mathbf{x}} \cdot \dot{\mathbf{x}}$ is a matrix and $\frac{\partial P}{\partial \mathbf{x}}$ has triple indexed elements where the dot, "$\cdot$", multiplication should be interpreted as:

$$\frac{\partial P}{\partial \mathbf{x}} \cdot \dot{\mathbf{x}} = \sum_{i=1}^{n} \frac{\partial P}{\partial x_i} \frac{dx_i}{dt} . \tag{62}$$

We substitute the expression for $\dot{\mathbf{x}}$ (55) into (61) and use definition (58) to obtain:

$$\dot{v} = \mathbf{x}^T A_{\mathrm{CL}}^T P(\mathbf{x})\,\mathbf{x} + \mathbf{x}^T P(\mathbf{x}) A_{\mathrm{CL}}\,\mathbf{x} + \mathbf{x}^T \left(\frac{\partial P}{\partial \mathbf{x}} \cdot \dot{\mathbf{x}} \right) \mathbf{x} \tag{63}$$

and thus to prove the stability we have to show that

$$-A_{\mathrm{CL}}^T P(\mathbf{x}) - P(\mathbf{x}) A_{\mathrm{CL}} - \frac{\partial P}{\partial \mathbf{x}} \cdot \dot{\mathbf{x}} > 0 , \tag{64}$$

or, using the ARE (50) and (58) we must prove that,

$$Q(\mathbf{x}) + P(\mathbf{x}) B R^{-1}(\mathbf{x})\, B^T P(\mathbf{x}) - \frac{\partial P}{\partial \mathbf{x}} \cdot \dot{\mathbf{x}} > 0 . \tag{65}$$

By construction all the eigenvalues of the matrix A_{CL} are negative, i.e.

$$\text{eigenvalues}(A_{\mathrm{CL}}(\mathbf{x})) < 0 \qquad \text{for all } \mathbf{x} , \tag{66}$$

so that the determinant of A_{CL} is nonzero for any $\mathbf{x}$. Therefore the only equilibrium point, $\dot{\mathbf{x}} = \mathbf{0}$, of the closed-loop system (55) is $\mathbf{x} = \mathbf{0}$ ▮

Furthermore, (65) proves that the closed-loop system (55) is locally asymptotically stable for $Q(\mathbf{x}) \geq 0$ and $R(\mathbf{x}) > 0$, assuming that $\partial P/\partial x_i$ is finite for all x_i (recall that by construction $P(\mathbf{x}) > 0$). Since A_{CL} is bound thus $\dot{\mathbf{x}}$ is small if $\mathbf{x}$ is small ▮

As shown latter on from (71)–(72), the condition for $\partial P/\partial x_i$ to be finite is that both $\partial Q/\partial x_i$ and $\partial R^{-1}/\partial x_i$ are finite.

Let us continue the derivation for the general case, $\dot{\mathbf{x}} \neq \mathbf{0}$, and compute the partial derivative P_{x_i} for each x_i of the vector $\mathbf{x}$:

$$P_{x_i} \equiv \frac{\partial P}{\partial x_i} \tag{67}$$

using the ARE in (50),

$$-P_{x_i}BR^{-1}B^TP - PBR^{-1}B^TP_{x_i} - PB(R^{-1})_{x_i}B^TP + P_{x_i}A + A^TP_{x_i} + Q_{x_i} = 0$$

now collecting terms and using (58) we obtain:

$$P_{x_i}A_{\text{CL}} + A_{\text{CL}}^TP_{x_i} + Q_{x_i} - PB(R^{-1})_{x_i}B^TP = 0 \tag{68}$$

where we define

$$Q_{x_i} \equiv \frac{\partial Q}{\partial x_i}, \qquad (R^{-1})_{x_i} \equiv \frac{\partial R^{-1}}{\partial x_i}\,. \tag{69}$$

Let us define the matrix S_i as:

$$S_i \triangleq Q_{x_i} - PB(R^{-1})_{x_i}B^TP \tag{70}$$

for each element x_i of the vector $\mathbf{x}$. Notice that S_i is a symmetric matrix and

$$S_i > 0 \qquad \textbf{if both} \qquad Q_{x_i} > 0 \quad \text{and} \quad -(R^{-1})_{x_i} > 0\,, \tag{71}$$

and **negative** definite ($S_i < 0$) **if** both matrices in (71) are negative definite.

Now substituting S_i of (70) into (68) we obtain the following n Lyapunov matrix equations in the unknowns P_{x_i}, which correspond to each x_i of $\mathbf{x}$,

$$P_{x_i}(A - BR^{-1}B^TP) + (A - BR^{-1}B^TP)^TP_{x_i} + S_i = 0\,,$$

or

$$P_{x_i}A_{\text{CL}} + A_{\text{CL}}^TP_{x_i} + S_i = 0\,. \tag{72}$$

Recall that the closed-loop system matrix of (55), A_{CL} (58), is asymptotic stable for every value of its state $\mathbf{x}$, since by construction, P is the unique positive definite solution of the ARE (50). Thus by the Lyapunov theorem

$$P_{x_i}(\mathbf{x}) > 0 \quad \textbf{iff} \quad S_i(\mathbf{x}) > 0\,, \tag{73}$$

and

$$-P_{x_i}(\mathrm{x}) > 0 \quad \text{iff} \quad -S_i(\mathrm{x}) > 0\,. \tag{74}$$

Let us substitute the system differential equations (55) into (64), to obtain a new form for the closed-loop stability condition for (55):

$$-A_{\mathrm{CL}}^T(\mathrm{x})P(\mathrm{x}) - P(\mathrm{x})A_{\mathrm{CL}}(\mathrm{x}) - \sum_{i=1}^{n} P_{x_i}(\mathrm{x})\,(A_{\mathrm{CL}}(\mathrm{x})\,\mathrm{x})_i \; > \; 0\,. \tag{75}$$

We expect (not proved) from (75), using (71) and (73)–(74), that a sufficient condition for stability of (55) would be:

$$Q_{x_i}x_i \geq 0 \qquad \text{and} \qquad -(R^{-1})_{x_i}x_i \geq 0 \qquad \text{for all } x_i \text{ of } \mathrm{x} \; \blacksquare \tag{76}$$

Scalar Global Stability Proof

From this point, we consider only the **scalar case**, where (75) becomes:

$$-a_{\mathrm{CL}}(x)\,(2p(x) + p_x(x)x) > 0\,. \tag{77}$$

Now by computing $a_{\mathrm{CL}}(x)p_x(x)$ from (68) and $a_{\mathrm{CL}}(x)p(x)$ from the ARE (50) and using $r_x/r^2 = -(r^{-1})_x$ we obtain:

$$(2q + q_x x) + \frac{p^2b^2}{r^2}(2r + r_x x) > 0 \qquad \text{for all } x\,. \tag{78}$$

Then, by using the ARE (50) to compute $p(x)$ and substituting the result ($p(x) > 0$) into (78) we obtain:

$$(2q + q_x x) + \left(\frac{a}{b}\right)^2\left(1 + \sqrt{1 + \frac{q}{r}\left(\frac{b}{a}\right)^2}\,\right)^2 (2r + r_x x) > 0 \qquad \text{for all } x\,. \tag{79}$$

By construction $r(x) > 0$ and $q(x) \geq 0$, so that a sufficient condition for the stability of (55) in the scalar case is that:

$$q_x x \geq 0 \qquad \text{and} \qquad r_x x \geq 0\,. \tag{80}$$

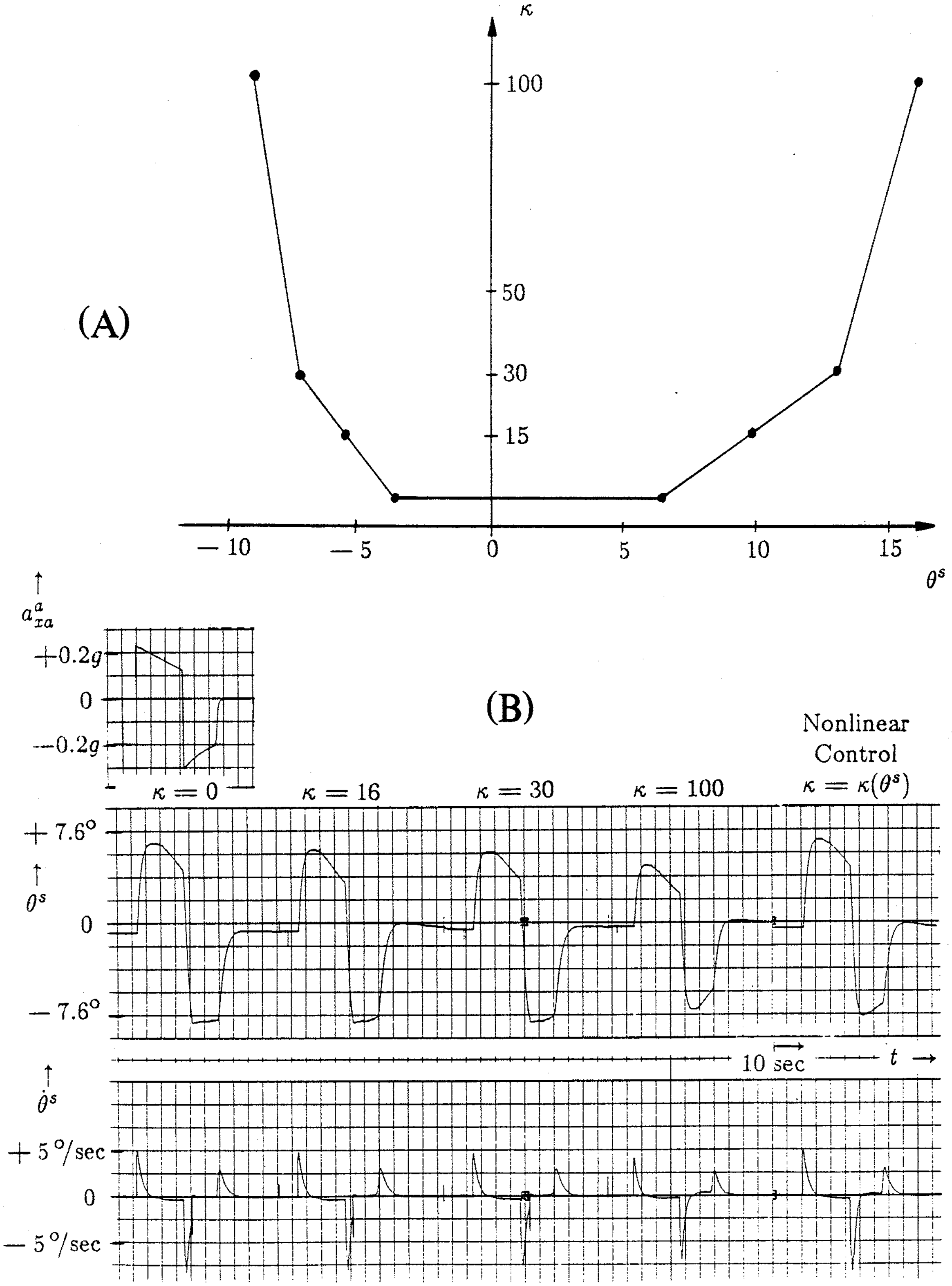


Figure 1. Nonlinear OWS. (A) Cost function "parameter" $\kappa(\theta^s)$ in (81). (B) Comparison of the nonlinear OWS for inputs a^a_{xa}, $\theta^a = 0$, $\dot{\theta}^a = 0$, for $\kappa = \kappa(\theta^s)$ in (A) to four linear ones $\kappa = 0, 16, 30, 100$.

2.3. Example of PLQ control for a linear plant

An example of the above theory was implemented on the GAT-1 flight simulator for the control of the pitch axis, θ^s, motion. The full equations describing this example are given in chapter VII, a brief description follows. The closed-loop portion of the system consists of four states: two of the motion-base $\theta^s(t)$, $\dot{\theta}^s(t)$ and two of the pilot's vestibular model x_{oto}, x_{scc}, which are used to compute the two vestibular errors e_{scc} and e_{oto}. The pitch motion is controlled by an electric motor with an angular velocity command input, $\dot{\theta}^c$. The simulator cab's center of gravity is above the pitch pivot and thus we have unstable open-loop motion-base dynamics.

The following nonquadratic cost function, J, was used to better approximate the hard bounds of the pitch motion ($-9 < \theta^s < 17$ degrees) and the asymmetry in these bounds:

$$J = \mathfrak{E}\{e_{\text{scc}}^2 + e_{\text{oto}}^2 + \rho\dot{\theta}^{c2} + \kappa(\theta^s)\theta^{s2}\} \tag{81}$$

where the function $\kappa(\theta^s)$ is shown in Figure 1(A). The nonlinear feedback law used was:

$$\dot{\theta}^c = -F(\theta^s)\begin{pmatrix} x_{\text{oto}} \\ x_{\text{scc}} \\ \theta^s \\ \dot{\theta}^s \end{pmatrix} \tag{82}$$

where the feedback function $F(\theta^s)$ was approximated by the interpolation between four values, computed for $\kappa = 0, 16, 30, 100$.

The experimental results are shown in Figure 1(B), where we compare the simulator response to the same input for five cases: $\kappa = 0, 16, 30, 100 \Rightarrow$ linear control and nonlinear control using $\kappa(\theta^s)$ (Figure 1(A)).

Looking at Figure 1(B) we notice the following:

1. The simulator hit the lower end stop for $\kappa = 0, 16, 30$ but not for $\kappa = 100$ and $\kappa = \kappa(\theta^s)$.
2. Comparing the two responses that did not hit the end stop $\kappa = 100$ and the nonlinear $\kappa = \kappa(\theta^s)$, we obtain "more motion" using the nonlinear control, moreover the positive response is as large as for the smallest $\kappa = 0$ linear control.

Since the interpolation of just four computed values for $F(\theta^s)$ was sufficient a very economical control system implementation is suggested from this example. This implementation uses a table lookup combined with interpolations for the function $F(\cdot)$. For example in a six degrees-of-freedom simulator where the matrix $F(\cdot)$ has 252 elements (6 controls times 42 states Section III.5) and we use ten interpolating points for Q(of 6 variables) we require only a table of 15,120 numbers.

2.4. PLQ Control of Nonlinear Plants

Definitions

Given a **nonlinear**, time invariant, finite dimensional system that can be written in the following state form:

$$\dot{\mathbf{x}}(t) = A(\mathbf{x})\,\mathbf{x}(t) + B(\mathbf{x})\,\mathbf{u}(t) \tag{83}$$

where $\mathbf{x}(t)$ is the state and $\mathbf{u}(t)$ is the input, with corresponding dimensions n and m. Furthermore given the feedback control law:

$$\mathbf{u}(t) = -F(\mathbf{x})\,\mathbf{x}(t) \tag{84}$$

where $F(\mathbf{x})$ is computed by:

$$F(\mathbf{x}) = R^{-1}(\mathbf{x})B^T(\mathbf{x})P(\mathbf{x}) \tag{85}$$

and $P(\mathbf{x})$ is the unique positive definite solution, $P(\mathbf{x}) > 0$, of the following ALGEBRAIC RICCATI EQUATION (ARE):

$$-P(\mathbf{x})\,B(\mathbf{x})\,R^{-1}(\mathbf{x})\,B^T(\mathbf{x})\,P(\mathbf{x}) + P(\mathbf{x})\,A(\mathbf{x}) + A^T(\mathbf{x})\,P(\mathbf{x}) + Q(\mathbf{x}) = 0 \tag{86}$$

where

$$Q(\mathbf{x}) \geq 0 \qquad \text{and} \qquad R(\mathbf{x}) > 0\,. \tag{87}$$

Scalar Case Result

The following scalar closed-loop system of (83) and feedback law (84), is globally stable **in the scalar case:**

$$\dot{x}(t) = (a(x) - b(x)f(x))x(t)\,. \tag{88}$$

when $q(x)$ and $r(x)$ are chosen such that (89) is true for all x_i;

$$(2q + \frac{dq}{dx}x) + \frac{p^2b^2}{r^2}(2r + \frac{dr}{dx}x) + 2p\Big(\frac{da}{dx} - \frac{db}{dx}r^{-1}bp\Big)x > 0 \blacksquare \tag{89}$$

Note that this condition is similar to (78).

Matrix Results

Given the following closed-loop system of (83) and feedback law (84) i.e.

$$\dot{\mathrm{x}}(t) = (A(\mathrm{x}) - B(\mathrm{x})F(\mathrm{x}))\mathrm{x}(t)\,. \tag{90}$$

we prove that there is **only one equilibrium** point $\mathbf{x} = \mathbf{0}$ ▮

We further prove that the equilibrium point is **locally stable** for any $Q(\mathrm{x})\geq 0$ and $R(\mathrm{x})>0$ such that S_i (defined in (93)) is finite for all indices i of the state x_i ▮

Global stability is conjectured for the following plausible condition that parallels the scalar condition (88):

$$2\Big(Q(\mathrm{x}) + P(\mathrm{x})B(\mathrm{x})R^{-1}(\mathrm{x})\,B^T(\mathrm{x})P(\mathrm{x})\Big) + \sum_{i=1}^{n} S_i x_i > 0 \blacksquare \tag{91}$$

where we define S_i in (93).

Proof

Repeating the same steps as before (58)–(69) and defining a more general S_i as:

$$S_i \triangleq PA_{x_i} + A_{x_i}^T P - P(B_{x_i}R^{-1}B^T + BR^{-1}B_{x_i}^T)P + Q_{x_i} - PB(R^{-1})_{x_i}B^TP \tag{92}$$

or

$$S_i = P(A_{x_i} - B_{x_i}R^{-1}B^TP) + (A_{x_i} - B_{x_i}R^{-1}B^TP)^TP + Q_{x_i} - PB(R^{-1})_{x_i}B^TP \tag{93}$$

where we define

$$A_{x_i} \equiv \frac{\partial A}{\partial x_i} \qquad B_{x_i} \equiv \frac{\partial B}{\partial x_i} \tag{94}$$

for each element x_i of the vector x. Notice that S_i is a *symmetric* matrix, but is **not** necessarily positive definite.

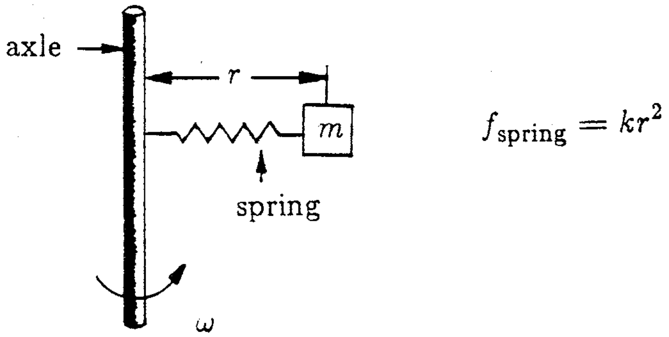


Figure 2. Angular rate control of a nonlinear plant.

Now substituting this new definition for S_i (94) into a parallel derivation (72)–(78), we obtain a similar sufficient condition for stability of the closed-loop system (90):

$$\sum_{i=1}^{n} S_i x_i > 0 \;\blacksquare \tag{95}$$

An example of this theory is given in the next section.

2.5. Example of PLQ Control for a Nonlinear Scalar Plant

In this example, figure 2, we show how to use PLQ to control the angular rate, ω, of a mass, m, connected to a rotating axle with a nonlinear spring which force, f_{spring}, is given by:

$$f_{\text{spring}} = kr^2 \,. \tag{96}$$

It is assumed that we have a point mass, m, that can only move along a radial line from the axle. We further **assume** that the *dynamics* of the *radial motion* can be *neglected* and the following algebraic relation can be used (centripetal force = spring force):

$$rm\omega^2 = kr^2 \,. \tag{97}$$

From (97) we calculate r as a function of the angular rate, ω:

$$r = \sqrt{\alpha}\,\omega^2 \,, \tag{98}$$

where the constant

$$\sqrt{\alpha} = \frac{m}{k} > 0 \,. \tag{99}$$

The moment of inertia, I, around the axle is given by:

$$I = mr^2 + mr_0^2\,, \tag{100}$$

where the first term is due to the mass m and the second term is due to the axle itself. Assuming linear viscous friction, $\beta\omega$, then the input torque, N, is given by:

$$N = I\dot{\omega} + \beta\omega\,. \tag{101}$$

Now by dividing (101) by I ($\neq 0$), using (100) and rearranging we obtain the following nonlinear plant equation:

$$\dot{\omega} = a(\omega)\omega + b(\omega)N\,, \tag{102}$$

where

$$a(\omega) = \frac{-\beta}{m(r_0^2 + \alpha\omega^4)}\,, \tag{103}$$

$$b(\omega) = \frac{1}{m(r_0^2 + \alpha\omega^4)}\,, \tag{104}$$

which is in the desired form (83).

Our choice of **pseudo cost function** is:

$$J = \mathfrak{E}\{\frac{1}{\rho}\omega^2 + N^2\}\,. \tag{105}$$

which corresponds to:

$$Q(\mathrm{x}) = 1/\rho\,, \quad R(\mathrm{x}) = 1\,, \tag{106}$$

functions of (86), and where we initially choose ρ as a constant.

The **control law** used is like (84) with an additional input, N_{ref},

$$N = -f(\omega)\omega + N_{\text{ref}}\,, \tag{107}$$

where the feedback gain, $f(\omega)$, is computed from the positive solution $p(\omega)$ of the ARE (110) by the equivalent of (85):

$$f(\omega) = b(\omega)p(\omega)\,. \tag{108}$$

Thus the closed loop system is given by:

$$\dot{\omega} = (a(\omega) - b(\omega)f(\omega))\omega + b(\omega)N_{\text{ref}}\,. \tag{109}$$

Let use compute the solution to the corresponding ARE (102), (105):

$$-p^2(\omega)b^2(\omega) + 2p(\omega)a(\omega) + \frac{1}{\rho} = 0\,, \tag{110}$$

from which the feedback gain, $f(\omega)$, is computed using (108) and the positive solution of (110):

$$f(\omega) = \frac{a(\omega)}{b(\omega)} + \frac{\sqrt{a^2(\omega) + \frac{1}{\rho}b^2(\omega)}}{b(\omega)}\,. \tag{111}$$

Thus from (109) and (111) we obtain the closed-loop system:

$$\dot{\omega} = -\sqrt{a^2(\omega) + \frac{1}{\rho}b^2(\omega)}\,\omega + b(\omega)N_{\text{ref}} \blacksquare \tag{112}$$

This closed-loop system (112) is stable by (89)

$$(2q + \frac{dq}{dx}x) + \frac{p^2b^2}{r^2}(2r + \frac{dr}{dx}x) + 2p\Big(\frac{da}{dx} - \frac{db}{dx}r^{-1}bp\Big)x > 0\,, \tag{89}$$

since $q = 1/\rho > 0$, $dq/dx = 0$, $r = 1 > 0$, $dr/dx = 0$, which leaves us to check only the last additive term of (89). Now since:

$$\Big(\frac{da}{d\omega} - \frac{db}{d\omega}r^{-1}bp\Big) = \frac{da}{d\omega}\frac{1}{a}a_{\text{CL}}\,, \tag{113}$$

where by (103), (104),

$$a_{\text{CL}} \triangleq a - br^{-1}bp = a\Big(1 + \frac{1}{\beta}r^{-1}bp\Big)\,, \tag{114}$$

and using

$$\frac{da}{d\omega}\omega = \frac{4\alpha\beta\omega^3}{m(r_o^2 + \alpha\omega^4)^2}\omega \geq 0 \qquad \text{for all } \omega\,, \tag{115}$$

in (113) with the facts that $a(\omega) < 0$ and that $a_{\text{CL}} < 0$ for all ω, we satisfy (89).

PLQ Control System Analysis

Let us first analyze the closed-loop system (112) for for the following two limit cases:

$$\rho \to \infty \;\Rightarrow\; \dot{\omega} = -|a(\omega)|\omega + b(\omega)N_{\text{ref}}\,, \tag{116}$$

$$\rho \to 0 \;\Rightarrow\; \dot{\omega} = -\frac{1}{\sqrt{\rho}}|b(\omega)|(\omega - N_{\text{ref}}) = \frac{-1}{\sqrt{\rho}m\Big(r_0^2 + \big(\frac{m}{k}\big)\omega^4\Big)}(\omega - N_{\text{ref}})\,, \tag{117}$$

where the last limit ($\rho \to 0$) corresponding to the case were the system state ω follows the reference input, N_{ref}, perfectly in the steady state.

PLQ Control Shaping

For small $\rho > 0$ the system follows the reference input but the dynamics still depend on the angular rate, ω. We can dispense of this dependence by using the following function for $\rho(\omega)$:

$$\sqrt{\rho} = \delta|b(\omega)|\,. \tag{118}$$

Then when approaching the limit $\delta \to 0$:

$$\delta \to 0 \quad \Rightarrow \quad \dot{\omega} = -\frac{1}{\delta}(\omega - N_{\text{ref}}) \;\blacksquare \tag{119}$$

which is also stable by condition (89) since:

$$\frac{d1/\rho}{d\omega}\omega \geq 0 \qquad \text{for all } \omega\,, \tag{120}$$

and by using (113)–(115).

3. Alternative Performance Criteria

This new *sign sensitive* cost includes a new term that depends on the relative sign of the vestibular outputs y^a and y^s of the airplane pilot and simulator pilot. The new term is in the form of a correlation function i.e. $\mathfrak{E}\{y^a y^s\}$ and thus also called a *correlation cost.* This *sign sensitive* cost leads to enhancement of the motion transitions which seems like a welcomed property. It should be noted that adding this term makes the actual solution much more sensitive to a change in the design parameter ρ (the relative weight of the control compared to the motion quality criteria). Furthermore the relation between the solution properties and the design parameters is not as clear.

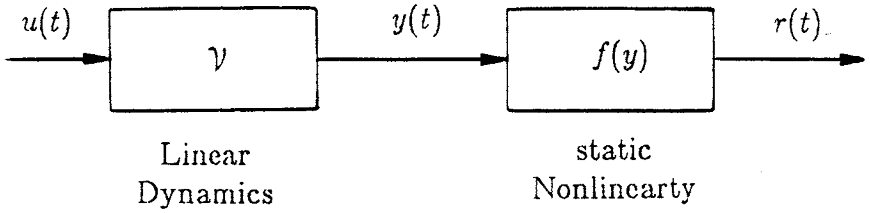


Figure 3. Including static sensor nonlinearity in the vestibular model.

3.1. Introduction

Hosman [Hosman79] formulated a simulator washout design problem as a parameter optimization problem for a given washout structure. His cost function J included the standard terms and also included a quadratic form of the vestibular error (similar to the formulation in Chapter II, Section II.1). It also included an extra term that added penalties for sign errors between the vestibular output of actual and simulator vestibular models, i.e. y^a and y^s. Specifically, if y^a and y^s have the same sign ($y^s y^a > 0$), then the penalty is zero, but if they have different signs ($y^s y^a < 0$), there is a given constant added to the cost function J. This is a non-quadratic term. It is used without any explanation or reference.

We offer three explanations for this added non-quadratic term in the cost J_e:

1. The effect of the limited dynamic range of the sensory cells in the vestibular organ, namely the effect of saturation.
2. The effect of the probability distributions in the afferent firing rates, namely that there are different firing rate probability distributions for different hair cells. These distributions range form Gaussian, where the standard deviation is independent of the means, at one extreme, to a Poisson distribution where the standard deviation is equal to the mean at the other extreme.
3. We used the vestibular error as the comparison operator to determine the quality of our simulation. Picking a correlation operation between two vestibular outputs y^a and y^s, rather than the vestibular error will also require a *sign sensitive* cost. The correlation operation is a common technique used in detection theory.

Next we look at these three explanations in detail.

3.1.1. Sensor Saturation

In order to take this effect into account, we will augment our vestibular model by a nonlinear output function called $f(y)$. This function will operate on the output of our previously used vestibular model's input y and its output will be called r which is our modified firing rate considering the limited dynamic range or saturation effect of the sensory cells, the hair cells. We use a modified comparison that uses the error between the outputs r of the actual and simulator pilots namely r^a and r^s rather then the outputs y^a and y^s. We define the new error e_r as

$$\mathbf{e}_r \overset{\Delta}{=} \mathbf{r}^a - \mathbf{r}^s\,. \tag{121}$$

Let us take a simple function $f_1(y)$ that will demonstrate the idea discussed. Let $f_1(y)$ be given by :

$$r = f_1(y) = \frac{y}{1+|y|}\,, \tag{122}$$

where $|y|$ in equation (122) is assumed to be in threshold units, so that the constant 1 in the denominator of (122) is meaningful. In our evaluation of r^a and r^s, we use (122), with appropriate superscripts, and from these we can write the firing rate error e_r as:

$$e_r = \frac{e - (y^a|y^s| - |y^a|y^s)}{(1+|y^a|)(1+|y^s|)}\,, \tag{123}$$

where we recall that $e = y^a - y^s$.

Let us analyze $|e_r|$ in the following three cases:

1. If $|y^a| \ll 1$ and $|y^s| \ll 1$ then:

$$|e_r| \approx \begin{cases} |e| & \text{for } y^a y^s \geq 0 \\ ||e| - 2|y^a y^s|| & \text{for } y^a y^s < 0 \end{cases}. \tag{124}$$

2. If $|y^a| \gg 1$ and $|y^s| \ll 1$ then:

$$|e_r| \approx 1\,. \tag{125}$$

3. If $|y^a| \gg 1$ and $|y^s| > 1$ then:

$$|e_r| \approx \begin{cases} \dfrac{|e|}{(1+|y^a|)(1+|y^s|)} & \text{for } y^a y^s \geq 0 \\ 2\dfrac{|y^s|}{1+|y^s|} & \text{for } y^a y^s < 0 \end{cases}. \tag{126}$$

From this analysis it is clear that for small y^a and y^s (less than 1), the error e represents the error e_r quite well but for $|y^a| \gg 1$ and $|y^s| \gg 1$, which is the more interesting case, the error e_r can be approximated by zero for $y^a y^s \geq 0$ and 2 for $y^a y^s < 0$, which is the function used by Hosman.

The property of the function $f_1(y)$, used in (122), is that it is an increasing function that has a decreasing slope with an increase in the value y for $y > 0$. Namely for

$$f(y) = -f(-y) \quad \text{and,} \quad \frac{df}{dy}(y_1) \geq \frac{df}{dy}(y_2) \geq 0 \quad \text{for all} \quad y_2 \geq y_1 \geq 0\,, \qquad (127)$$

In the next subsection, we will give another example $f_2(y)$ that would have the same kinds of properties. Furthermore, we also present a general theory that shows that the required properties of the function f(y) in (127) would give us this *sign sensitivity* property.

3.1.2. Firing Rate Statistics

As noted in [Wilson79] the coefficient of variation (CV), which is the ratio between the standard deviation and the mean, varies as a function of the diameter of the afferent fibers in the vestibular organ, that is related to the connections of the two types of hair cells. For a Poisson distribution CV=1 while for a Gaussian distribution CV=0. In the vestibular organ we find a continuum of values of CV between 0 and 1. The Gaussian distribution corresponds to Type II hair cells which are referred to as regular units while Type I hair cells have a Poisson distribution and are referred to as irregular units. It seems that Type I hair cells are more sensitive then Type II ones. We assume that increasing the input corresponds to increasing the mean firing rate of the hair cells. Thus for Type II hair cells the "noise" in the firing rate is independent of the input level and the signal to noise ratio improves with an increase of the input. In contrast for Type I hair cells the signal to noise ratio is constant.

On the basis of this knowledge, it could be said that our previous discussion in chapter II assumes only a Gaussian distribution. Furthermore, psychophysical experiments show that the threshold for detection depends on the value of the signal applied, i.e. the input motion, and that the dependence is a power law, as suggested by Stevens, or logarithm law as suggested by Webber-Fechner which leads to the same resulting property. We account for this change in noise level as a function of the mean by normalizing our output to a threshold that depends on the mean. By that we obtain a function that is similar to the function $f_1(y)$, we discussed before in Subsection 3.1.1. Let us choose an example for such a function $f_2(y)$ which corresponds to the general findings of Stevens, i.e. has the property,

$$\frac{df_2(y)}{dy} = \begin{cases} \dfrac{1}{1-y} & \text{for } y < 0 \\ \dfrac{1}{1+y} & \text{for } y \geq 0 \end{cases} \tag{128}$$

Using equation (128) we solve and find $f_2(y)$,

$$r = f_2(y) = \begin{cases} -\ln(1-y) & \text{for } y < 0 \\ \ln(1+y) & \text{for } y > 0 \end{cases} \tag{129}$$

The definition of the derivative of $f_2(y)$ in (128) accounts for the decrease in sensitivity of r as a function of y for an increases or a decreases of y from zero.

Now we proceed to calculate e_r defined in (121) using (129):

$$|e_r| = \begin{cases} |\ln(1+|y^a|)(1+|y^s|)| & \text{for } y^a y^s < 0 \\ \left|\ln \dfrac{1+|y^a|}{1+|y^s|}\right| & \text{for } y^a y^s \geq 0 \end{cases} \tag{130}$$

Analyzing (130) in the three cases defined before we obtain results that have a similar characteristics to (124)–(126) obtained for $f_1(y)$, although much more complex.

Next we introduce a theorem that proves that for our class of functions $f(y)$ there exist a relation between the magnitude of e_r and the relative sign of y^a and y^s.

Theorem

Given a function $f(y)$ which has the following properties:

(i) Differentiable on the open interval $(-\infty, \infty)$.

(ii) Odd, i.e.

$$f(y) = -f(-y). \tag{131}$$

(iii) $\frac{df}{dy}(y)$ is a decreasing function for all $y > 0$ i.e.

$$\frac{df}{dy}(y_1) \geq \frac{df}{dy}(y_2) \geq 0 \qquad \text{for all} \quad y_2 > y_1 \geq 0\,. \tag{132}$$

Also, given two finite points $y^a < \infty$ and $y^s < \infty$, that have a fixed finite distance between them e, then for the following function:

$$e_r(y^a, y^s) \triangleq f(y^a) - f(y^s)\,, \tag{133}$$

we have

$$|e_r(y^a y^s < 0)| \geq |e_r(y^a y^s \geq 0)| \geq 0 \qquad \text{for all} \quad |y^a - y^s| = e = \text{constant} \blacksquare \tag{134}$$

Proof

Let us define four points $y^a_<$, $y^s_<$, $y^a_\geq$, and $y^s_\geq$ such that:

$$y^a_< y^s_< < 0 \quad \text{and} \quad y^a_\geq y^s_\geq \geq 0\,, \tag{135}$$

and such that

$$y^a_\geq - y^s_\geq = e = y^a_< - y^s_<\,. \tag{136}$$

For our convince we assume that (still general due to the anti symmetry of $f(y)$ (131)),

$$y^a_\geq > y^s_\geq \geq 0 \tag{137}$$

and thus,

$$e > 0\,, \quad y^a_< > y^s_<\,, \tag{138}$$

and further using (136) we have,

$$\left|y^a_\geq\right| > \left|y^a_<\right|, \quad \left|y^a_\geq\right| > \left|y^s_<\right|. \tag{139}$$

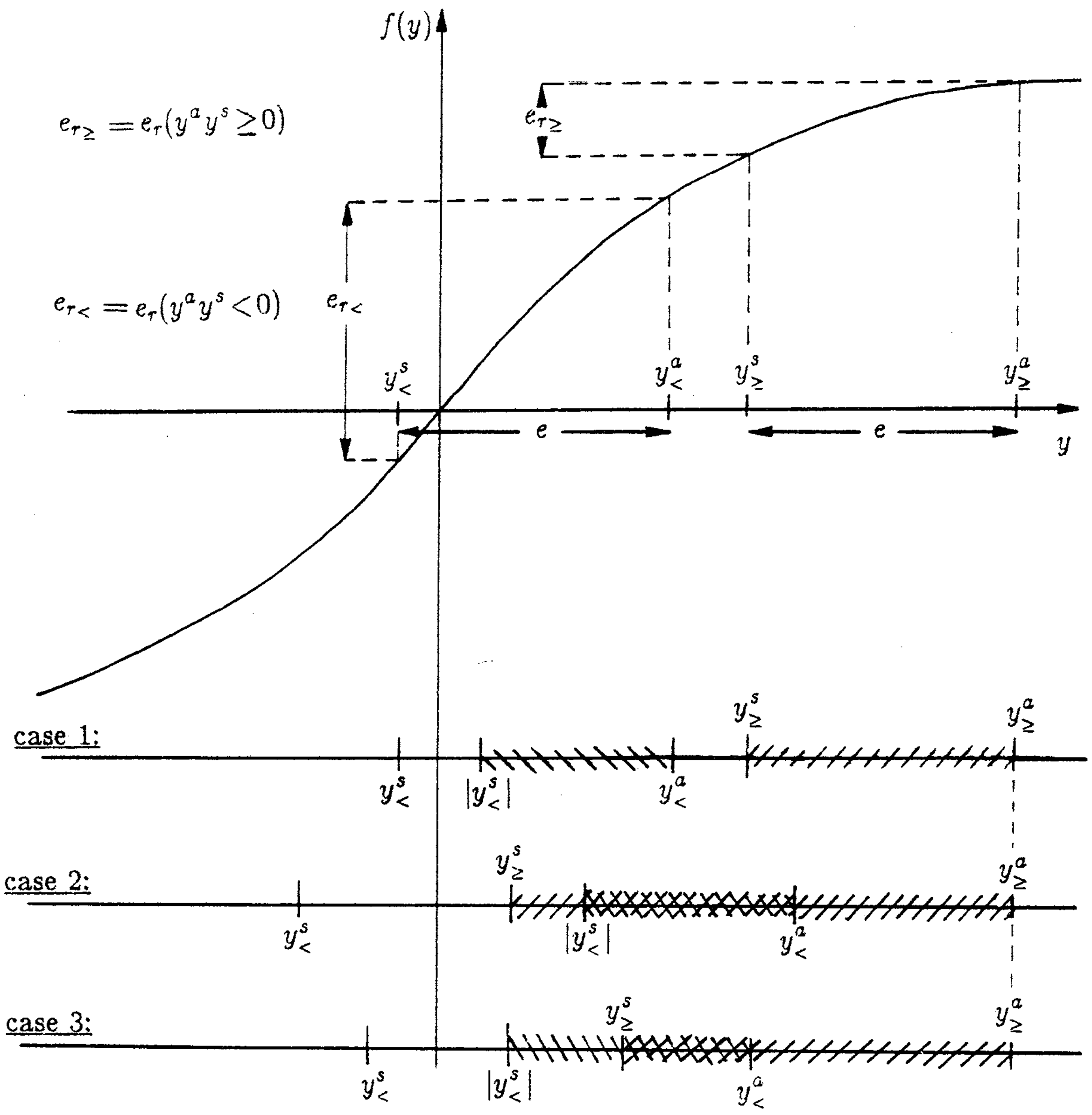


Figure 4. Example of the relative locations of the four points $y^a_<$, $y^s_<$, $y^a_\geq$, $y^s_\geq$ in the three cases, using $y^a_\geq > y^s_\geq \geq 0$ (137).

Let us distinguish between three cases according to the absolute values of the four points using relations (138)–(139) (Figure 4):

$$\underline{\text{case 1:}} \quad |y^s_\geq| \geq |y^a_<| \quad \text{and} \quad |y^s_\geq| \geq |y^s_<| . \tag{140}$$

$$\underline{\text{case 2:}} \quad |y^a_<| \geq |y^s_\geq| \quad \text{and} \quad |y^s_<| \geq |y^s_\geq| . \tag{141}$$

$$\underline{\text{case 3:}} \quad |y^s_<| > |y^s_\geq| > |y^a_<| \quad \text{or} \quad |y^a_<| > |y^s_\geq| > |y^s_<| . \tag{142}$$

The classification into these three cases is done by fixing the value of one point, say $y^a_{\geq}$, and checking all six possible arrangements of the points $|y^s_{\geq}|$, $|y^a_{<}|$, $|y^s_{<}|$, on the positive real axis. Let us first prove case 1 and use it to prove cases 2 and 3.

Proof of case 1

Let us define $e_{r<}$ as $e_r(y^a_{<}\,,\,y^s_{<})$ (recall $y^a_{<}y^s_{<} < 0$) and define $e_{r\geq}$ as $e_r(y^a_{\geq}\,,\,y^s_{\geq})$ (recall $y^a_{\geq}y^s_{\geq} \geq 0$). Now by the Lagrange theorem of calculus (an extension of Rolls theorem) and properties (i), (ii) of $f(y)$, there exists an intermediate point θ, $y_2 > \theta > y_1$ such that:

$$e_r = \frac{df}{dy}(\theta)\,(y_2 - y_1)\,. \tag{143}$$

Now according to (137) and (138) let us define the points $\theta_<$ and $\theta_\geq$ such that,

$$y^a_{<} > \theta_< > y^s_{<} \quad \text{and} \quad y^a_{\geq} > \theta_\geq > y^s_{\geq}\,. \tag{144}$$

Let us evaluate the following using (143), (138) and the fact that $\frac{df}{dy} > 0$ for all y, (131)-(132),

$$\begin{aligned}
|e_{r<}| - |e_{r\geq}| &= \left|\frac{df}{dy}(\theta_<)\,e\right| - \left|\frac{df}{dy}(\theta_\geq)\,e\right| = \\
&= \left(\frac{df}{dy}(\theta_<) - \frac{df}{dy}(\theta_\geq)\right)e = \\
&\geq \left(\min\left(\frac{df}{dy}(\theta_<)\right) - \max\left(\frac{df}{dy}(\theta_\geq)\right)\right)e = \\
&= \left(\frac{df}{dy}\Big(\max(|y^a_{<}|\,,\,|y^s_{<}|)\Big) - \frac{df}{dy}\Big(\min(|y^a_{\geq}|\,,|y^s_{\geq}|)\Big)\right)e
\end{aligned} \tag{145}$$

Now using property (132), the definition of case 1 (140), (137) and (139), we obtain from (145) that:

$$|e_{r<}| - |e_{r\geq}| \geq 0\,, \tag{146}$$

which proves case 1. QED

Proof of Cases 2 and 3

Cases 2 and 3 are proven by subtracting off the common regions of $f(y)$ using the fact that $f(y)$ is an odd function (131). Thus we obtain new pairs of points with a new smaller difference, e. These new points obeys the conditions of case 1, proven before. QED

3.1.3. Correlation Cost

Another approach is to replace/augment our error $e \overset{\Delta}{=} y^a - y^s$, comparison operation by the *correlation* operation $\mathfrak{E}\{y^a y^s\}$. Using *correlation* the relative sign of the y^a and y^s is also essential. Furthermore, the *correlation* operation suggested here is the optimal method to detect the "known" *reference signal*, y^a, in additive white Gaussian noise. The source of the *reference signal* used by the simulator pilot's brain is from the other sensors and the simulator pilot's expectation of what is about to happen. Thus we need to find a washout system such that we have:

$$\max(\mathfrak{E}\{y^a y^s\}) \quad \text{or} \quad \min(-\mathfrak{E}\{y^a y^s\}) . \tag{147}$$

In order to make this correlation criteria meaningful it is necessary to add the constraint that the energy in y^s is bound i.e.

$$\mathfrak{E}\{\lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{T/2} \mathbf{y}^{s^T}(t)\mathbf{y}^s(t)\,dt\} \le \upsilon , \tag{148}$$

where υ is a given constant. We next construct a new cost of the form (147) with the constraint (148) added by using lagrange multipliers.

3.2. Problem Formulation

From the introduction it is clear that it is desirable to include in the cost function, J, a sensitivity to the sign of each $y^s y^a$. To capture this idea we add to our previous J (III.7) a term of the form $-\mathbf{y}^{s^T}\mathbf{y}^a$. Let us define:

$$\mathbf{y}^{s+a} = \mathbf{y}^s + \mathbf{y}^a , \tag{149}$$

and use (149) to construct the following "modified" cost function:

$$J \overset{\Delta}{=} \mathfrak{E}\{\int_{-\infty}^{\infty} \mathbf{e}^T(t)Q_-\mathbf{e}(t) - \mathbf{y}^{s+a^T}(t)Q_+\mathbf{y}^{s+a}(t) + \rho\mathbf{u}^{\ell^T}(t)R\mathbf{u}^{\ell}(t)\,dt\} , \tag{150}$$

where

$$Q_- \ge 0 , \quad Q_+ \ge 0 , \quad D^{\ell^T} R D^{\ell} > 0 , \quad \rho > 0 . \tag{151}$$

Note that if $Q_- = Q_+ = Q$ then:

$$\mathbf{e}^T Q \mathbf{e} - \mathbf{y}^{s+a^T} Q \mathbf{y}^{s+a} = -4\mathbf{y}^{s^T} Q \mathbf{y}^a\,, \tag{152}$$

which is the cost term we were after. Note that this cost term tends to make y^s as large as possible independent of the value of y^a, this point is further discussed (the case of $q = 1$ in (156)) in conjunction with the deterministic and the stochastic solutions.

Next the cost function (150) is formulated as the usual quadratic cost function with the generalization that R_1 in (III.59) is not necessarily positive semidefinite. This generalized cost also arises when there are conflicting objective and was studied by Willems [Willems71], Jonckheere and Silverman [Jonckheere78]. Let us define the vector $\mathbf{y}^{sa}$ as:

$$\mathbf{y}^{sa}(t) = \begin{pmatrix} \mathbf{y}^s(t) \\ \mathbf{y}^a(t) \end{pmatrix}. \tag{153}$$

Thus (149) can be written as:

$$J = \mathfrak{E}\{\int\limits_{-\infty}^{\infty} \mathbf{y}^{sa^T}(t) Q_{sa} \mathbf{y}^{sa}(t) + \rho \mathbf{u}^{\ell^T}(t) R \mathbf{u}^{\ell}(t)\, dt\}\,, \tag{154}$$

where

$$Q_{sa} = \begin{pmatrix} (Q_- - Q_+) & -(Q_- + Q_+) \\ -(Q_- + Q_+) & (Q_- - Q_+) \end{pmatrix}. \tag{155}$$

For our problem it is sufficiently general to restrict the choice of Q_- and Q_+ as follows:

$$Q_- = Q\,, \quad Q_+ = qQ\,, \qquad \text{where} \qquad Q \geq 0\,, \quad 0 \leq q \leq 1\,. \tag{156}$$

For $q = 0$ we return to our previous J (III.7) while in the other extreme, $q = 1$, we obtain the case given by (152) which is "over use" of our simplified model to account for sensitivity to the sign of $y^s y^a$. The choice of the best value for the parameter q was based on simulation results. Further analysis and experimental validation are recommended as further research.

As before the system equation are give by (III.1)–(III.5):

$$\mathcal{S}^s: \quad \begin{aligned} \dot{\mathrm{x}}^s(t) &= A^s\mathrm{x}^s(t) + B^s\mathbf{u}^s(t) && \text{(III.1)}\\ \mathrm{y}^s(t) &= C^s\mathrm{x}^s(t) + D^s\mathbf{u}^s(t) && \text{(III.2)}\\ \mathbf{u}^\ell(t) &= C^\ell\mathrm{x}^s(t) + D^\ell\mathbf{u}^s(t) && \text{(III.3)} \end{aligned}$$

$$\mathcal{S}^a: \quad \begin{aligned} \dot{\mathrm{x}}^a(t) &= A^a\mathrm{x}^a(t) + B^a\mathbf{u}^a(t) && \text{(III.4)}\\ \mathrm{y}^a(t) &= C^a\mathrm{x}^a(t) + D^a\mathbf{u}^a(t) && \text{(III.5)} \end{aligned}$$

Since our new cost (154) is also quadratic then we can separate the optimization problem into a deterministic and stochastic subproblem as before, section III.2. It should be noted that initially it is not clear that this optimization problem has a solution. We show that due to the special structure of our problem the solution exists and is unique.

3.3. Solution

This section parallels sections III.3–III.4 and thus is very brief and does not include full derivations.

3.3.1. Deterministic Solution

In this solution we us the cost form (150) which just adds a term in the derivation in section III.3. The new optimal solution for $\overline{W}(j\omega)$ is:

$$\begin{aligned} \overline{W}(j\omega) =& \Big(S^{s^H}(j\omega)(Q_- - Q_+)S^s(j\omega) + \rho S^{\ell^H}(j\omega)RS^\ell(j\omega)\Big)^{-1} \\ & \Big(S^{s^H}(j\omega)(Q_- + Q_+)S^a(j\omega)\Big). \end{aligned} \tag{157}$$

Comparing this new result with the "old" one (III.49) we see that Q is replaced by the two combinations:

$$Q_{mp} \triangleq Q_- - Q_+ = (1-q)Q\,, \qquad Q_{pm} \triangleq Q_- + Q_+ = (1+q)Q\,, \tag{158}$$

where we used (156) to get Q_{mp} and Q_{pm} in terms of the scalar parameter q. Now note that $q = 1$ implies that $Q_{mp} = 0$ which is a special singular case to be aware of. When $q = 1$, $\overline{W}$ increases with no bound as $\rho \to 0$ (as reducing the cost on the motion). This limit, $q = 1$ is also pointed out in the stochastic solution and further discussed latter on.

3.3.2. Stochastic Solution

The equivalent of the augmented stochastic system ((III.53)–(III.55)) for this "modified" problem is:

$$\tilde{S}: \quad \begin{aligned} \dot{\tilde{\mathrm{x}}}(t) &= \tilde{A}(t)\tilde{\mathrm{x}}(t) + \tilde{B}\tilde{\mathrm{u}}^s(t) + \tilde{H}(t)\mathrm{n}(t) && (159)\\ \tilde{\mathrm{y}}(t) &= \tilde{C}(t)\tilde{\mathrm{x}}(t) + \tilde{D}\tilde{\mathrm{u}}^s(t) && (160) \end{aligned}$$

where

$$\tilde{\mathrm{x}}(t) = \begin{pmatrix} \tilde{\mathrm{x}}^s(t) \\ \tilde{\mathrm{x}}^a(t) \\ \tilde{\mathrm{x}}^n(t) \end{pmatrix}, \quad \tilde{\mathrm{y}}(t) = \begin{pmatrix} \tilde{\mathrm{y}}^s(t) \\ \tilde{\mathrm{y}}^a(t) \\ \tilde{\mathrm{u}}^\ell(t) \end{pmatrix}, \tag{161}$$

and

$$\tilde{A}(t) = \begin{pmatrix} A^s & 0 & 0 \\ 0 & A^a & B^aC^n(t) \\ 0 & 0 & A^n(t) \end{pmatrix}, \quad \tilde{B} = \begin{pmatrix} B^s \\ 0 \\ 0 \end{pmatrix}, \quad \tilde{H}(t) = \begin{pmatrix} 0 \\ 0 \\ B^n(t) \end{pmatrix}, \tag{162}$$

$$\tilde{C}(t) = \begin{pmatrix} C^s & 0 & 0 \\ 0 & C^a & D^aC^n(t) \\ C^\ell & 0 & 0 \end{pmatrix}, \quad \tilde{D} = \begin{pmatrix} D^s \\ 0 \\ D^\ell \end{pmatrix}. \tag{163}$$

The stochastic optimization criterion $\tilde{J}$ derived from (154) can also be written as:

$$\tilde{J} = \mathfrak{E}\{\tilde{\mathrm{x}}^T(t)R_1(t)\tilde{\mathrm{x}}(t) + 2\tilde{\mathrm{x}}^T(t)R_{12}(t)\tilde{\mathrm{u}}^s(t) + \tilde{\mathrm{u}}^{s^T}(t)R_2\tilde{\mathrm{u}}^s(t)\}, \tag{164}$$

where

$$R_1(t) = \tilde{C}^T(t)\tilde{Q}\tilde{C}(t), \tag{165}$$

$$R_{12}(t) = \tilde{C}^T(t)\tilde{Q}\tilde{D}, \tag{166}$$

$$R_2 = \tilde{D}^T\tilde{Q}\tilde{D}, \tag{167}$$

and where

$$\tilde{Q} = \begin{pmatrix} (1-q)Q & -(1+q)Q & 0 \\ -(1+q)Q & (1-q)Q & 0 \\ 0 & 0 & \rho R \end{pmatrix}. \tag{168}$$

Expansion of (165)–(167) in terms of the given sub-matrices is given in appendix B. Note that $R_2 > 0$ since $\rho > 0$, $D^{\ell^T}RD^\ell > 0$, $Q \geq 0$ and $0 \leq q \leq 1$. Remember that R_1 is not necessarily positive semidefinite.

The problem of minimizing (164) subject to the system constraint (159)-(160), is a stochastic state feedback optimization problem, the steady state solution in the time invariant case, if exists is [Willems71]:

$$\tilde{\mathbf{u}}^s(t) = -F\tilde{\mathbf{x}}(t)\,, \tag{169}$$

where F is given by

$$F = R_2^{-1}(\tilde{B}^T P + R_{12}^T)\,, \tag{170}$$

and P is the real unique positive semidefinite solution of the ARE:

$$\begin{aligned} 0 = & -P\tilde{B}R_2^{-1}\tilde{B}^T P + P(\tilde{A} - \tilde{B}R_2^{-1}R_{12}^T) + (\tilde{A} - \tilde{B}R_2^{-1}R_{12}^T)^T P \\ & + R_1 - R_{12}R_2^{-1}R_{12}^T\,. \end{aligned} \tag{171}$$

The condition for which such a solution P exists for (171) such that all the controllable eigenvalues of the closed loop system, (eigenvalue($\tilde{A} - \tilde{B}F$) ≤ 0), is if and only if [Willems71]:

$$\begin{aligned} H(-j\omega, j\omega) = & R_2 + R_{12}(j\omega I - \tilde{A})^{-1}\tilde{B} + \tilde{B}^T(-j\omega I - \tilde{A}^T)^{-1}R_{12}^T \\ & + \tilde{B}^T(-j\omega I - \tilde{A}^T)^{-1}R_1(j\omega I - \tilde{A})^{-1}\tilde{B} \geq 0\,, \end{aligned} \tag{172}$$

for all real ω. This shows that there are conditions under which our problem has a unique solution—although not possible to test for due to our given dimension of $\tilde{A}$. Furthermore if we find a unique solution for P such that the closed loop eigenvalues are stable, (< 0), then from Willems theorem it is the solution to our optimization problem. Next we use the derivation in appendix B and the stochastic solution properties derived in chapter III to show that this is indeed the case.

In appendix B we expanded the ARE (171) into six block matrix equations similar to what was done in chapter III. Comparing the two results we see the following:

(i) Block 1,1 is the same ARE as in chapter III with the exception that Q is replaced by $Q_{mp} = (1 - q)Q$.

(ii) Blocks 1,2, 1,3 are the same sylvester linear equations as in chapter III with the exception that Q is replaced by either $Q_{mp} = (1 - q)q$ or by $Q_{pm} = (1 + q)Q$.

Let us first discuss the solution for P^s given by the ARE in block 1,1. From observation (i) above and since $Q_{mp} = (1-q)Q \geq 0$ we conclude that a unique solution $P^s \geq 0$ exists subject to the the usual conditions of existence state in chapter III where we use Q_{mp} instead of Q. Thus from the solution properties show in subsection III.5.1 then the closed-loop optimal washout system (OWS) is stable, i.e. the closed-loop eigenvalues ≤ 0.

The rest of the equation blocks of the ARE are linear sylvester equations and thus if they have a solution it is unique. Following the same arguments as before then the conditions of existence are the same as in chapter III where we use Q_{mp} instead of Q. Only subexpression (E42) in appendix B which depends only on Q_{mp} is important for the solution existence since (E44) and (E45) are the constants in the equations in blocks 1,2 and 1,3 respectively. Thus we conclude that unique stable solution exists to our modified problem ARE subject to the usual solution existence conditions stated in chapter III where we us Q_{mp} instead of Q. Now from Willems theorem the solution we obtained is indeed the solution to our modified optimization problem (159)-(168). Note that from this discussion it follows that all the solution properties discussed in section III.5 carry on to this modified problem. In the next subsection we discuss the special singular case where $q = 1$.

3.4. Special Properties of the Solution

Let us analyze the limit case $q = 1$ (or $Q_{mp} = 0$). In this case the equation for block 1,1 is given by (like (III.92)):

$$-\frac{1}{\rho}P^s B^s (D^{\ell^T} R D^{\ell})^{-1} B^{s^T} P^s + P^s \mathcal{A}^s + \mathcal{A}^{s^T} P^s + \mathcal{Q} = 0\,, \tag{173}$$

and its coefficients $\mathcal{A}^s$ and $\mathcal{Q}$ of (III.92), (III.95), (III.96) ((E42) and (E43) of appendix B) simplify to:

$$\mathcal{A}^s = A^s - B^s (D^{\ell^T} R D^{\ell})^{-1} D^{\ell^T} R C^{\ell}\,, \tag{174}$$

$$\mathcal{Q} = \rho C^{\ell^T}\Big(I - R D^{\ell} (D^{\ell^T} R D^{\ell})^{-1} D^{\ell^T}\Big) R C^{\ell}\,. \tag{175}$$

In the special case where the matrix $C^\ell = 0$ then (173)-(175) simplify to ($\mathcal{A}^s = A^s$, $\mathcal{Q} = 0$):

$$-\frac{1}{\rho}P^sB^s(D^{\ell^T}RD^\ell)^{-1}B^{s^T}P^s + P^sA^s + A^{s^T}P^s = 0\,, \tag{176}$$

i.e. there are only limitations on the control variables of the motion-base. In the case that A^s is asymptotically stable then the only solution $P^s \geq 0$ is $P^s = 0$ which corresponds to zero feedback gains, $F^s = 0$. Thus the only "operations" the OWS does is to shape the input $\mathbf{u}^a$ by gains F^n and add a filtered version of them through a vestibular model using gains F^a. This OWS form is a slightly more sophisticated version of the *motion scaling* method used in some flight simulator designs. This is also the method used in the original design of the *Link* GAT-1 flight simulator as discussed chapter VII (the GAT-1 solution there **does not** fit the formulation here!). Now we can understand why this simple design method works well in some cases, like in a hovering simulation on a large six degree-of-freedom motion-base such as the VMS at NASA AMES.

In general when $\rho \to 0$ then $\mathcal{Q} \to 0$ (175) and thus (173), (170):

$$\rho \to 0 \qquad \Rightarrow \qquad F^s \neq F^s(\rho)\,. \tag{177}$$

While in the special case before (stable plant and $C^\ell = 0$) we obtained a special case of (177) i.e. $F^s = 0$. Intuitively what is happening is that as $\rho \to 0$ then we no longer have restrictions on the motion and thus the feedback is only needed in order to stabilize the plant (it necessary). "Once" the "plant is stable" it behaves like our special case before of a stable plant.

From the ARE blocks 1,2 and 1,3 (appendix B) we see that for small ρ then:

$$P^a \neq P^a(\rho) \qquad \text{and} \qquad P^n \neq P^n(\rho)\,, \tag{178}$$

and thus:

$$F^a \propto \frac{1}{\rho} \qquad \text{and} \qquad F^n \propto \frac{1}{\rho}\,. \tag{179}$$

Combined with result (177) we see that as $\rho \to 0$ then the gains increase without bound. This solution behavior is similar to that obtain for the deterministic solution.

This result is also consistent with the comment we already made about minimizing a cost function of the form $-\mathfrak{E}\{y^s y^a\}$ which likes y^s to be as large as possible (note that y^a is like a constant, we cannot control it).

The solution to this weakness of this formulation is to add a cost term $\mathrm{y}^{s^T} Q_{y^s} \mathrm{y}^s$ that would put a bound on a quadratic norm of y^s. Using this type of term in the cost (154) gives the following modified cost matrix Q_{sa}:

$$J = \mathfrak{E}\{\int_{-\infty}^{\infty} \mathrm{y}^{sa^T}(t) Q_{sa} \mathrm{y}^{sa}(t) + \rho \mathbf{u}^{\ell^T}(t) R \mathbf{u}^{\ell}(t)\, dt\}\,, \tag{154}$$

where

$$Q_{sa} = \begin{pmatrix} Q_{y^s} + (Q_- - Q_+) & -(Q_- + Q_+) \\ -(Q_- + Q_+) & (Q_- - Q_+) \end{pmatrix}. \tag{180}$$

Analysis of the resulting block matrices in this case (appendix C) reveals that what this does is the same as limiting the value of q to less then 1.

References

[Bass66] Bass, R.W. and Webber, R.F., "Optimal Nonlinear Feedback Control Derived from Quartic and Higher-Order Performance Criteria," *IEEE Transactions on Automatic Control*, **AC-11**, no. 3, pp. 448–454, July, 1966.

[Gelb74] Gelb, A., editor, *Applied Optimal Estimation*, the M.I.T. Press, pp. 203–207, 1974.

[Hosman79] Hosman, R.J.A.W., van der Vaart, J.C. and van de Moesdijk, G.A.J., "Optimalization and Evaluation of Linear Motion Filters," Annual Conference on Manual Control, March, 1979.

[Jazwinski70] Jazwinski, A.H., *Stochastic Processes and Filtering Theory*, pp. 272–278, pp. 339–340, Academic Press, 1970.

[Jonckheere78] Jonckheere, E.A. and Silverman, L.M., "Spectral Theory of Linear-Quadratic Optimal Control Problem: Discrete-Time Single-Input Case," *IEEE Transactions on Circuits and Systems*, **CAS-25**, no. 9, pp. 810–825, Sep., 1978.

[Kosut79] Kosut, R.L., "Nonlinear Optimal Cue-Shaping Filters for Motion Base Simulators," *Journal Guidance and Control*, **2**, pp. 486–490, Dec., 1979.

[Sandor77] Sandor, J. and Williamson, D., "Design of Nonlinear Regulators for Linear Plants," *IEEE Transactions on Automatic Control*, **AC-22**, pp. 47-50, Feb., 1977.

[Willems71] Willems, J.C., "Least Squares Stationary Optimal Control and the Algebraic Riccati Equation," *IEEE Transactions on Automatic Control*, **AC-16**, no. 6, pp. 621–634, Dec., 1971.

[Wilson79] Wilson, V.J. and Melvill Jones, G., *Mammalian Vestibular Physiology*, Pleum Press, pp. 32–34, 1979.

```
      Appendix IV.A: Derivation of the "Best" FX, FU Parameters for a Linear
      -------------  Approximation of F(X,U)-F(E{X},E{U}).
                     (edited Macsyma output)

(D3)                   Wednesday, Jul 28, 1982  1:53pm

(C4) DECLARE(E,ADDITIVE)$
(C5) J:E((FFB-FX*X-FU*U)^2);
                                                 2
(D5)                      E((- FX X - FU U + FFB) )

(C6) JE:EV(EXPAND(%));

          2  2                                  2  2                   2
(D6)  E(FX  X  + 2 FU FX U X - 2 FFB FX X + FU  U  - 2 FFB FU U + FFB  )

(C7) JX:DIFF(JE,FX);     /* for min need JX=0 */

      d       2  2                                  2  2                   2
(D7) --- (E(FX  X  + 2 FU FX U X - 2 FFB FX X + FU  U  - 2 FFB FU U + FFB  ))
     dFX

(C8) JU:DIFF(JE,FU);     /* for min need also JU=0 */

      d       2  2                                  2  2                   2
(D8) --- (E(FX  X  + 2 FU FX U X - 2 FFB FX X + FU  U  - 2 FFB FU U + FFB  ))
     dFU

(C9) JX:EV(SUBST(FX^2*E(X^2),E(FX^2*X^2),JX),INFEVAL,NOUNS)$
(C10) JX:EV(SUBST(2*FU*FX*E(U*X),E(2*FU*FX*U*X),JX),INFEVAL,NOUNS)$
(C11) JX:EV(SUBST(-2*FX*E(FFB*X),E(-2*FFB*FX*X),JX),INFEVAL,NOUNS);

                            2
(D11)               2 FX E(X ) + 2 FU E(U X) - 2 E(FFB X)

(C12) JU:EV(SUBST(FU*FX*2*E(X*U),E(X*U*2*FU*FX),JU),INFEVAL,NOUNS)$
(C13) JU:EV(SUBST(FU^2*E(U^2),E(FU^2*U^2),JU),INFEVAL,NOUNS)$
(C14) JU:EV(SUBST(-2*FU*E(FFB*U),E(-2*FFB*FU*U),JU),INFEVAL,NOUNS);

                                          2
(D14)               2 FX E(U X) + 2 FU E(U ) - 2 E(FFB U)

(C15) FXFU:SOLVE([JX,JU],[FX,FU])$
(C16) FX:PART(FXFU,1,1);
                                               2
                          E(FFB U) E(U X) - E(U ) E(FFB X)
(D16)              FX = - --------------------------------
                                  2      2     2
                               E(U ) E(X ) - E (U X)

(C17) FU:PART(FXFU,1,2);
                                     2
                         E(FFB U) E(X ) - E(FFB X) E(U X)
(D17)               FU = --------------------------------
                                 2      2     2
                              E(U ) E(X ) - E (U X)
```

Appendix IV.B: Derivation of Block Equations for a "Sign Sensitive" Cost (Macsyma output edited a bit).

```
(D4)                Tuesday, Aug 24, 1982  11:13pm
(C5)  ALIAS(T,TRANSPOSE)$
(C6)  BOTHCASES:TRUE$
(C7)  MATRIX_ELEMENT_MULT:"."$
(C8)  MATRIX_ELEMENT_TRANSPOSE:NONSCALARS$
(C9)  DOTSCRULES:TRUE$
(C10) DECLARE([q,rho],SCALAR)$
(C11) DECLARE([Aa.Ba,Ca,Da,As.Bs.Cs,Ds.An.Bn,Cn,A1,B1,C1.D1],NONSCALAR)$
(C12) DECLARE([Paa,Pa.Pan.Ps.Pn.Pnn],NONSCALAR)$
(C13) /* DECLARE([Q,R,R2])$ */
/* not used so that T(Q)=Q, T(R)=R ... (all are symatric matrices) */
DECLARE(T.ADDITIVE)$
(C14) PR(MVAR):=SUBST('A[s],As.SUBST('B[s],Bs,SUBST('C[s],Cs.SUBST('D[s],Ds,
SUBST('A[a].Aa,SUBST('B[a],Ba.SUBST('C[a].Ca,SUBST('D[a].Da,
SUBST('A[1],A1.SUBST('B[1].B1.SUBST('C[1].C1,SUBST('D[1].D1,SUBST('R[2],R2,
SUBST('A[n].An.SUBST('C[n],Cn.SUBST('P[a,a],Paa.SUBST('P[a],Pa,
SUBST('P[a,n],Pan,SUBST('P[s].Ps.SUBST('P[n],Pn,SUBST('P[n,n],Pnn,
SUBST(Aa^T,T(Aa).SUBST(Ba^T,T(Ba).SUBST(Ca^T,T(Ca).SUBST(Da^T,T(Da),
SUBST(As^T,T(As),SUBST(Bs^T,T(Bs).SUBST(Cs^T,T(Cs).SUBST(Ds^T,T(Ds),
SUBST(An^T.T(An),SUBST(Cn^T.T(Cn).SUBST(C1^T,T(C1).SUBST(D1^T,T(D1),
SUBST(Pa^T,T(Pa).SUBST(Pan^T,T(Pan),SUBST(Pn^T,T(Pn),
MVAR))))))))))))))))))))))))))))))))))))$
(C15) TRANSLATE(PR);
(D15)                             [PR]
(C16) A%:MATRIX([As,0,0],[0,Aa,Ba.Cn],[0,0,An])$
(C17) B%:MATRIX([Bs],[0],[0])$
(C18) C%:MATRIX([Cs,0,0],[0,Ca,Da.Cn],[C1,0,0])$
(C19) D%:MATRIX([Ds],[0],[D1])$
(C20) P:MATRIX([Ps,Pa,Pn],[T(Pa),Paa,Pan],[T(Pn),T(Pan),Pnn])$
(C21) Q%:MATRIX([Qmp,-Qpm,0],[-Qpm,Qmp,0],[0,0,rho*R])$
(C22) R1:T('C%) . 'Q% . 'C%$
(C23) R12:T('C%) . 'Q% . 'D%$
(C24) RR2:T('D%).'Q%.'D%$
(C25) RICCATI:-'P.'RR%.'P+'P.'AR%+T('AR%).'P+'QR%$
(C26) RR%:'B%.[R2^^(-1)].T('B%)$
(C27) AR%:'A%-'B%.[R2^^(-1)].T('R12)$
(C28) QR%:'R1-'R12.[R2^^(-1)].T('R12)$
(C29) BLOCK(
PRINT(" A% =",PR(A%),"          ","B% =",PR(B%),"
","C% =",PR(C%),"        ","D% =",PR(D%),"
"," P =",PR(P),"
","Q% =",Q%,"
","The Algebraic Riccati Equation: ",SUBST('AR%^T,T('AR%),RICCATI),"
","RR% =",SUBST('B%^T,T('B%),SUBST(R[2]^^(-1),[R2^^(-1)],RR%)),"
","AR% =",SUBST('R[1,2]^T,T('R12),SUBST(R[2]^^(-1),[R2^^(-1)],AR%)),"
"," QR% =",SUBST('R[1,2],'R12,SUBST(R[2]^^(-1),[R2^^(-1)],
                  SUBST('R[1],R1,SUBST('R[1,2]^T,T('R12),QR%)))),"
",'R[1],"=",SUBST('C%^T,T('C%),R1),"
",'R[1,2],"=",SUBST('C%^T,T('C%),R12),"
",'R[2],"=",SUBST('D%^T,T('D%),RR2)),
R1:EV(R1,NOUNS),
R12:EV(R12,NOUNS),
RR2:EV(RR2,NOUNS))$
```

$$A\% = \begin{bmatrix} A_s & 0 & 0 \\ 0 & A_a & B_a . C_n \\ 0 & 0 & A_n \end{bmatrix} \qquad B\% = \begin{bmatrix} B_s \\ 0 \\ 0 \end{bmatrix}$$

$$C\% = \begin{bmatrix} C_s & 0 & 0 \\ 0 & C_a & D_a . C_n \\ C_1 & 0 & 0 \end{bmatrix} \qquad D\% = \begin{bmatrix} D_s \\ 0 \\ D_1 \end{bmatrix}$$

$$P = \begin{bmatrix} P_s & P_a & P_n \\ P_a^T & P_{a, a} & P_{a, n} \\ P_n^T & P_{a, n}^T & P_{n, n} \end{bmatrix}$$

$$Q\% = \begin{bmatrix} Qmp & -Qpm & 0 \\ -Qpm & Qmp & 0 \\ 0 & 0 & rho\ R \end{bmatrix}$$

The Algebraic Riccati Equation: $QR\% - P . RR\% . P + P . AR\% + AR\%^T . P$

$$RR\% = B\% . R_2^{<-1>} . B\%^T$$

$$AR\% = A\% - B\% . R_2^{<-1>} . R_{1, 2}^T$$

$$QR\% = R1 - R_{1, 2} . R_2^{<-1>} . R_{1, 2}^T$$

$$R_1 = C\%^T . Q\% . C\% \qquad R_{1, 2} = C\%^T . Q\% . D\% \qquad R_2 = D\%^T . Q\% . D\%$$

```
(C30) BLOCK(
PRINT('R[1],"=",PR(R1),"
",'R[1,2],"=",PR(R12),"
",'R[2],"=",PR(RR2)))$
```

$$R_1 = \mathrm{Col}\ 1 = \begin{bmatrix} C_s^T . Qmp . C_s + rho\ (C_1^T . R . C_1) \\ \\ - C_a^T . Qpm . C_s \\ \\ - C_n^T . D_a^T . Qpm . C_s \end{bmatrix}$$

$$\mathrm{Col}\ 2 = \begin{bmatrix} - C_s^T . Qpm . C_a \\ \\ C_a^T . Qmp . C_a \\ \\ C_n^T . D_a^T . Qmp . C_a \end{bmatrix} \mathrm{Col}\ 3 = \begin{bmatrix} - C_s^T . Qpm . D_a . C_n \\ \\ C_a^T . Qmp . D_a . C_n \\ \\ C_n^T . D_a^T . Qmp . D_a . C_n \end{bmatrix}$$

$$R_{1,\ 2} = \begin{bmatrix} C_s^T . Qmp . D_s + rho\ (C_1^T . R . D_1) \\ \\ - C_a^T . Qpm . D_s \\ \\ - C_n^T . D_a^T . Qpm . D_s \end{bmatrix}$$

$$R_2 = D_s^T . Qmp . D_s + rho\ (D_1^T . R . D_1)$$

```
(C31) RR%:FACTOR(EV(RR%,NOUNS,EXPAND))$
(C32) AR%:A%+FACTOR(SUBST(Qmp,T(Qmp),SUBST(Qpm,T(Qpm),
        SUBST(R2^^(-1),T(R2^^(-1)),SUBST(R,T(R),
        EV(AR%-'A%,NOUNS,EXPAND))))))$
(C33) QR%:FACTOR(SUBST(Qpm,T(Qpm),SUBST(Qmp,T(Qmp),SUBST(R2^^(-1),T(R2^^(-1)),
        SUBST(R,T(R),
        EV(QR%-'R1,NOUNS,EXPAND))))))+FACTOR(R1)$


(C34) EQ:EV(RICCATI,NOUNS)$


(C35) EQ:SUBST(Qmp,T(Qmp),EQ)$
(C36) EQ:SUBST(Qpm,T(Qpm),EQ)$
(C37) EQ:SUBST(Q,T(Q),EQ)$
(C38) EQ:SUBST(R,T(R),EQ)$
(C39) EQ:SUBST(R2^^(-1),T(R2^^(-1)),EQ)$
(C40) EQ:SUBST((Qm-Qp),Qmp,SUBST((Qm+Qp),Qpm,EQ))$

(C41) EQ11:ISOLATE(EQ[1,1],Ps)$
(C44) EQ12:ISOLATE(EQ[1,2],Pa)$
(C45) EQ13:ISOLATE(EQ[1,3],Pn)$

(C46) (FOR ii:1 THRU 3 DO
       FOR ij:ii THRU 3 DO
          PRINT("

","Block ",ii,",",ij,"
",PR(EQ[ii,ij]))),
PRINT("

","Block 1 , 1
",PR(EQ[1,1]),"

","Block 1 , 2
",PR(EQ[1,2]),"

","Block 1 , 3
",PR(EQ[1,3]),"

","Block 1 , 1
",PR(EQ11),"

","Block 1 , 2
",PR(EQ12),"

","Block 1 , 3
",PR(EQ13),"

","Subexpreations

"),
(FOR i:LENGTH(LABELS(E)) STEP -1 THRU 1 DO PRINT
(LABELS(E)[i],"=",PR(EV(LABELS(E)[i])),"
"))$
```

Block 1 , 1

$$(- C_s^T . (Qm - Qp) . D_s . R_2^{<-1>} . B_s^T + A_s^T$$

$$- \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T)) . P_s$$

$$+ P_s . (- B_s . R_2^{<-1>} . D_s^T . (Qm - Qp) . C_s$$

$$- \mathrm{rho}\ (B_s . R_2^{<-1>} . D_1^T . R . C_1) + A_s) - P_s . B_s . R_2^{<-1>} . B_s^T . P_s$$

$$- C_s^T . (Qm - Qp) . D_s . R_2^{<-1>} . D_s^T . (Qm - Qp) . C_s$$

$$- \mathrm{rho}\ (C_s^T . (Qm - Qp) . D_s . R_2^{<-1>} . D_1^T . R . C_1)$$

$$+ C_s^T . (Qm - Qp) . C_s - \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . (Qm - Qp) . C_s)$$

$$- \mathrm{rho}^2\ (C_1^T . R . D_1 . R_2^{<-1>} . D_1^T . R . C_1) + \mathrm{rho}\ (C_1^T . R . C_1)$$

Block 1 , 2

$$(- C_s^T . (Qm - Qp) . D_s . R_2^{<-1>} . B_s^T + A_s^T$$

$$- \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . B_s^T)) . P_a$$

$$+ P_s . B_s . R_2^{<-1>} . D_s^T . (Qp + Qm) . C_a - P_s . B_s . R_2^{<-1>} . B_s^T . P_a$$

$$- C_s^T . (Qp + Qm) . C_a + C_s^T . (Qm - Qp) . D_s . R_2^{<-1>} . D_s^T . (Qp + Qm) . C_a$$

$$+ \mathrm{rho}\ (C_1^T . R . D_1 . R_2^{<-1>} . D_s^T . (Qp + Qm) . C_a) + P_a . A_a$$

```
Block 1 , 3

    T                     <- 1>    T    T
(- C  . (Qm - Qp) . D  . R      . B  + A
    s                s    2        s    s

        T             <- 1>    T
- rho (C  . R . D  . R      . B )) . P
        1        1    2        s      n

             <- 1>    T
+ P  . B  . R      . D  . (Qp + Qm) . D  . C
   s    s    2        s                a    n

             <- 1>    T         T
- P  . B  . R      . B  . P  - C  . (Qp + Qm) . D  . C
   s    s    2        s    n    s                a    n

   T                     <- 1>    T
+ C  . (Qm - Qp) . D  . R      . D  . (Qp + Qm) . D  . C  + P  . A
   s                s    2        s                a    n    n    n

        T             <- 1>    T
+ rho (C  . R . D  . R      . D  . (Qp + Qm) . D  . C )
        1        1    2        s                a    n

+ P  . B  . C
   a    a    n
```

Block 1 , 1

$$- P_s \, . \, B_s \, . \, R_2^{<-1>} \, . \, B_s^T \, . \, P_s + P_s \, . \, E42 + E43 + E42^T \, . \, P_s$$

Block 1 , 2

$$- P_s \, . \, B_s \, . \, R_2^{<-1>} \, . \, B_s^T \, . \, P_a + P_a \, . \, A_a + E44 + E42^T \, . \, P_a$$

Block 1 , 3

$$- P_s \, . \, B_s \, . \, R_2^{<-1>} \, . \, B_s^T \, . \, P_n + P_n \, . \, A_n + E45 + E42^T \, . \, P_n$$

Subexpreations

$$E42 = - B_{s} . R_{2}^{<-1>} . D_{s}^{T} . (Qm - Qp) . C_{s}$$

$$- \mathrm{rho}\ (B_{s} . R_{2}^{<-1>} . D_{1}^{T} . R . C_{1}) + A_{s}$$

$$E43 = - C_{s}^{T} . (Qm - Qp) . D_{s} . R_{2}^{<-1>} . D_{s}^{T} . (Qm - Qp) . C_{s}$$

$$- \mathrm{rho}\ (C_{s}^{T} . (Qm - Qp) . D_{s} . R_{2}^{<-1>} . D_{1}^{T} . R . C_{1})$$

$$+ C_{s}^{T} . (Qm - Qp) . C_{s} - \mathrm{rho}\ (C_{1}^{T} . R . D_{1} . R_{2}^{<-1>} . D_{s}^{T} . (Qm - Qp)$$

$$. C_{s}) - \mathrm{rho}^{2}\ (C_{1}^{T} . R . D_{1} . R_{2}^{<-1>} . D_{1}^{T} . R . C_{1}) + \mathrm{rho}\ (C_{1}^{T} . R . C_{1})$$

$$E44 = P_{s} . B_{s} . R_{2}^{<-1>} . D_{s}^{T} . (Qp + Qm) . C_{a} - C_{s}^{T} . (Qp + Qm) . C_{a}$$

$$+ C_{s}^{T} . (Qm - Qp) . D_{s} . R_{2}^{<-1>} . D_{s}^{T} . (Qp + Qm) . C_{a}$$

$$+ \mathrm{rho}\ (C_{1}^{T} . R . D_{1} . R_{2}^{<-1>} . D_{s}^{T} . (Qp + Qm) . C_{a})$$

$$E45 = P_{s} . B_{s} . R_{2}^{<-1>} . D_{s}^{T} . (Qp + Qm) . D_{a} . C_{n}$$

$$- C_{s}^{T} . (Qp + Qm) . D_{a} . C_{n} + C_{s}^{T} . (Qm - Qp) . D_{s} . R_{2}^{<-1>} . D_{s}^{T}$$

$$. (Qp + Qm) . D_{a} . C_{n} + \mathrm{rho}\ (C_{1}^{T} . R . D_{1} . R_{2}^{<-1>} . D_{s}^{T} . (Qp + Qm)$$

$$. D_{a} . C_{n}) + P_{a} . B_{a} . C_{n}$$

Appendix IV.C: Derivation of Block Equations for a "Sign Senstive"
------------- Cost with a Term Added (involvs Qs) to Remove
Cost Singularity (Macsyma output edited a bit).

```
(D4)                Thursday, Aug 26, 1982  7:14am
(C5) ALIAS(T,TRANSPOSE)$
(C6) BOTHCASES:TRUE$
(C7) MATRIX_ELEMENT_MULT:"."$
(C8) MATRIX_ELEMENT_TRANSPOSE:NONSCALARS$
(C9) DOTSCRULES:TRUE$
(C10) DECLARE([q,rho],SCALAR)$
(C11) DECLARE([Aa,Ba,Ca,Da,As,Bs,Cs,Ds,An,Bn,Cn,A1,B1,C1,D1],NONSCALAR)$
(C12) DECLARE([Paa,Pa,Pan,Ps,Pn,Pnn],NONSCALAR)$
(C13) /* DECLARE([Q,R,R2])$ */
/* not used so that T(Q)=Q, T(R)=R ... (all are symatric matrices) */
DECLARE(T,ADDITIVE)$
(C14) PR(MVAR):=SUBST('A[s],As,SUBST('B[s],Bs,SUBST('C[s],Cs,SUBST('D[s],Ds,
SUBST('A[a],Aa,SUBST('B[a],Ba,SUBST('C[a],Ca,SUBST('D[a],Da,
SUBST('A[1],A1,SUBST('B[1],B1,SUBST('C[1],C1,SUBST('D[1],D1,SUBST('R[2],R2,
SUBST('A[n],An,SUBST('C[n],Cn,SUBST('P[a,a],Paa,SUBST('P[a],Pa,
SUBST('P[a,n],Pan,SUBST('P[s],Ps,SUBST('P[n],Pn,SUBST('P[n,n],Pnn,
SUBST(Aa^T,T(Aa),SUBST(Ba^T,T(Ba),SUBST(Ca^T,T(Ca),SUBST(Da^T,T(Da),
SUBST(As^T,T(As),SUBST(Bs^T,T(Bs),SUBST(Cs^T,T(Cs),SUBST(Ds^T,T(Ds),
SUBST(An^T,T(An),SUBST(Cn^T,T(Cn),SUBST(C1^T,T(C1),SUBST(D1^T,T(D1),
SUBST(Pa^T,T(Pa),SUBST(Pan^T,T(Pan),SUBST(Pn^T,T(Pn),
MVAR))))))))))))))))))))))))))))))))))))$
(C15) TRANSLATE(PR);
(D15)                            [PR]
(C16) A%:MATRIX([As,0,0],[0,Aa,Ba,Cn],[0,0,An])$
(C17) B%:MATRIX([Bs],[0],[0])$
(C20) P:MATRIX([Ps,Pa,Pn],[T(Pa),Paa,Pan],[T(Pn),T(Pan),Pnn])$
(C21) Q%:MATRIX([Qsmp,-Qpm,0],[-Qpm,Qmp,0],[0,0,rho*R])$
(C22) R1:T('C%) . 'Q% . 'C%$
(C23) R12:T('C%) . 'Q% . 'D%$
(C24) RR2:T('D%).'Q%.'D%$
(C25) RICCATI:-'P.'RR%.'P+'P.'AR%+T('AR%).'P+'QR%$
(C26) RR%:'B%.[R2^^(-1)].T('B%)$
(C27) AR%:'A%-'B%.[R2^^(-1)].T('R12)$
(C28) QR%:'R1-'R12.[R2^^(-1)].T('R12)$
(C29) BLOCK(
PRINT(" A% =",PR(A%),"          ","B% =",PR(B%),"
","C% =",PR(C%),"         ","D% =",PR(D%),"
"," P =",PR(P),"
","Q% =",SUBST((Qs+Qmp),Qsmp,Q%),"
","The Algebraic Riccati Equation: ",SUBST('AR%^T,T('AR%),RICCATI),"
","RR% =",SUBST('B%^T,T('B%),SUBST(R[2]^^(-1),[R2^^(-1)],RR%)),"
","AR% =",SUBST('R[1,2]^T,T('R12),SUBST(R[2]^^(-1),[R2^^(-1)],AR%)),"
"," QR% =",SUBST('R[1,2],'R12,SUBST(R[2]^^(-1),[R2^^(-1)],
                 SUBST('R[1],R1,SUBST('R[1,2]^T,T('R12),QR%)))),"
",'R[1],"=",SUBST('C%^T,T('C%),R1),"
",'R[1,2],"=",SUBST('C%^T,T('C%),R12),"
",'R[2],"=",SUBST('D%^T,T('D%),RR2)),
R1:EV(R1,NOUNS),
R12:EV(R12,NOUNS),
RR2:EV(RR2,NOUNS))$
```

$$A\% = \begin{bmatrix} A_s & 0 & 0 \\ 0 & A_a & B_a . C_n \\ 0 & 0 & A_n \end{bmatrix} \qquad B\% = \begin{bmatrix} B_s \\ 0 \\ 0 \end{bmatrix}$$

$$C\% = \begin{bmatrix} C_s & 0 & 0 \\ 0 & C_a & D_a . C_n \\ C_1 & 0 & 0 \end{bmatrix} \qquad D\% = \begin{bmatrix} D_s \\ 0 \\ D_1 \end{bmatrix}$$

$$P = \begin{bmatrix} P_s & P_a & P_n \\ P_a^T & P_{a,a} & P_{a,n} \\ P_n^T & P_{a,n}^T & P_{n,n} \end{bmatrix}$$

$$Q\% = \begin{bmatrix} Qs + Qmp & - Qpm & 0 \\ - Qpm & Qmp & 0 \\ 0 & 0 & rho\ R \end{bmatrix} \qquad \begin{aligned} Qsmp &= Qs + Qm - Qp \\ Qmp &= Qm - Qp \\ Qpm &= Qm + Qp \end{aligned}$$

The Algebraic Riccati Equation: $QR\% - P . RR\% . P + P . AR\% + AR\%^T . P$

$$RR\% = B\% . R_2^{<-1>} . B\%^T$$

$$AR\% = A\% - B\% . R_2^{<-1>} . R_{1,2}^T$$

$$QR\% = R1 - R_{1,2} . R_2^{<-1>} . R_{1,2}^T$$

$$R_1 = C\%^T . Q\% . C\% \qquad R_{1,2} = C\%^T . Q\% . D\% \qquad R_2 = D\%^T . Q\% . D\%$$

```
(C30) BLOCK(
PRINT('R[1],"=",PR(R1),"
",'R[1,2],"=",PR(R12),"
",'R[2],"=",PR(RR2)))$
```

```
              [  T                     T           ]
              [ C  . Qsmp . C  + rho (C  . R . C ) ]
              [  s           s         1        1  ]
              [                                    ]
              [             T                      ]
R  =  Col 1 = [          - C  . Qpm . C            ]
 1            [             a          s           ]
              [                                    ]
              [           T    T                   ]
              [        - C  . D  . Qpm . C         ]
              [           n    a          s        ]

         [     T              ]         [     T                    ]
         [  - C  . Qpm . C    ]         [  - C  . Qpm . D  . C    ]
         [     s          a   ]         [     s          a    n   ]
         [                    ]         [                         ]
         [    T               ]         [    T                    ]
 Col 2 = [   C  . Qmp . C     ] Col 3 = [   C  . Qmp . D  . C     ]
         [    a          a    ]         [    a          a    n    ]
         [                    ]         [                         ]
         [  T    T            ]         [  T    T                 ]
         [ C  . D  . Qmp . C  ]         [ C  . D  . Qmp . D  . C  ]
         [  n    a          a ]         [  n    a          a    n ]


         [  T                     T           ]
         [ C  . Qsmp . D  + rho (C  . R . D ) ]
         [  s           s         1        1  ]
         [                                    ]
         [             T                      ]
 R     = [          - C  . Qpm . D            ]
  1, 2   [             a          s           ]
         [                                    ]
         [           T    T                   ]
         [        - C  . D  . Qpm . D         ]
         [           n    a          s        ]


       T                     T
 R  = D  . Qsmp . D  + rho (D  . R . D )
  2    s           s         1        1
```

```
(C31) RR%:FACTOR(EV(RR%,NOUNS,EXPAND))$
(C32) AR%:A%+FACTOR(SUBST(Qsmp,T(Qsmp),SUBST(Qmp,T(Qmp),SUBST(Qpm,T(Qpm),
        SUBST(R2^^(-1),T(R2^^(-1)),SUBST(R,T(R),
        EV(AR%-'A%,NOUNS,EXPAND)))))))$
(C33) QR%:FACTOR(SUBST(Qsmp,T(Qsmp),SUBST(Qpm,T(Qpm),SUBST(Qmp,T(Qmp),
        SUBST(R2^^(-1),T(R2^^(-1)),SUBST(R,T(R),
        EV(QR%-'R1,NOUNS,EXPAND)))))))+FACTOR(R1)$


(C34) EQ:EV(RICCATI,NOUNS)$


(C35) EQ:SUBST(Qsmp,T(Qsmp),EQ)$
(C36) EQ:SUBST(Qmp,T(Qmp),EQ)$
(C37) EQ:SUBST(Qpm,T(Qpm),EQ)$
(C38) EQ:SUBST(Q,T(Q),EQ)$
(C39) EQ:SUBST(R,T(R),EQ)$
(C40) EQ:SUBST(R2^^(-1),T(R2^^(-1)),EQ)$
(C41) EQ:SUBST((Qs+Qm-Qp),Qsmp,SUBST((Qm-Qp),Qmp,SUBST((Qm+Qp),Qpm,EQ)))$

(C42) EQ11:ISOLATE(EQ[1,1],Ps)$
(C45) EQ12:ISOLATE(EQ[1,2],Pa)$
(C46) EQ13:ISOLATE(EQ[1,3],Pn)$



(C47) PRINT("
"),
PRINT("

","Block 1 , 1
",PR(EQ[1,1]),"

","Block 1 , 2
",PR(EQ[1,2]),"

","Block 1 , 3
",PR(EQ[1,3]),"

","Block 1 , 1
",PR(EQ11),"

","Block 1 , 2
",PR(EQ12),"

","Block 1 , 3
",PR(EQ13),"

","Subexpreations

"),
(FOR i:LENGTH(LABELS(E)) STEP -1 THRU 1 DO PRINT
(LABELS(E)[i],"=",PR(EV(LABELS(E)[i])),"
"))$
```

Block 1 , 1

$$(- C_s^T . (Qs - Qp + Qm) . D_s . R_2^{<- 1>} . B_s^T + A_s^T$$

$$- \mathrm{rho} (C_1^T . R . D_1 . R_2^{<- 1>} . B_s^T)) . P_s$$

$$+ P_s . (- B_s . R_2^{<- 1>} . D_s^T . (Qs - Qp + Qm) . C_s$$

$$- \mathrm{rho} (B_s . R_2^{<- 1>} . D_1^T . R . C_1) + A_s) - P_s . B_s . R_2^{<- 1>} . B_s^T . P_s$$

$$- C_s^T . (Qs - Qp + Qm) . D_s . R_2^{<- 1>} . D_s^T . (Qs - Qp + Qm) . C_s$$

$$- \mathrm{rho} (C_s^T . (Qs - Qp + Qm) . D_s . R_2^{<- 1>} . D_1^T . R . C_1)$$

$$+ C_s^T . (Qs - Qp + Qm) . C_s - \mathrm{rho} (C_1^T . R . D_1 . R_2^{<- 1>} . D_s^T$$

$$. (Qs - Qp + Qm) . C_s) - \mathrm{rho}^2 (C_1^T . R . D_1 . R_2^{<- 1>} . D_1^T . R . C_1)$$

$$+ \mathrm{rho} (C_1^T . R . C_1)$$

Block 1 , 2

$$(- C_s^T . (Qs - Qp + Qm) . D_s . R_2^{<- 1>} . B_s^T + A_s^T$$

$$- \text{rho} \, (C_1^T . R . D_1 . R_2^{<- 1>} . B_s^T)) . P_a$$

$$+ P_s . B_s . R_2^{<- 1>} . D_s^T . (Qp + Qm) . C_a - P_s . B_s . R_2^{<- 1>} . B_s^T . P_a$$

$$+ C_s^T . (Qs - Qp + Qm) . D_s . R_2^{<- 1>} . D_s^T . (Qp + Qm) . C_a$$

$$- C_s^T . (Qp + Qm) . C_a + \text{rho} \, (C_1^T . R . D_1 . R_2^{<- 1>} . D_s^T . (Qp + Qm)$$

$$. C_a) + P_a . A_a$$

Block 1 , 3

$$(- C_s^T . (Qs - Qp + Qm) . D_s . R_2^{<- 1>} . B_s^T + A_s^T$$

$$- \text{rho} \, (C_1^T . R . D_1 . R_2^{<- 1>} . B_s^T)) . P_n$$

$$+ P_s . B_s . R_2^{<- 1>} . D_s^T . (Qp + Qm) . D_a . C_n$$

$$- P_s . B_s . R_2^{<- 1>} . B_s^T . P_n + C_s^T . (Qs - Qp + Qm) . D_s . R_2^{<- 1>} . D_s^T$$

$$. (Qp + Qm) . D_a . C_n - C_s^T . (Qp + Qm) . D_a . C_n + P_n . A_n$$

$$+ \text{rho} \, (C_1^T . R . D_1 . R_2^{<- 1>} . D_s^T . (Qp + Qm) . D_a . C_n)$$

$$+ P_a . B_a . C_n$$

Block 1 , 1

$$- P_s \, . \, B_s \, . \, R_2^{<-1>} \, . \, B_s^T \, . \, P_s + P_s \, . \, E43 + E44 + E43^T \, . \, P_s$$

Block 1 , 2

$$- P_s \, . \, B_s \, . \, R_2^{<-1>} \, . \, B_s^T \, . \, P_a + P_a \, . \, A_a + E45 + E43^T \, . \, P_a$$

Block 1 , 3

$$- P_s \, . \, B_s \, . \, R_2^{<-1>} \, . \, B_s^T \, . \, P_n + P_n \, . \, A_n + E46 + E43^T \, . \, P_n$$

```
Subexpreations

                 <- 1>    T
 E43 = - B    . R      . D  . (Qs - Qp + Qm) . C
          s      2        s                     s

                 <- 1>    T
  - rho (B    . R      . D  . R . C ) + A
          s      2        1        1      s

          T                          <- 1>    T
 E44 = - C  . (Qs - Qp + Qm) . D  . R      . D  . (Qs - Qp + Qm) . C
          s                     s    2        s                     s

          T                          <- 1>    T
  - rho (C  . (Qs - Qp + Qm) . D  . R      . D  . R . C )
          s                     s    2        1        1

     T                               T             <- 1>    T
  + C  . (Qs - Qp + Qm) . C  - rho (C  . R . D  . R      . D
     s                     s         1        1    2        s

                               2   T             <- 1>    T
  . (Qs - Qp + Qm) . C  ) - rho  (C  . R . D  . R      . D  . R . C )
                      s            1        1    2        1        1

          T
  + rho (C  . R . C )
          1        1

                  <- 1>    T
 E45 = P  . B  . R      . D  . (Qp + Qm) . C
        s    s    2        s                a

                                <- 1>    T
  + C  . (Qs - Qp + Qm) . D  . R      . D  . (Qp + Qm) . C
     T
     s                     s    2        s                a

     T                          T             <- 1>    T
  - C  . (Qp + Qm) . C  + rho (C  . R . D  . R      . D  . (Qp + Qm) . C )
     s                a         1        1    2        s                a

                  <- 1>    T
 E46 = P  . B  . R      . D  . (Qp + Qm) . D  . C
        s    s    2        s                a    n

     T                          <- 1>    T
  + C  . (Qs - Qp + Qm) . D  . R      . D  . (Qp + Qm) . D  . C
     s                     s    2        s                a    n

     T                               T             <- 1>    T
  - C  . (Qp + Qm) . D  . C  + rho (C  . R . D  . R      . D
     s                a    n         1        1    2        s

  . (Qp + Qm) . D  . C ) + P  . B  . C
                 a    n     a    a    n

(C48) STATUS(RUNTIME);                    (D48)  71492 MSEC
```

```
/*    Appendix IV.P-A:  Macsyma Program */
/*    ---------------                   */


TIMEDATE();
DECLARE(E,ADDITIVE)$
J:E((FFB-FX*X-FU*U)^2);
JE:EV(EXPAND(%));
JX:DIFF(JE,FX);
JU:DIFF(JE,FU);
JX:EV(SUBST(FX^2*E(X^2),E(FX^2*X^2),JX),INFEVAL,NOUNS)$
JX:EV(SUBST(2*FU*FX*E(U*X),E(2*FU*FX*U*X),JX),INFEVAL,NOUNS)$
JX:EV(SUBST(-2*FX*E(FFB*X),E(-2*FFB*FX*X),JX),INFEVAL,NOUNS);
JU:EV(SUBST(FU*FX*2*E(X*U),E(X*U*2*FU*FX),JU),INFEVAL,NOUNS)$
JU:EV(SUBST(FU^2*E(U^2),E(FU^2*U^2),JU),INFEVAL,NOUNS)$
JU:EV(SUBST(-2*FU*E(FFB*U),E(-2*FFB*FU*U),JU),INFEVAL,NOUNS);
FXFU:SOLVE([JX,JU],[FX,FU])$
FX:PART(FXFU,1,1);
FU:PART(FXFU,1,2);
```

```
/*    Appendix IV.P-B: Macsyma Program    */
/*    ---------------                     */
LINEL:70$
WRITEFILE(DSK,JEHUDA)$
TIMEDATE();
ALIAS(T,TRANSPOSE)$
BOTHCASES:TRUE$
MATRIX_ELEMENT_MULT:"."$
MATRIX_ELEMENT_TRANSPOSE:NONSCALARS$
DOTSCRULES:TRUE$
DECLARE([q,rho],SCALAR)$
DECLARE([Aa,Ba,Ca,Da,As,Bs,Cs,Ds,An,Bn,Cn,A1,B1,C1,D1],NONSCALAR)$
DECLARE([Paa,Pa,Pan,Ps,Pn,Pnn],NONSCALAR)$
/* DECLARE([Q,R,R2])$ */
/* not used so that T(Q)=Q, T(R)=R ... (all are symatric matrices) */
DECLARE(T,ADDITIVE)$
PR(MVAR):=SUBST('A[s],As,SUBST('B[s],Bs,SUBST('C[s],Cs,SUBST('D[s],Ds,
SUBST('A[a],Aa,SUBST('B[a],Ba.SUBST('C[a],Ca,SUBST('D[a],Da,
SUBST('A[1],A1,SUBST('B[1],B1.SUBST('C[1],C1,SUBST('D[1],D1,SUBST('R[2],R2,
SUBST('A[n],An,SUBST('C[n],Cn,SUBST('P[a,a],Paa,SUBST('P[a],Pa,
SUBST('P[a,n],Pan,SUBST('P[s],Ps,SUBST('P[n],Pn,SUBST('P[n,n],Pnn,
SUBST(Aa^T,T(Aa),SUBST(Ba^T,T(Ba),SUBST(Ca^T,T(Ca),SUBST(Da^T,T(Da),
SUBST(As^T,T(As),SUBST(Bs^T,T(Bs),SUBST(Cs^T,T(Cs),SUBST(Ds^T,T(Ds),
SUBST(An^T,T(An),SUBST(Cn^T,T(Cn),SUBST(C1^T,T(C1),SUBST(D1^T,T(D1),
SUBST(Pa^T,T(Pa),SUBST(Pan^T,T(Pan),SUBST(Pn^T,T(Pn),
MVAR))))))))))))))))))))))))))))))))))))$
TRANSLATE(PR);
A%:MATRIX([As,0,0],[0,Aa,Ba.Cn],[0,0,An])$
B%:MATRIX([Bs],[0],[0])$
C%:MATRIX([Cs,0,0],[0.Ca,Da.Cn],[C1,0,0])$
D%:MATRIX([Ds],[0],[D1])$
P:MATRIX([Ps,Pa,Pn],[T(Pa),Paa,Pan],[T(Pn),T(Pan),Pnn])$
Q%:MATRIX([Qmp,-Qpm,0],[-Qpm,Qmp,0],[0,0,rho*R])$
R1:T('C%) . 'Q% . 'C%$
R12:T('C%) . 'Q% . 'D%$
RR2:T('D%).'Q%.'D%$
RICCATI:-'P.'RR%.'P+'P.'AR%+T('AR%).'P+'QR%$
RR%:'B%.[R2^^(-1)].T('B%)$
AR%:'A%-'B%.[R2^^(-1)].T('R12)$
QR%:'R1-'R12.[R2^^(-1)].T('R12)$
BLOCK(
PRINT(" A% =",PR(A%),"          ","B% =",PR(B%),"
","C% =",PR(C%),"        ","D% =",PR(D%),"
"," P =",PR(P),"
","Q% =",Q%,"
","The Algebraic Riccati Equation: ",SUBST('AR%^T,T('AR%),RICCATI),"
","RR% =",SUBST('B%^T,T('B%),SUBST(R[2]^^(-1),[R2^^(-1)],RR%)),"
","AR% =",SUBST('R[1,2]^T,T('R12),SUBST(R[2]^^(-1),[R2^^(-1)],AR%)),"
"," QR% =",SUBST('R[1,2],'R12,SUBST(R[2]^^(-1),[R2^^(-1)],
                 SUBST('R[1],R1,SUBST('R[1,2]^T,T('R12),QR%)))),"
",'R[1],"=",SUBST('C%^T,T('C%),R1),"
",'R[1,2],"=",SUBST('C%^T,T('C%),R12),"
",'R[2],"=",SUBST('D%^T,T('D%),RR2)),
R1:EV(R1,NOUNS),
R12:EV(R12,NOUNS),
RR2:EV(RR2,NOUNS))$
```

```
BLOCK(
PRINT('R[1],"=",PR(R1),"
",'R[1,2],"=",PR(R12),"
",'R[2],"=",PR(RR2)))$
RR%:FACTOR(EV(RR%,NOUNS,EXPAND))$
AR%:A%+FACTOR(SUBST(Qmp,T(Qmp),SUBST(Qpm,T(Qpm),
        SUBST(R2^^(-1),T(R2^^(-1)),SUBST(R,T(R),
        EV(AR%-'A%,NOUNS,EXPAND))))))$
QR%:FACTOR(SUBST(Qpm,T(Qpm),SUBST(Qmp,T(Qmp),SUBST(R2^^(-1),T(R2^^(-1)),
        SUBST(R,T(R),
        EV(QR%-'R1,NOUNS,EXPAND))))))+FACTOR(R1)$
EQ:EV(RICCATI,NOUNS)$
EQ:SUBST(Qmp,T(Qmp),EQ)$
EQ:SUBST(Qpm,T(Qpm),EQ)$
EQ:SUBST(Q,T(Q),EQ)$
EQ:SUBST(R,T(R),EQ)$
EQ:SUBST(R2^^(-1),T(R2^^(-1)),EQ)$
EQ:SUBST((Qm-Qp),Qmp,SUBST((Qm+Qp),Qpm,EQ))$
EQ11:ISOLATE(EQ[1,1],Ps)$
EQ12:ISOLATE(EQ[1,2],Pa)$
EQ13:ISOLATE(EQ[1,3],Pn)$
PRINT("
"),
PRINT("
","Block 1 , 1
",PR(EQ[1,1]),"

","Block 1 , 2
",PR(EQ[1,2]),"

","Block 1 , 3
",PR(EQ[1,3]),"

","Block 1 , 1
",PR(EQ11),"

","Block 1 , 2
",PR(EQ12),"

","Block 1 , 3
",PR(EQ13),"

","Subexpreations

"),
(FOR i:LENGTH(LABELS(E)) STEP -1 THRU 1 DO PRINT
(LABELS(E)[i],"=",PR(EV(LABELS(E)[i])),"
"))$
STATUS(RUNTIME);
CLOSEFILE(OUTPUT,>);
```

```
/*    Appendix IV.P-C: Macsyma Program    */
/*    ---------------                     */
LINEL:70$
WRITEFILE(DSK,JEHUDA)$
TIMEDATE();
ALIAS(T,TRANSPOSE)$
BOTHCASES:TRUE$
MATRIX_ELEMENT_MULT:"."$
MATRIX_ELEMENT_TRANSPOSE:NONSCALARS$
DOTSCRULES:TRUE$
DECLARE([q,rho],SCALAR)$
DECLARE([Aa,Ba,Ca,Da,As,Bs,Cs,Ds,An,Bn,Cn,Al,Bl,Cl,Dl],NONSCALAR)$
DECLARE([Paa,Pa,Pan,Ps,Pn,Pnn],NONSCALAR)$
/* DECLARE([Q,R,R2])$ */
/* not used so that T(Q)=Q, T(R)=R ... (all are symatric matrices) */
DECLARE(T,ADDITIVE)$
PR(MVAR):=SUBST('A[s],As,SUBST('B[s],Bs,SUBST('C[s],Cs,SUBST('D[s],Ds,
SUBST('A[a],Aa.SUBST('B[a],Ba,SUBST('C[a],Ca,SUBST('D[a],Da,
SUBST('A[l],Al,SUBST('B[l],Bl.SUBST('C[l],Cl,SUBST('D[l],Dl,SUBST('R[2],R2,
SUBST('A[n],An.SUBST('C[n],Cn,SUBST('P[a,a].Paa,SUBST('P[a],Pa,
SUBST('P[a,n],Pan,SUBST('P[s],Ps,SUBST('P[n],Pn,SUBST('P[n,n],Pnn,
SUBST(Aa^T,T(Aa),SUBST(Ba^T,T(Ba),SUBST(Ca^T,T(Ca),SUBST(Da^T,T(Da),
SUBST(As^T,T(As),SUBST(Bs^T,T(Bs),SUBST(Cs^T,T(Cs),SUBST(Ds^T,T(Ds),
SUBST(An^T,T(An),SUBST(Cn^T,T(Cn),SUBST(Cl^T,T(Cl),SUBST(Dl^T,T(Dl),
SUBST(Pa^T,T(Pa),SUBST(Pan^T,T(Pan),SUBST(Pn^T,T(Pn),
MVAR))))))))))))))))))))))))))))))))))))$
TRANSLATE(PR);
A%:MATRIX([As,0,0],[0,Aa,Ba.Cn],[0,0,An])$
B%:MATRIX([Bs],[0],[0])$
C%:MATRIX([Cs,0,0],[0,Ca,Da.Cn],[Cl,0,0])$
D%:MATRIX([Ds],[0],[Dl])$
P:MATRIX([Ps,Pa,Pn],[T(Pa),Paa,Pan],[T(Pn),T(Pan),Pnn])$
Q%:MATRIX([Qsmp,-Qpm,0],[-Qpm,Qmp,0],[0,0,rho*R])$
R1:T('C%) . 'Q% . 'C%$
R12:T('C%) . 'Q% . 'D%$
RR2:T('D%).'Q%.'D%$
RICCATI:-'P.'RR%.'P+'P.'AR%+T('AR%).'P+'QR%$
RR%:'B%.[R2^^(-1)].T('B%)$
AR%:'A%-'B%.[R2^^(-1)].T('R12)$
QR%:'R1-'R12.[R2^^(-1)].T('R12)$
BLOCK(
PRINT(" A% =",PR(A%),"          ","B% =",PR(B%),"
","C% =",PR(C%),"        ","D% =",PR(D%),"
"," P =",PR(P),"
","Q% =",SUBST((Qs+Qmp),Qsmp,Q%),"
","The Algebraic Riccati Equation: ",SUBST('AR%^T,T('AR%),RICCATI),"
","RR% =",SUBST('B%^T,T('B%),SUBST(R[2]^^(-1),[R2^^(-1)],RR%)),"
","AR% =",SUBST('R[1,2]^T,T('R12),SUBST(R[2]^^(-1),[R2^^(-1)],AR%)),"
"," QR% =",SUBST('R[1,2],'R12,SUBST(R[2]^^(-1),[R2^^(-1)],
                 SUBST('R[1],R1,SUBST('R[1,2]^T,T('R12),QR%)))),"
",'R[1],"=",SUBST('C%^T,T('C%),R1),"
",'R[1,2],"=",SUBST('C%^T,T('C%),R12),"
",'R[2],"=",SUBST('D%^T,T('D%),RR2)),
R1:EV(R1,NOUNS),
R12:EV(R12,NOUNS),
RR2:EV(RR2,NOUNS))$
```

```
BLOCK(
PRINT('R[1],"=",PR(R1),"
",'R[1,2],"=",PR(R12),"
",'R[2],"=",PR(RR2)))$
RR%:FACTOR(EV(RR%,NOUNS,EXPAND))$
AR%:A%+FACTOR(SUBST(Qsmp,T(Qsmp),SUBST(Qmp,T(Qmp),SUBST(Qpm,T(Qpm),
        SUBST(R2^^(-1),T(R2^^(-1)),SUBST(R,T(R),
        EV(AR%-'A%,NOUNS,EXPAND)))))))$
QR%:FACTOR(SUBST(Qsmp,T(Qsmp),SUBST(Qpm,T(Qpm),SUBST(Qmp,T(Qmp),
        SUBST(R2^^(-1),T(R2^^(-1)),SUBST(R,T(R),
        EV(QR%-'R1,NOUNS,EXPAND)))))))+FACTOR(R1)$
EQ:EV(RICCATI,NOUNS)$
EQ:SUBST(Qsmp,T(Qsmp),EQ)$
EQ:SUBST(Qmp,T(Qmp),EQ)$
EQ:SUBST(Qpm,T(Qpm),EQ)$
EQ:SUBST(Q,T(Q),EQ)$
EQ:SUBST(R,T(R),EQ)$
EQ:SUBST(R2^^(-1),T(R2^^(-1)),EQ)$
EQ:SUBST((Qs+Qm-Qp),Qsmp,SUBST((Qm-Qp),Qmp,SUBST((Qm+Qp),Qpm,EQ)))$
EQ11:ISOLATE(EQ[1,1],Ps)$
EQ12:ISOLATE(EQ[1,2],Pa)$
EQ13:ISOLATE(EQ[1,3],Pn)$
PRINT("
"),
PRINT("

","Block 1 , 1
",PR(EQ[1,1]),"

","Block 1 , 2
".PR(EQ[1,2]),"

","Block 1 , 3
",PR(EQ[1,3]),"

","Block 1 , 1
",PR(EQ11),"

","Block 1 , 2
",PR(EQ12),"

","Block 1 , 3
",PR(EQ13),"

","Subexpreations

"),
(FOR i:LENGTH(LABELS(E)) STEP -1 THRU 1 DO PRINT
(LABELS(E)[i],"=",PR(EV(LABELS(E)[i])),"
"))$
STATUS(RUNTIME);
CLOSEFILE(OUTPUT,>);
```

Chapter $\underline{\overline{\text{V}}}$

Design Examples

In this chapter we present several design examples that demonstrate the *optimal washout design methodology* which leads to an Optimal Washout System design (OWS). In the first section, we present a simple example of a design of a simulator which has only "yaw" motion. This *academic* example can be solved analytically and thus can further show the properties of the OWS. In Section 2, we present a design for a simulator that has both lateral linear motion and rotatory motion around a horizontal axis which points forward (i.e. roll motion). This design has some practical applications, and serves as a "prototype" for a full scale six-degree-of-freedom simulator design, due to the axis decoupling theorem proven in chapter $\underline{\overline{\text{III}}}$, subsection $\underline{\overline{\text{III}}}$.5.2. A third design example which was implemented using a Link GAT-1 flight simulator is shown in chapter $\underline{\overline{\text{VII}}}$.

1. A One-Degree-of-Freedom Example

This is an *academic* example of the stochastic subproblem that can be solved analytically and thus can illuminate our design methodology and the effect of the *sign sensitive cost.*

1.1. Derivation of the Washout Filter

We are given an actual rotatory motion along a vertical axis (yaw motion) and our task is to design the rotatory motion of a simulator which similarly rotates around a vertical axis. We denote the actual angular velocity by $u^a(t)\, rad/sec$ and model $u^a(t)$ as a first order stochastic process with break frequency $\eta\, rad/sec$, and mean square value $v\eta/2$. That is, $u^a(t)$ is given by

$$\dot{u}^a(t) = -\eta u^a(t) + \eta n(t) \tag{1}$$

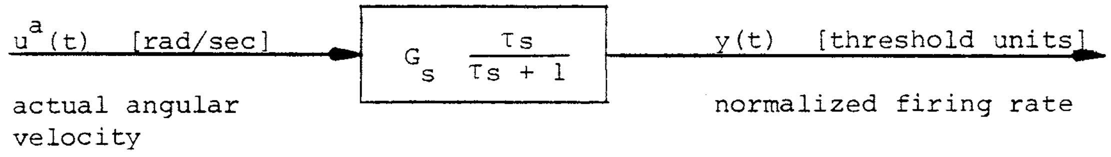


Figure 1. The model of the semicircular canal.

where n(t) is white noise with intensity $v\,(rad/sec)^2 sec$. The values of η and v will be selected so that the spectrum of u^a, as given by (1), best matches the spectrum of the actual motion [Zarchan79].

The part of the vestibular system that perceives rotatory motion is the semicircular canal, which, in order to keep this example as simple as possible, will be modeled by a first order system (Figure 1), with a time constant $\tau = 5.9\,seconds$ and $G_s = 40.2\,sec/rad$. The value of G_s is elected so that one threshold unit of y corresponds to an input u of 1.45 deg/sec^2 at an angular frequency of 0.94 rad/sec. Furthermore, we assume that system $\mathcal{S}^a = \mathcal{S}^s$. The input to the model of the semicircular canal is the angular velocity $u^a(t)$ (or $u^s(t)$) and the output is the normalized firing rate $y^a(t)$ (or $y^s(t)$). The state equation model of the semicircular canal of the pilot in the actual airplane, namely of $\mathcal{S}^a$ ((III.1)-(III.2)) is thus

$\mathcal{S}^a$:

$$\dot{x}^a(t) = -\frac{1}{\tau}x^a(t) - \frac{1}{\tau}u^a(t) \tag{2}$$

$$y^a(t) = G_s x^a(t) + G_s u^a(t) \tag{3}$$

and that for the semicircular canal of the pilot in the simulator is $\mathcal{S}^s$ ((III.4)-(III.5)),

$\mathcal{S}^s$:

$$\dot{x}^s(t) = -\frac{1}{\tau}x^s(t) - \frac{1}{\tau}u^s(t) \tag{4}$$

$$y^a(t) = G_s x^a(t) + G_s u^a(t) \tag{5}$$

$$u^\ell(t) = u^a(t) \tag{6}$$

where we choose $C^\ell = 0$ and $D^\ell = 1$. The stochastic process generation system $\mathcal{N}$ ((III.50)-(III.51)) is

$\mathcal{N}$:

$$\dot{x}^n(t) = -\eta x^n(t) + \eta n(t) \tag{7}$$

$$u^a(t) = x^n(t)\,. \tag{8}$$

The augmented system S ((III.53)–(III.54) is thus

$$\dot{\mathrm{x}}(t) = \begin{pmatrix} -1/\tau & 0 & 0 \\ 0 & -1/\tau & -1/\tau \\ 0 & 0 & -\eta \end{pmatrix} \mathrm{x}(t) + \begin{pmatrix} -1/\tau \\ 0 \\ 0 \end{pmatrix} u^s(t) + \begin{pmatrix} 0 \\ 0 \\ \eta \end{pmatrix} n(t) \tag{9}$$

$$\mathrm{y}^{sa} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \mathrm{x}(t) \tag{10}$$

where we also define

$$e(t) = (G_s\,,\ -G_s\,,\ G_s)\,\mathrm{x}(t) - G_s u^s(t)\,, \tag{11}$$

$$y^{s+a} = y^s + y^a\,. \tag{12}$$

The criterion J (IV.154) using (IV.156) is

$$J = \mathfrak{E}\{e^2(t) - q y^{s+a2}(t) + \rho u^{s2}(t)\} \quad \rho > 0\,,\ \ 0 \leq q < 1\,, \tag{13}$$

where Q in (IV.154) and R in (IV.156) are both set equal to 1. The design parameter ρ represents the relative weight of the control u^s. The design parameter q represents a continues variation between a vestibular error based cost ($q = 0$) to a correlation cost ($q = 1$) (*sign sensitive* cost). The optimal washout filter is given by (III.79),

$$U^s(s) = k \frac{s + \beta/\tau}{s + \alpha/\tau} \frac{s + \gamma/\tau}{s + 1/\tau} U^a(s) \tag{14}$$

where

$$k = -F^n \tag{15}$$
$$\alpha = 1 - F^s \tag{16}$$
$$\beta = 1 - F^a/F^n \tag{17}$$
$$\gamma = 1 + F^a - F^s \tag{18}$$

and, from (IV.166)–(IV.170) and appendix IV.B,

$$F^s = \frac{(1-q)G_s^2 - P^s/\tau}{(1-q)G_s^2 + \rho} \tag{19}$$

$$F^a = -\frac{(1+q)G_s^2 + P^a/\tau}{(1-q)G_s^2 + \rho} \tag{20}$$

$$F^n = -\frac{(1+q)G_s^2 + P^n/\tau}{(1-q)G_s^2 + \rho} \tag{21}$$

and further from Appendix IV.B

$$P^s = \tau\rho\left(\sqrt{1+\frac{1}{\rho}G_s^2(1-q)}-1\right) \geq 0\,, \tag{22}$$

$$P^a = -\tau\rho\frac{\frac{1}{\rho}G_s^2(1+q)\sqrt{1+\frac{1}{\rho}G_s^2(1-q)}}{1+\frac{1}{\rho}G_s^2(1-q)+\sqrt{1+\frac{1}{\rho}G_s^2(1-q)}} < 0\,, \tag{23}$$

$$P^n = -\tau\rho\frac{\frac{1}{\rho}G_s^2(1+q)}{1+\eta\tau\left(1+(1-q)\frac{1}{\rho}G_s^2\right)+(1+\eta\tau)\sqrt{1+\frac{1}{\rho}G_s^2(1-q)}} < 0\,. \tag{24}$$

For the case $q = 0$ (no *sign sensitive* term in J_e) we have (see (III.156)) $\gamma = 1$ (18), and thus (14) simplifies to,

$$U^s(s) = k\frac{s+\beta/\tau}{s+\alpha/\tau}U^a(s)\,. \tag{25}$$

For an exclusive *correlation cost* i.e. $q = 1$, we have $F^s = 0$ ((19), (22)) and $\alpha = 1$ (16), thus (14) simplifies to,

$$U^s(s) = k\frac{(s+\beta/\tau)(s+\gamma/\tau)}{(s+1/\tau)^2}U^a(s) \tag{26}$$

where

$$k(q=1) = -\frac{1}{\rho}G_s^2\frac{1+2\eta\tau}{1+\eta\tau} \tag{27}$$

$$\beta(q=1) = \frac{\eta\tau}{1+2\eta\tau} \tag{28}$$

$$\gamma(q=1) = 1-\frac{1}{\rho}G_s^2\,. \tag{29}$$

A standard computer program package for solving the Linear Quadratic Gaussian problem has been used to obtain numerical results for the gain k, the pole α/τ, and the zero β/τ of the washout filter for the case $q = 0$ (not with a sign sensitive cost). These variables are plotted in Figure 2 and 3 as a function of ρ, for various values of η, with $\tau = 5.9\,sec$ and $q = 0$. The mean square values of $\mathfrak{E}\{e^2(t)\}$ and $\mathfrak{E}\{{u^s}^2(t)\}$ are plotted in Figure 4.

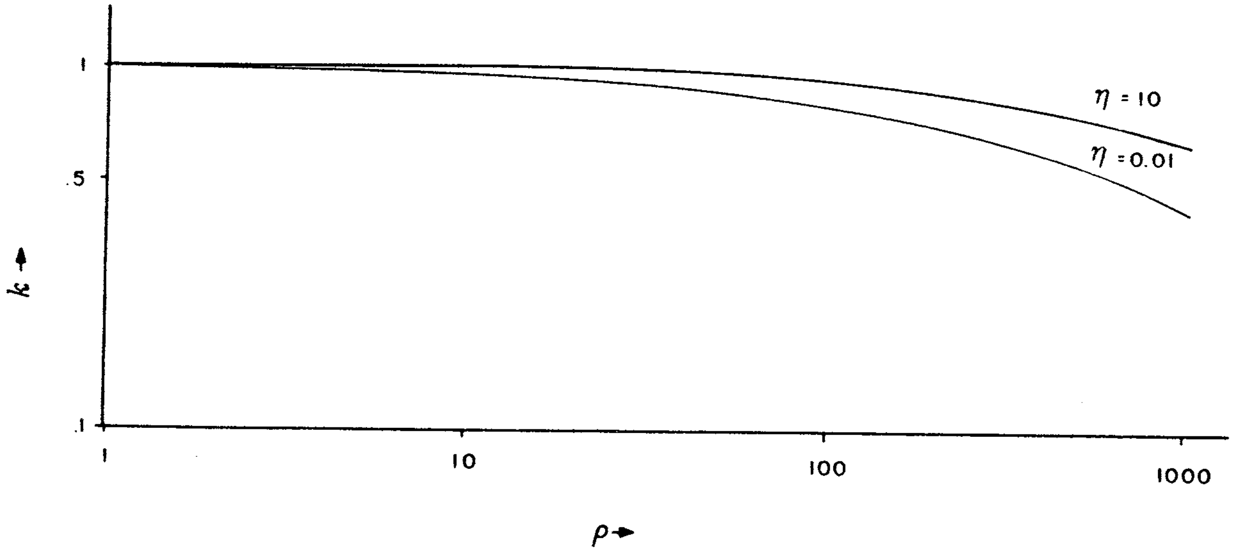


Figure 2. The gain k, as a function of ρ for two extreme values of η *rad/sec*.

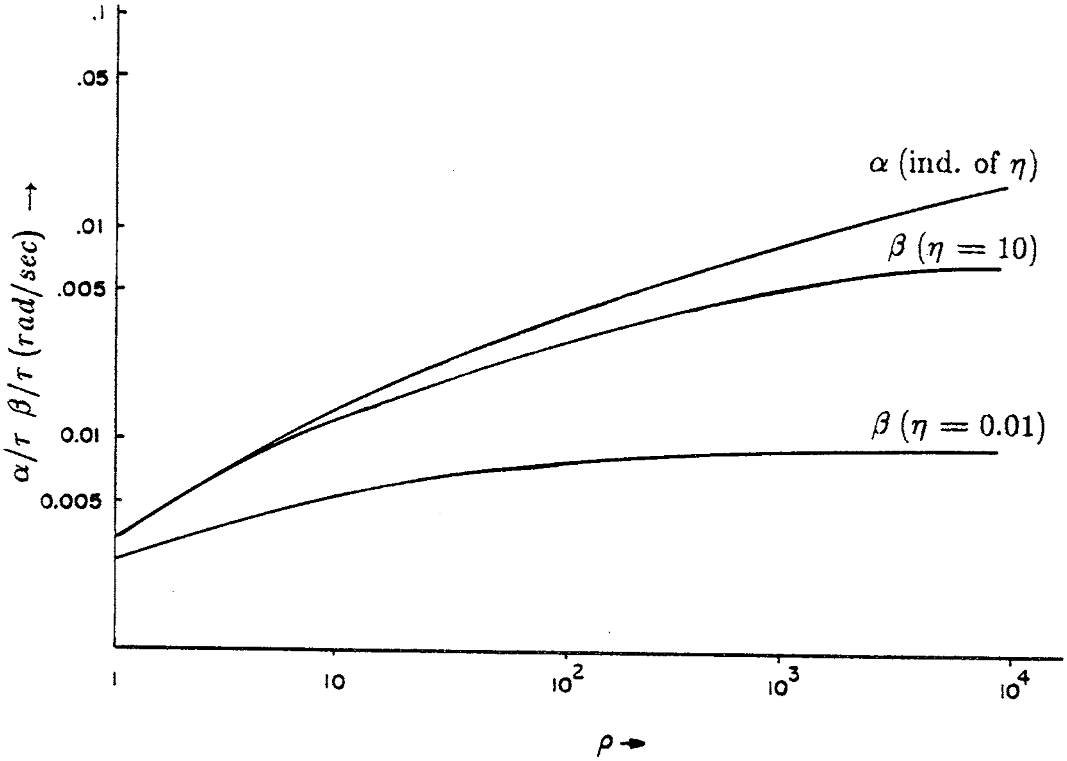


Figure 3. The pole location α/τ, and the zero location β/τ, as a function of ρ for two extreme cases of η *rad/sec*.

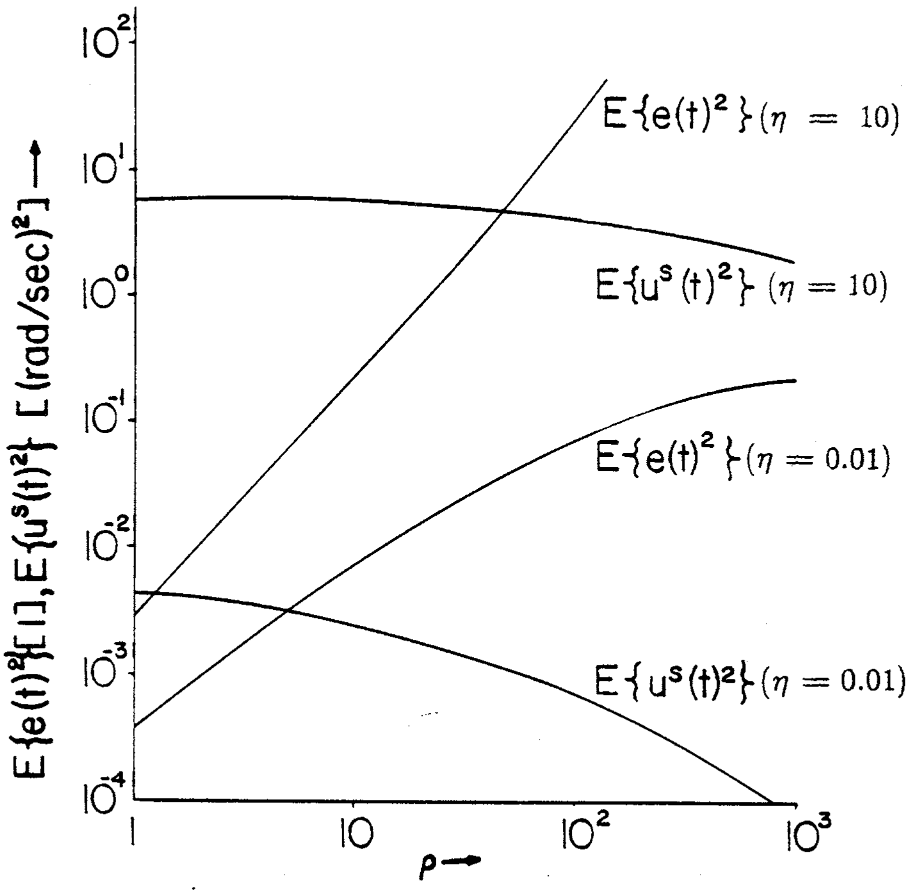


Figure 4. The mean square values of the input to the simulator, $\mathfrak{E}\{u^{s2}(t)\}$, and those of the errors, $\mathfrak{E}\{e^2(t)\}$, as a function of ρ, for two extreme values of $\eta\, rad/sec$.

1.2. Discussion of the Results

The washout filter we just derived is a lead-lag filter with a zero at some $\beta/\tau > 0\, rad/sec$, while conventional designs typically have a zero at the origin. Note, however, that the lower the frequency of the actual motion (i.e., the smaller η is), the closer β/τ is to the origin. This feature that the location of the zero should be a function of the spectrum of the actual motion to be simulated (specifically, the higher the frequency, the further the zero should be from the origin) is usually overlooked in conventional designs. In fact, for high values of η and large values of ρ, $\beta \to \alpha/2$.

Next, note that the pole α/τ does *not* depend upon the spectrum of the actual motion and its location, for relatively small simulator motions (i.e. for large values of

ρ) and is at 0.16 rad/sec. This means that for small simulator motions, the washout filter has a time constant of approximately 5.9 *seconds*, which is the time constant of the semicircular canal. This fact is well known by conventional designers. What is typically overlooked is that for simulators with relatively large excursions (i.e. for small ρ), the poles should have a time constant which is considerably larger than the time constant of the semicircular canal.

Finally, as we expected, when ρ increases (i.e. as the simulator motion decreases), the mean square values of the input to the simulator $\mathfrak{E}\{{u^s}^2(t)\}$ decrease, and the mean square values of the errors $\mathfrak{E}\{e^2(t)\}$ increase. From Figure 4, we can see that increasing ρ increases the errors $e(t)$ considerably, while decreasing the input signal $u^s(t)$ only moderately. Note that the values of η have a significant effect on the mean square values of $e(t)$ and $u^s(t)$.

2. A Two-Degree-of-Freedom Example

In this section, we use the optimal simulator design procedure presented in Chapter III to design the washout filter *matrix* for a two-degree-of-freedom simulator: sway linear motion and roll angular motion (or what leads to an identical problem—surge linear motion and pitch angular motion). We present numerical simulations, Bode plots and root-locus diagrams of the poles and zeros of the designed transfer function of the washout filter. In this example, we assume that the motion-base dynamics can be neglected.

We represent our physiological outputs of the vestibular models by $y^a(t)$ and $y^s(t)$ in *threshold units*. Our design objective is to minimize the difference $e(t)$ of these two signals, so that the simulator pilot's brain will receive a signal which is as similar as possible to the signal received in the actual flight situation. Error signals, e(t), below the numerical value of 1.0 correspond to error signals that are below the pilot's threshold. Since our vestibular model output for both semicircular canal and otolith are in *threshold units*, it is plausible to assume that there is equal sensitivity to each one of these errors, and thus we shall weight them equally.

2.1. The Model for the Vestibular System

We use a linear model for the otolith (Figure 5) where $Sp_y(t)$ is the specific

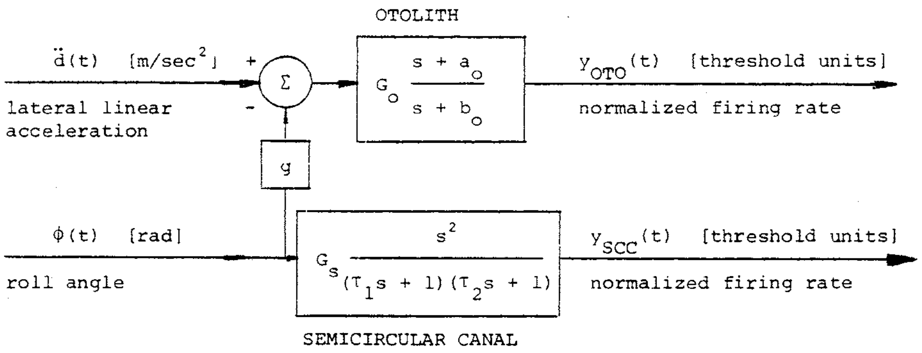


Figure 5. The model of the vestibular system subject to sway linear motion and roll rotation.

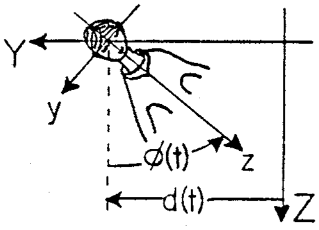


Figure 6. The coordinate axes Y-Z and y-z.

force and $y_{oto}(t)$ is the normalized firing rate. The values of the parameters chosen for this model are justified in [Hosman78, Zacharias78].

$$G_o = 2.16\, s^2/m\,,\ a_o = 0.076\, rad/s\,,\ b_o = 0.19\, rad/s\,. \qquad (30)$$

The value of G_o is selected so that one threshold unit of y_{oto} corresponds to an input Sp_y of $0.47\, m/s^2$ at an angular frequency of $0.94\, rad/s$[1].

The part of the vestibular system that perceives rotatory motion is the semicircular canal. We use a linear model for the semicircular canal (Figure 5) where $\phi(t)$ is the angular motion and $y_{scc}(t)$ is the normalized firing rate. The numerical values for this model are, [Hosman78],

$$G_s = 233\, s^2/rad\,,\ \tau_1 = 5.9\, sec\,,\ \tau_2 = 0.003\, sec\,. \qquad (31)$$

[1] In fact we should have chosen $G_o = 5.32\, s^2/m$ which is consistent with Hosman's measured threshold of Sp_y of $0.47\, m/s^2$ at an angular frequency of approximately zero rad/s, not at $0.94\, rad/s$.

The value of G_s is selected so that one threshold unit of y_{scc} corresponds to an input ϕ of $1.45\, deg/s^2$ at an angular frequency of $0.94\, rad/s$.

Roll motion, i.e. rotatory motion along a horizontal axis is perceived by both the semicircular canals and the otolith because both angular acceleration and a change in specific force are generated. We proceed now by computing the inputs to the otoliths and to the semicircular canal under a combined lateral linear motion and roll motion. Let the y-z axes, which are attached to the person moved, be rotated with a roll angle $\phi(t)$ with respect to the axes Y-Z which are fixed in space (Figure 6). Let the person also have a linear motion d(t) along the Y axis, i.e. a lateral motion. The gravity is always directed up along -Z.

We proceed by making the following assumptions: (i) $\phi(t)$ is a small angle so that $\sin\phi(t)$ can be replaced by $\phi(t)$ and $\cos\phi(t)$ by 1, and (ii) the addition to the specific force in the z direction is small compared to one g, so that Sp_z will not be considered and only Sp_y is taken into account.

The simplified block diagram describing the vestibular system ($\mathcal{V}'$ of Subsection II.9.3, Figure II.13) is shown in Figure 5. The state equations of the vestibular system of the airplane pilot subjected to the actual motion are:

$$\mathcal{S}^a: \quad \begin{aligned} \dot{\mathbf{x}}^a(t) &= A^a\mathbf{x}^a(t) + B^a\mathbf{u}^a(t) && (32)\\ \mathbf{y}^a(t) &= C^a\mathbf{x}^a(t) + D^a\mathbf{u}^a(t), && (33) \end{aligned}$$

where the state vector $\mathbf{x}^a$ has the following three coordinates: x_1^a is the state of the otolith model; x_2^a and x_3^a are the states of the semicircular canal model; and where,

$$A^a = \begin{pmatrix} -b_o & 0 & 0 \\ 0 & -a_s & 1 \\ 0 & -b_s & 0 \end{pmatrix}, \quad B^a = \begin{pmatrix} G_o(a_o - b_o) & -G_o g(a_o - b_o) \\ 0 & -a_s b_s G_s \\ 0 & -b_s^2 G_s \end{pmatrix}, \tag{34}$$

$$C^a = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad D^a = \begin{pmatrix} G_o & -G_o g \\ 0 & G_s b_s \end{pmatrix}, \tag{35}$$

with,

$$a_s = \frac{\tau_1 + \tau_2}{\tau_1 \tau_2}, \quad b_s = \frac{1}{\tau_1 \tau_2}, \tag{36}$$

$$\mathbf{u}^a(t) = \begin{pmatrix} \ddot{d}^a(t) \\ \phi^a(t) \end{pmatrix}, \quad \mathbf{y}^a(t) = \begin{pmatrix} y_{oto}^a(t) \\ y_{scc}^a(t) \end{pmatrix}. \tag{37}$$

Also, $\ddot{d}^a(t)$ and $\phi^a(t)$ are the actual linear sway acceleration and the actual roll angle respectively, to which the pilot in the airplane is subjected, while $y^a_{\mathrm{oto}}(t)$ and $y^a_{\mathrm{scc}}(t)$ are the resulting normalized firing rates of the otolith and the semicircular canal respectively.

The system $\mathcal{S}^s$, neglecting the dynamics of the simulator, is the vestibular system of the pilot in the simulator and has equations similar to (32)–(37) with the superscript "a" replaced by "s". Furthermore, in the simulated system $\mathcal{S}^s$, we add two additional states, the displacement $d^s(t)$ and velocity $\dot{d}^s(t)$ of the simulator. These are added so that we can limit the simulator displacement and velocity. The overall state equations of the vestibular system of the simulator pilot subjected to the simulator motion augmented by the two states $d^s(t)$ and $\dot{d}^s(t)$ are thus

$$\mathcal{S}^s: \quad \begin{aligned} \dot{\mathbf{x}}^s(t) &= A^s\mathbf{x}^s(t) + B^s\mathbf{u}^s(t) && (38)\\ \mathbf{y}^s(t) &= C^s\mathbf{x}^s(t) + D^s\mathbf{u}^s(t) && (39)\\ \mathbf{u}^\ell(t) &= C^\ell\mathbf{x}^s(t) + D^\ell\mathbf{u}^s(t) && (40) \end{aligned}$$

where the state vector $\mathbf{x}^s$ has the following three coordinates: x^a_1 is the state of the otolith model; x^a_2 and x^a_3 are the states of the semicircular canal model and x^a_4 and x^a_5 are the displacement, $d^s(t)$, and the velocity, $\dot{d}^s(t)$ of the simulator, respectively, and where[2],

$$A^s = \begin{pmatrix} -b_o & 0 & 0 & 0 & 0 \\ 0 & -a_s & 1 & 0 & 0 \\ 0 & -b_o & 0 & 0 & 0 \\ 0 & 0 & 0 & 0* & 1 \\ 0 & 0 & 0 & 0 & 0* \end{pmatrix}, \quad B^s = \begin{pmatrix} G_o(a_o - b_o) & -G_og(a_o - b_o) \\ 0 & -a_sb_sG_s \\ 0 & -b_sG_s \\ 0 & 0 \\ 1 & 0 \end{pmatrix}, \qquad (41)$$

$$C^s = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}, \quad D^s = \begin{pmatrix} G_o & -G_og \\ 0 & G_sb_s \end{pmatrix}, \qquad (42)$$

with a_s, b_s are defined in (36) and

$$\mathbf{u}^s(t) = \begin{pmatrix} \ddot{d}^s(t) \\ \phi^s(t) \end{pmatrix}, \quad \mathbf{y}^s(t) = \begin{pmatrix} y^s_{\mathrm{oto}}(t) \\ y^s_{\mathrm{scc}}(t) \end{pmatrix}. \qquad (43)$$

We also add two additional output signals, with the output equation (III.23):

$$\mathbf{u}^\ell(t) = C^\ell\mathbf{x}^s(t) + D^\ell\mathbf{u}^s(t) \qquad (44)$$

[2] We have used a value of -10^{-5} for the $0*$ entries of A^s for numerical convenience.

$$\mathbf{u}^{\ell}(t) = \begin{pmatrix} \ddot{d}^s(t) \\ \phi^s(t) \\ d^s(t) \\ \dot{d}^s(t) \end{pmatrix}, \quad C^{\ell} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad D^{\ell} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \tag{45}$$

and $d^s(t)$ and $\dot{d}^s(t)$ are the sway linear displacement and velocity of the simulator. Incorporating these outputs, $\mathbf{u}^{\ell}(t)$, into the cost function is instrumental in placing bounds on the displacement and velocity of the simulator.

2.2. Derivation of the Optimal Washout Transfer Matrix

We shall assume that both $\ddot{d}^a(t)$ and $\phi^a(t)$ are first order stochastic processes, i.e.

$$\dot{\mathbf{u}}^a(t) = \begin{pmatrix} -\beta_1 & 0 \\ 0 & -\beta_2 \end{pmatrix} \mathbf{u}^a(t) + \begin{pmatrix} \beta_1 & 0 \\ 0 & \beta_2 \end{pmatrix} \mathbf{n}(t) \tag{46}$$

where $\mathbf{u}^a(t) = coluom(\ddot{d}^a(t)\,,\ \phi^a(t))$, $\mathbf{n}(t) = coluom(n_1(t)\,,\ n_2(t))$, and where $n_1(t)$ and $n_2(t)$ are independent white noise processes. As in Chapter III (III.50)–(III.51), we use the notation

$$A^n = \begin{pmatrix} -\beta_1 & 0 \\ 0 & -\beta_2 \end{pmatrix}, \quad B^n = \begin{pmatrix} \beta_1 & 0 \\ 0 & \beta_2 \end{pmatrix}, \quad C^n = I \tag{47}$$

The criterion J to be optimized is selected to be:

$$J \triangleq \mathfrak{E}\{\mathbf{e}^T(t)Q\mathbf{e}(t)\} + \rho \mathbf{u}^{\ell^T}(t)R\mathbf{u}^{\ell}(t) \tag{48}$$

where

$$Q = \begin{pmatrix} q_1 & 0 \\ 0 & q_2 \end{pmatrix}, \quad R = \begin{pmatrix} r_1 & 0 & 0 & 0 \\ 0 & r_2 & 0 & 0 \\ 0 & 0 & r_3 & 0 \\ 0 & 0 & 0 & r_4 \end{pmatrix} \tag{49}$$

and where we require that $Q \geq 0$, $r_1 > 0$, $r_2 > 0$, $\rho > 0$, and in order to weigh separately the errors and the limitations we select

$$q_1^2 + q_2^2 = 1\,, \quad \sum_i^4 r_i^2 = 1 \tag{50}$$

The augmented linear system $\tilde{S}$ and its optimization criterion are given in (III.53)–(III.62).

The detailed form of the washout filter matrix is obtained by substituting the values for A^s, B^s, C^s, D^s, C^ℓ, D^ℓ, A^a, B^a, C^a, D^a, A^n, B^n, C^n, Q, and R for the present problem as given in (34)–(36), (41)–(42), (44)–(45), (47)–(49) into ($\underline{\text{III}}$.53)–($\underline{\text{III}}$.64) and ($\underline{\text{III}}$.79) in Section $\underline{\text{III}}$.4. A standard computer program package for solving the Linear-Quadratic-Gaussian problem has been used to obtain the numerical results of the solution of the Riccati equation, the P matrix, and then to obtain the F matrix. The washout filter, W(s) is obtained by using ($\underline{\text{III}}$.79), and its elements will be denoted as follows:

$$W(s) = \begin{pmatrix} W_{1,1}(s) & W_{1,2}(s) \\ W_{2,1}(s) & W_{2,2}(s) \end{pmatrix} \tag{51}$$

The elements $W_{i,j}(s)$, $i, j = 1, 2$, turn out to be transfer functions of dimension 5, which corresponds to the dimensions of the matrix A^s, since A^a is a factor in A^s (same vestibular model, see subsection $\underline{\text{III}}$.5.3). For the following numerical values of the parameters,

$$\rho = 1\,,\ \beta_1 = 0.01\,rad/sec\,,\ \beta_2 = 0.025\,rad/sec\,, \tag{52}$$

$$q_1 = q_2 = 0.707\,,\ r_3 = 0.999\,,\ r_1 = r_2 = r_4 = 5.77 \times 10^{-3} \tag{53}$$

the optimal washout filter is computed to be as follows (open-loop implementation see subsection $\underline{\text{III}}$.7.1):

$$W_{1,1}(s) = 0.5617(s - 0.0118)(s + 10^{-5})^2(s + .0772)(s + 0.1588)/D(s) \tag{54}$$

$$W_{1,2}(s) = 0.03471(s - 0.0259)(s + 10^{-5})^2(s + 0.1006)(s + 0.1697)/D(s) \tag{55}$$

$$W_{2,1}(s) = -1.15\times10^{-4}(s+0.085)(s+0.1695)(s+333)(s^2+0.3673s+0.0709)/D(s) \tag{56}$$

$$W_{2,2}(s) = (s + 0.0737)(s + 0.1689)(s + 0.2641)(s^2 + 0.3312s + 0.1120)/D(s) \tag{57}$$

where

$$D(s) = (s + 0.0746)(s + 0.1689)(s + 0.2650)(s^2 + 0.3325s + 0.1123) \tag{58}$$

It can clearly be seen from (55) and (56) that $W_{1,2}(s)$ and $W_{2,1}(s)$ are different from zero, implying that to simulate a linear acceleration one should also use angular motions and vice versa. Namely, we are willing to deliberately introduce some semicircular canal errors in order to reduce the total otolith error and vice versa. The Bode plots of each of the components of $W(s)$ are presented in Figure 7.

For the following set of parameters in (53) we have plotted, in Figure 8, the locus of common *poles* of $W_{ij}(s)$, $i, j = 1, 2$ and, as can be seen, the poles essentially move to the left as ρ increases. These poles do not depend upon the value of β_1 and β_2, as noted in section $\underline{\text{III}}$.5. Using this last property and assuming the values of Q and R are "chosen properly", the pole location in our design can be compared to those found empirically through experimental studies [Sinacori77]. Increasing the value of ρ corresponds to decreasing the "fidelity of the motion" given to the pilot. From the simulations in Figures 14 and 15, it would appear that a value of $\rho = 1$ would correspond to reasonably high fidelity. Thus, from (54) and (58), $W_{1,1}(s)$ can be approximated by a third order system with a single pole at $0.26\, rad/sec$ and a complex pair with a natural frequency $\omega_n = 0.33\, rad/sec$ and damping $\varsigma = 1$. These can be loosely compared to empirical best settings for commonly used second order washout filters reported by Sinacori [Sinacori77], which has an $\omega_n = 0.33\, rad/sec$ and $\varsigma = 0.7$, which gives motion quality that is judged experimentally to be high fidelity motion.

The zeroes of $W_{1,1}(s)$, $W_{1,2}(s)$ and $W_{2,1}(s)$ are given in Figures 9–11. The zeroes of $W_{2,2}(s)$ have almost the same location as the poles, so that $W_{2,2}(s) \approx 1$. The double zeroes of $W_{1,1}(s)$ and $W_{1,2}(s)$ near the origin act as double differentiators so that the DC gain of the washout filter for constant velocities, $\dot{d}^a(t)$, and accelerations $\ddot{d}^a(t)$, is zero, thus the general behavior for $W_{1,1}(s)$ is one of a high pass filter, as used in other existing washouts [Sinacori77]. Note that the zeroes of $W_{2,1}(s)$ at $-0.17\, rad/sec$ and $-333\, rad/sec$ are the inverse of the two time constants of the semicircular canals.

Next, we present the initial gains, i.e. $W(j\infty)$, and the asymptotic gains, i.e. $W(0)$, for a step input as a function of ρ for three different values of β_1 and β_2 (Figures 12 and 13).

2.3. Simulations

We present the results of two schematic simulations. In the first simulation, we interpret the two degrees of freedom as surge linear motion with pitch angular motion. An airplane is accelerated forward at $2\,m/s^2$ for $10\,seconds$, and then flies with a constant speed for another $25\,seconds$. We use the same cost function parameters given in (53), with $\rho = 10$, $\beta_1 = 1\,rad/sec$ and $\beta_2 = 0.1\,rad/sec$. The results of the simulation are plotted in Figure 14.

The next simulation is a *coordinated turn*. Here we interpret the two degrees of freedom as sway linear motion and roll angular motion. By *coordinated turn*, we mean that the sway linear motion and the roll angle are so related that the acceleration vector is at all times along the z axis. The parameters for the design are the same as in (52)–(53). The sway motion for the simulation is as follows: $10\,seconds$ of a $0.5\,m/s^3$ rate of increase in acceleration (roll into the turn), then $50\,seconds$ of constant acceleration at $5\,m/s^2$ (turn time), and then another $10\,seconds$ of a $-0.5\,m/s^3$ rate of decrease in acceleration (roll out of the turn), over all it has a peak bank angle of $29\,degrees$, and a load factor of $1.15\,g$. We terminate the simulation with $30\,seconds$ of constant velocity at $300\,m/s$, the results are shown in Figure 15. The required sway travel can be achieved easily on the FSAA six degree of freedom flight simulator and is only a little bit too large for the VMS five degree of freedom flight simulator—both simulators are at NASA, Ames, in the U.S.A..

3. Conclusions

The examples presented in this chapter are preliminary investigation of the feasibility of the optimal simulator design approach. The results so far are promising. Furthermore, the "optimal" washout filters derived here have parameters of the same order of magnitude as the conventional filters in use today. On the other hand, the "optimal" washout filters can be "tuned" by a non-expert using our computer design method to satisfy a variety of additional conditions such as: different travel lengths of the simulator, different flight trajectories, and different emphasis on motion cues.

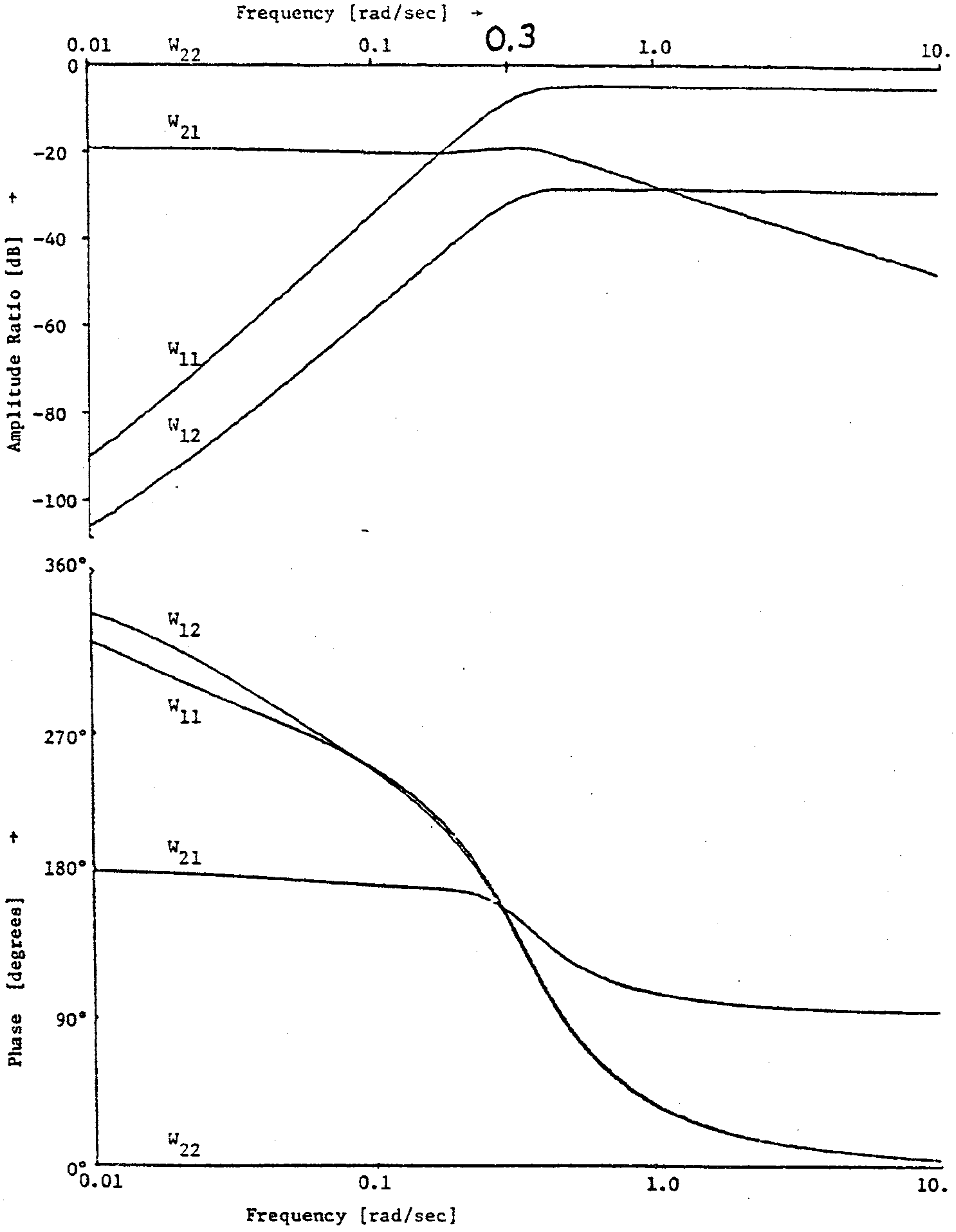


Figure 7. Bode plots for each component of the transfer matrix W(s) (design parameters given in (53)-(52)).

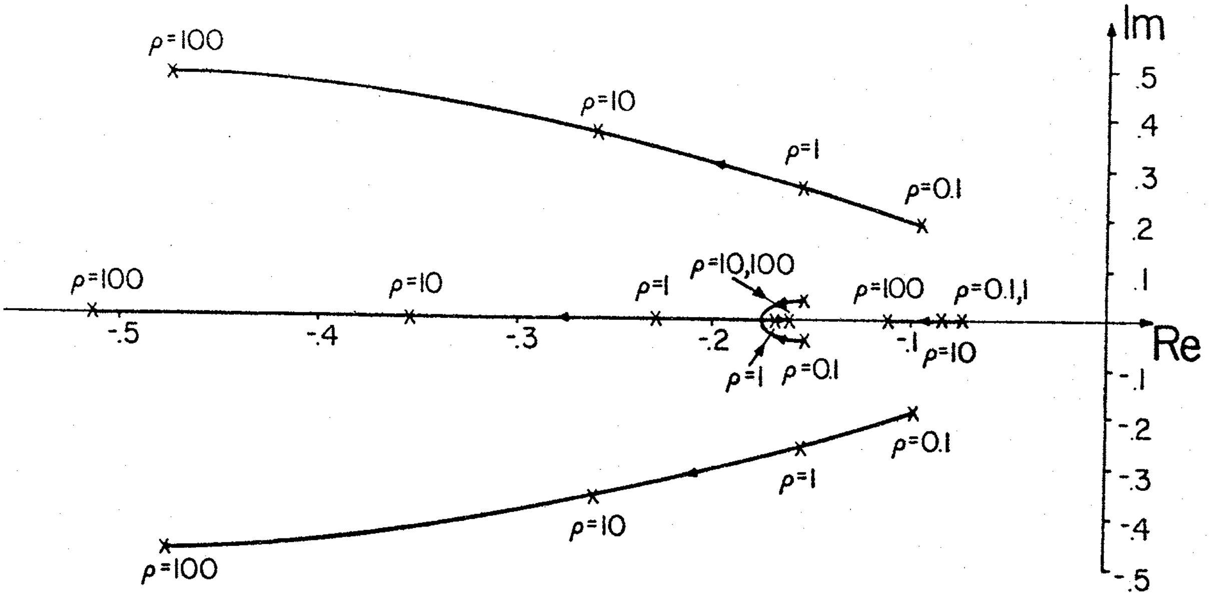


Figure 8. The locus of the common poles of $W_{i,j}(s)$, $i, j = 1, 2$, as a function of ρ.

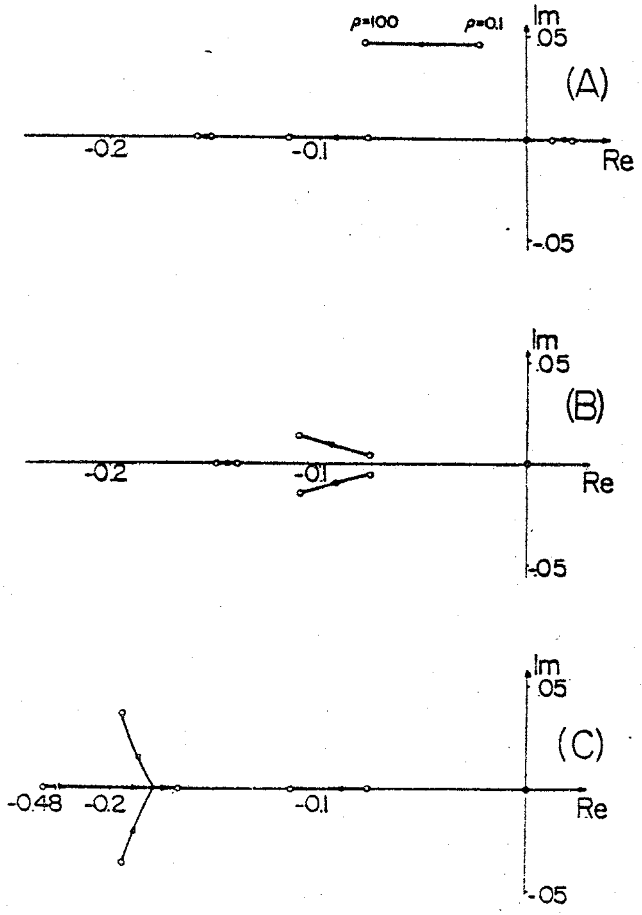


Figure 9. The locus of the zeros of $W_{1,1}(s)$ as a function of ρ for three values of β_1 and β_2 *rad/sec*, ρ varies between 0.1 and 100, using design parameters (53). (A) $\beta_1 = 10^{-5}$, $\beta_2 = 10^{-5}$ (B) $\beta_1 = 0.1$, $\beta_2 = 0.01$; (C) $\beta_1 = 1$, $\beta_2 = 0.1$.

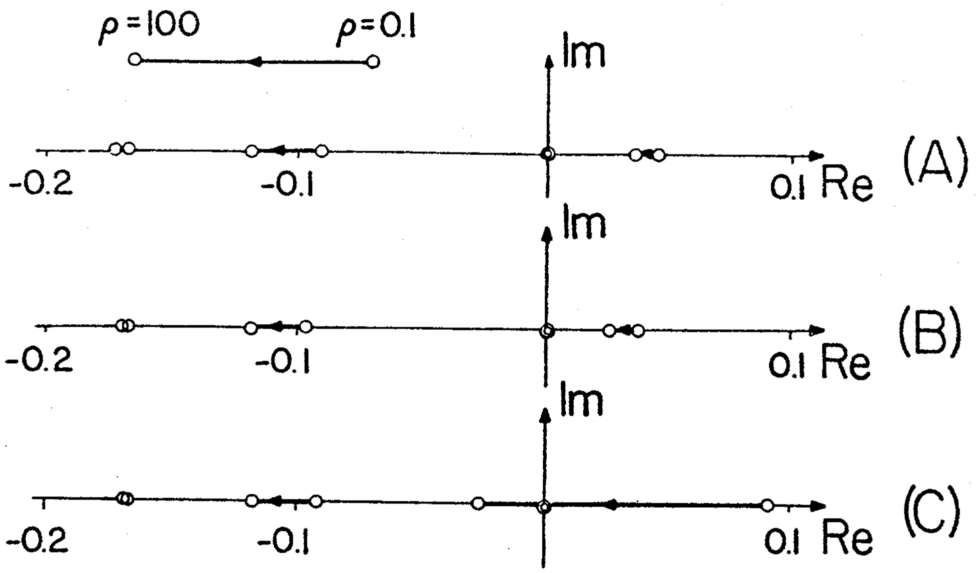


Figure 10. The locus of the zeros of $W_{1,2}(s)$ as a function of ρ for three values of β_1 and β_2 *rad/sec*, ρ varies between 0.1 and 100, using design parameters (53). (A) $\beta_1 = 10^{-5}$, $\beta_2 = 10^{-5}$ (B) $\beta_1 = 0.1$, $\beta_2 = 0.01$; (C) $\beta_1 = 1$, $\beta_2 = 0.1$.

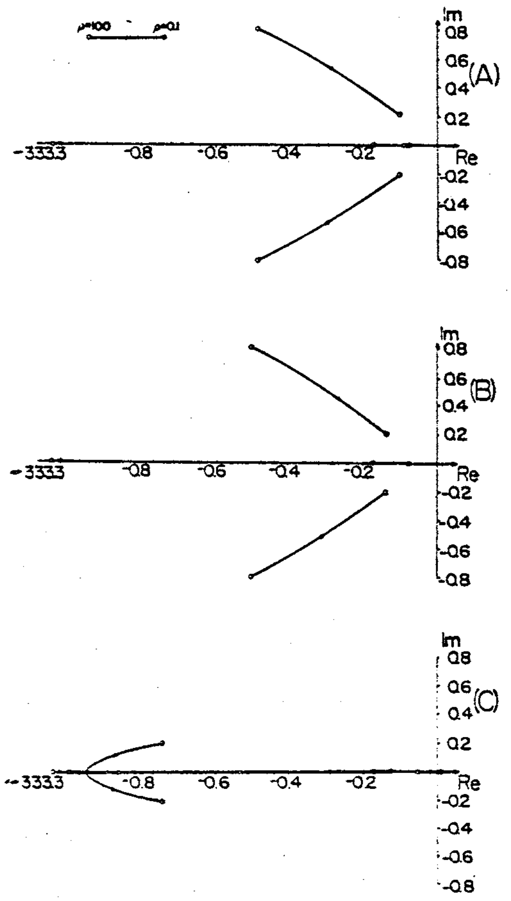


Figure 11. The locus of the zeros of $W_{2,1}(s)$ as a function of ρ for three values of β_1 and β_2 *rad/sec*, ρ varies between 0.1 and 100, using design parameters (53). (A) $\beta_1 = 10^{-5}$, $\beta_2 = 10^{-5}$ (B) $\beta_1 = 0.1$, $\beta_2 = 0.01$; (C) $\beta_1 = 1$, $\beta_2 = 0.1$.

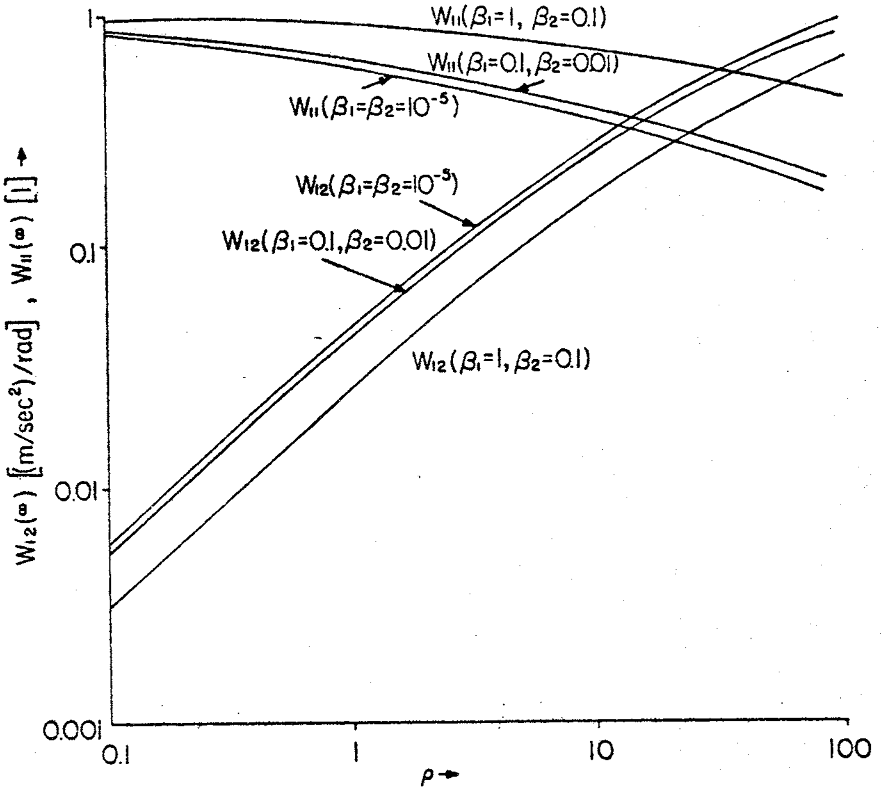


Figure 12. $W_{1,1}(j\infty)$ and $W_{1,2}(j\infty)$ as functions of ρ, and three values of β_1 and β_2 *rad/sec*. $W_{21}(j\infty) \approx 10^{-4}$ and $W_{2,2}(j\infty) \approx 1$ for all these values of ρ, β_1 and β_2.

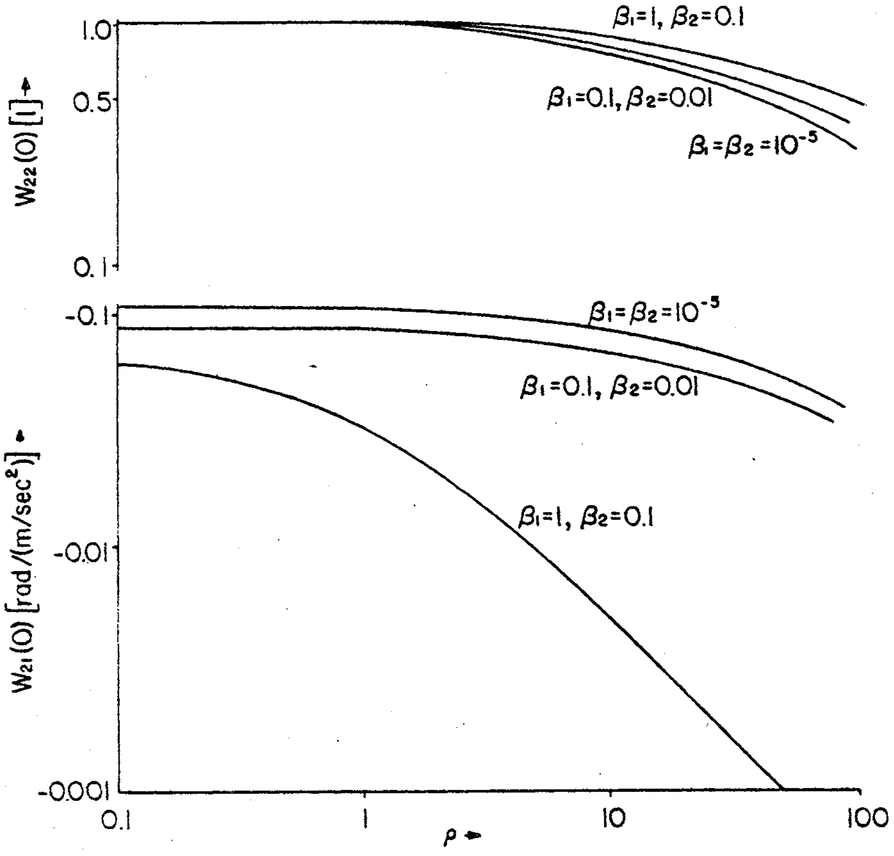


Figure 13. The DC gain $W_{2,1}(0)$ and $W_{2,2}(0)$ as functions of ρ, and three values of β_1 and β_2 *rad/sec*. $W_{1,1}(0)$ and $W_{1,2}(0) \approx 0$ for all these values of ρ, β_1 and β_2.

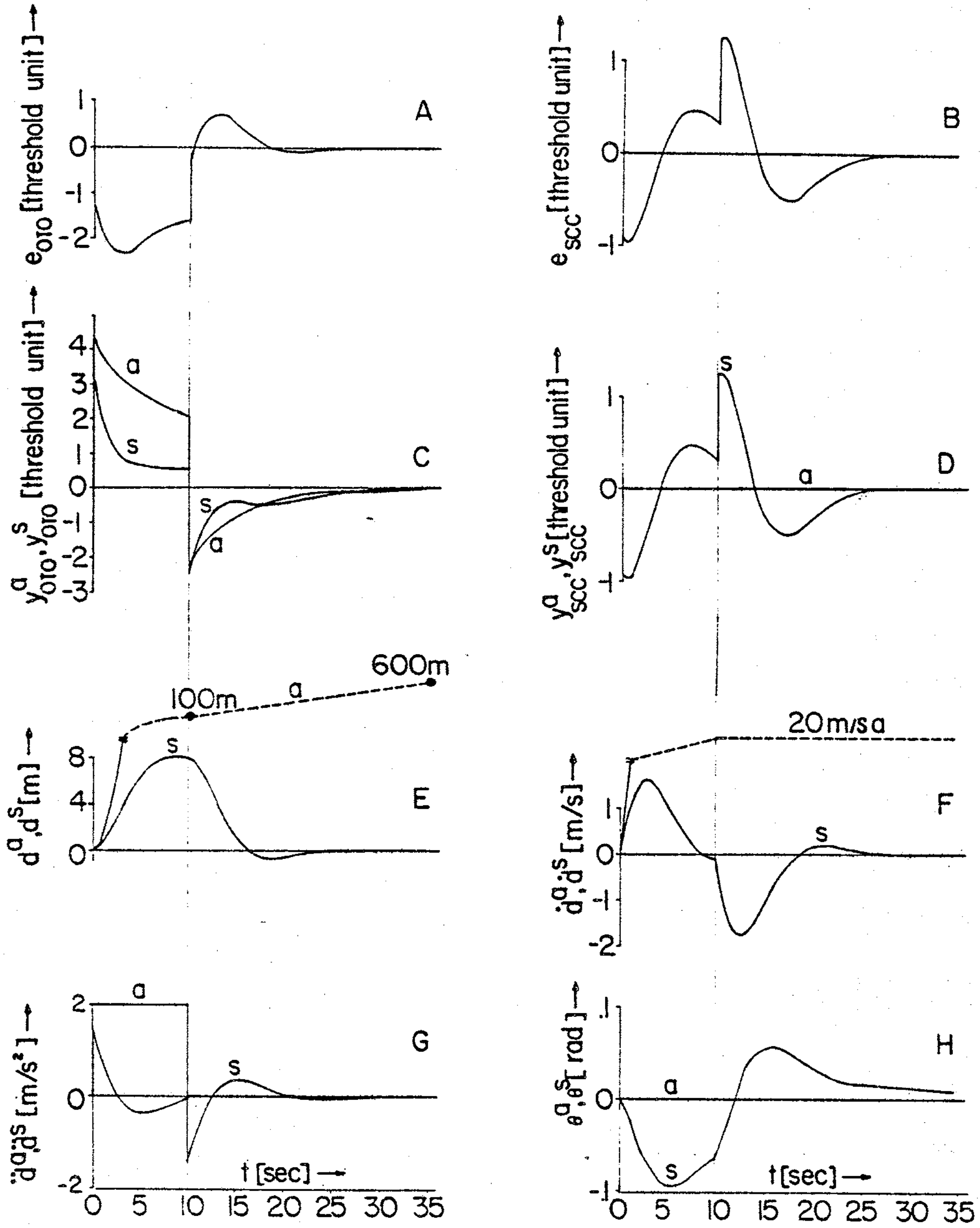


Figure 14. Numerical simulation for an actual airplane accelerated forward 2 m/sec^2 for 10 *sec*, which then flies at a constant speed for 25 *sec*. The superscripts "a" and "s" above the curves stand for the actual and the simulated systems. (Note the change of scale in E and F and that the pitch angle θ is defined in the opposite direction than usual).

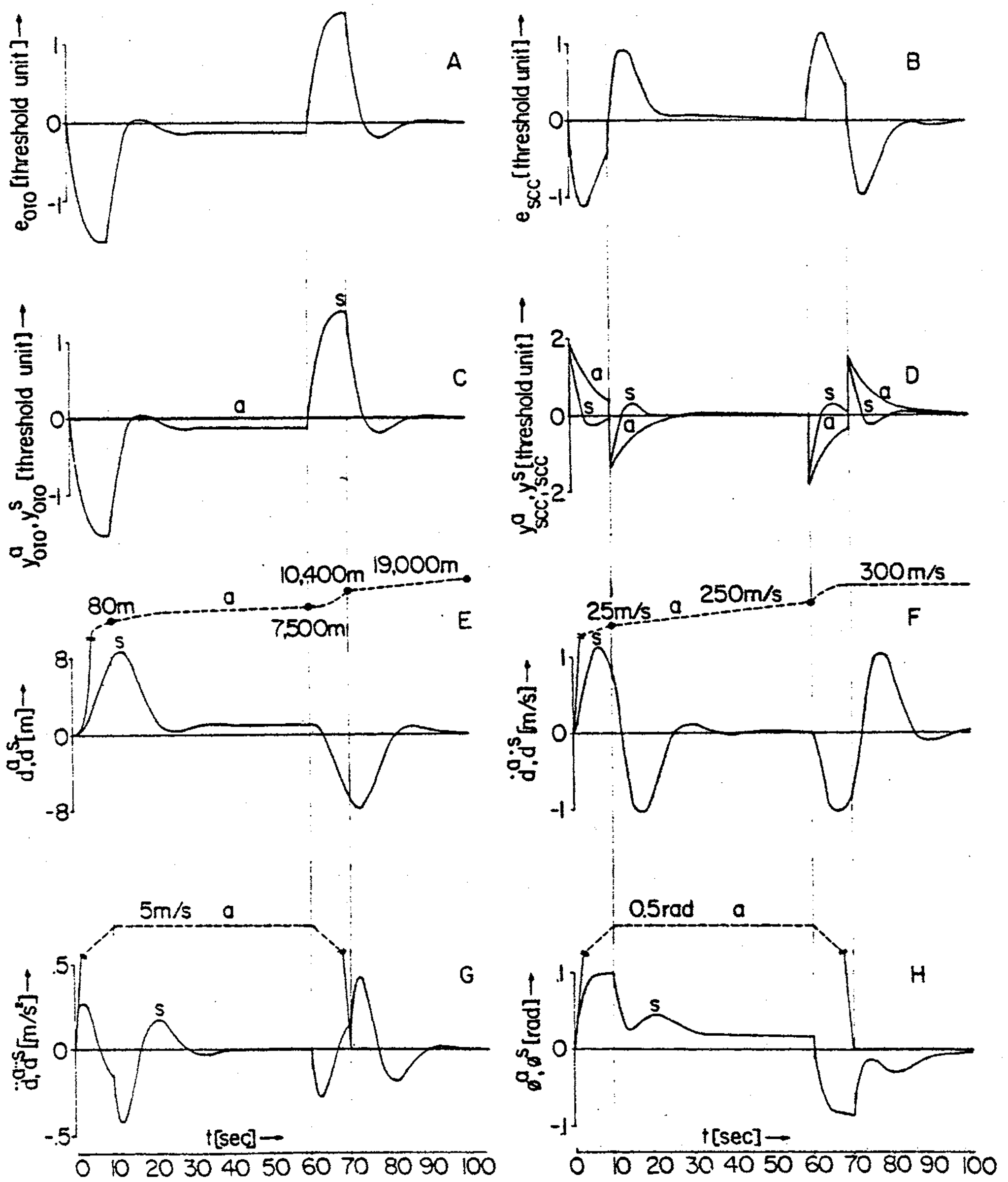


Figure 15. Numerical simulation of a coordinated turn. The superscripts "a" and "s" above the curves stand for the actual and the simulated systems, respectively. (Note the change of scale in E, F, G, H).

References

[Hosman78] Hosman, R.J.A.W. and van der Vaart, J.C., "Vestibular Models and Thresholds of Motion Perception. Results of Tests in a Flight Simulator", LR-265, Delft University Technical Department of Aerospace Engineering, Delft, The Netherlands, April, 1978.

[Sinacori77] Sinacori, J.B., "The Determination of Some Requirements for a Helicopter Flight Research Simulation Facility", Technical Report no. 1097-1, Systems Technology, Inc., Contract NAS2-9421 (CR-152066), NASA, AMES, pp. 2–12, Sep., 1977.

[Zacharias78] Zacharias, C.L., "Motion Cue Models for Pilot Vehicle Analysis," Report AMRL-TR-78-2, Bolt, Beranek and Newman, Inc., Cambridge, MA., May, 1978.

[Zarchan79] Zarchan, P., "Representation of Realistic Evasive Maneuvers by the Use of Shaping Filters," *Journal of Guidance and Control*, 2, no. 4, pp. 290–295, July, 1979.

Chapter VI

Washout System Implementation

Although we already have the OWS "solution" in Chapter III, there are still many considerations to be made before we look into the details of which computer and/or hardware to use. Beyond that we highlight here the solution properties developed in Chapter III which can aid in the OWS implementation. We start from general structure considerations and continue to more specific implementation limitations. The following topics are discussed:

1. OWS form—*open-loop*, *closed-loop*, or "both" (*model following*).
2. Merging the deterministic and stochastic solutions.
3. Axis transformations.
4. Head rotations.
5. Time-varying OWS.
6. Computation delay.
7. Sampling.
8. Implementation with a *sign sensitive cost.*
9. PLQ implementation.

We first consider only the OWS form for the stochastic solution. Merging of this solution with the deterministic one is discussed later.

In general, one would use as an input, besides u^a, all the states of the computed airplane, including the pilot control inputs, x^{na}, which constitute all or most of the required states $\mathrm{x}^n(t)$ of $\mathcal{N}$. The rest of the $\mathcal{N}$ states have to be estimated using a Kalman filter or some other estimation technique which is quite tedious and well avoided.

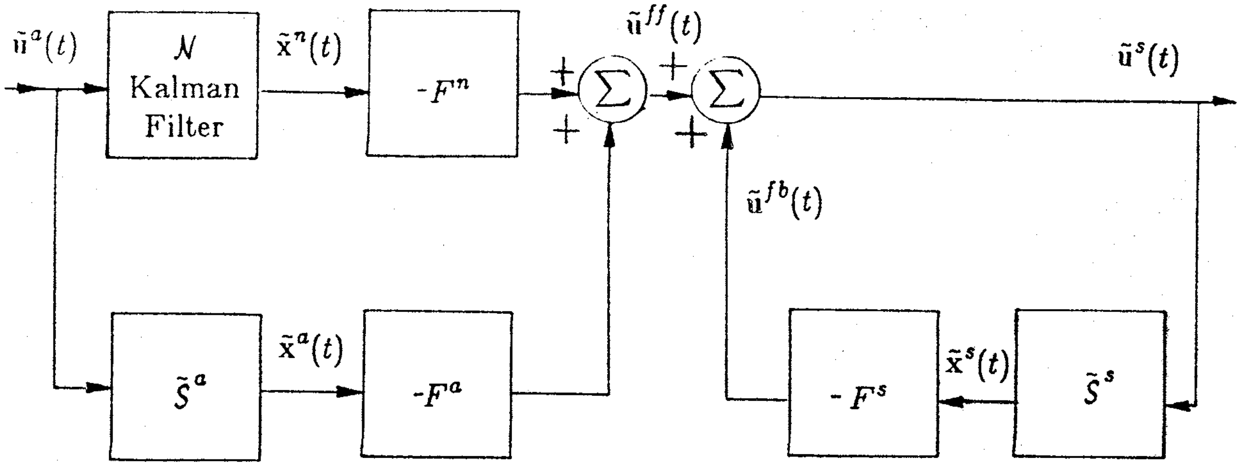


Figure 1. General block diagram of a CLOSED-LOOP OPTIMAL WASHOUT SYSTEM assuming zero mean input, i.e. $\bar{u}^c(t) \equiv 0$.

The open-loop OWS form is equivalent to an *Optimal Washout Filter* that shapes the motion-base input command and is the washout implementation commonly used. We recommend the closed-loop OWS implementation which has several advantages: (i) Solving the motion-base drift problem; (ii) Improving the linearity of the simulator motion system (smoother motion with less vibrations); (iii) Simpler to implement.

In Figure 4 we show an OWS for a system that includes both a *sign sensitive cost* and consideration of head rotations. This OWS requires the implementation of two vestibular models, one for the reference pilot and one for the simulator pilot and thus is more complex. In order to include PLQ control (Chapter IV), i.e. a nonquadratic cost, the constant optimal gains have to be replaced by functions of the limiting variables, which can be implemented efficiently by a small table lookup and interpolation.

In summary the OWS implementations described in this chapter have the following advantages over current washout filters: they make use of the future expected airplane motions, better account for hard limits by use of PLQ and take into account axis transformations and head movements. The closed-loop OWS is simpler to implement and as a free bonus gives the control system design for the motion-base itself.

1. Optimal Washout System Form of the Stochastic Solution

Here we basically discuss the possible implementations of equation (III.69) of Chapter III, which when reordered can be written as:

$$\tilde{\mathrm{u}}^s(t) = -F^a(t)\tilde{\mathrm{x}}^a(t) - F^n(t)\tilde{\mathrm{x}}^n(t) - F^s\tilde{\mathrm{x}}^s(t)\,. \tag{1}$$

There are two basic implementations of (1) plus a third combined one.

(i) An open-loop OWS, which is commonly called a WASHOUT FILTER.
(ii) A closed-loop OWS.
(iii) Model following OWS.

In Subsection III.4.2, an *open-loop* OWS was derived which results in the solution (III.73)-(III.77) or (III.79)-(III.80) under assumption (III.70),

$$\tilde{W}(s) = -\Big(I - F^a(sI - A^s + B^sF^s)^{-1}B^s\Big)\Big(F^a(sI - A^a)^{-1}B^a + F^n\mathcal{C}^{n-1}\Big) \tag{2}$$

where

$$\tilde{\mathrm{U}}^s(s) = \tilde{W}(s)\tilde{\mathrm{U}}^a(s)\,. \tag{3}$$

This solution is a command shaping filter, called usually a *washout filter* with input $\tilde{\mathrm{u}}^a(t)$ (or $\tilde{\mathrm{x}}^n(t)$) and output $\tilde{\mathrm{u}}^s(t)$. In this calculation, models for all the systems involved are used, including one for the physical existing motion-base, $\mathcal{M}$. The use of a model for the motion-base is the drawback of this OWS implementation form. First of all, this motion-base model is generally not accurate enough and usually cannot inherently be stably implemented due to several integrals i.e. poles at the origin. For example, position is always the integral of velocity, which causes drift in the OWS calculation and in the physical motion-base. Furthermore, the motion-base has some nonlinear "limiting logic" which are not included in the motion-base linear dynamic model. Second, implementing a motion-base model in the OWS increases substantially the computation required in real time, when these computations can be avoided by use of measurements of the existing, working, motion-base states. These two considerations lead to a new notion—a *closed-loop* OWS.

The implementation of a *closed-loop* OWS is shown in Figure 1. We decomposed the calculation of $\tilde{\mathrm{u}}^s(t)$ into two parts: feed-forward-- $\tilde{\mathrm{u}}^{ff}(t)$ and feedback— $\tilde{\mathrm{u}}^{fb}(t)$; which are defined as:

$$\tilde{\mathrm{u}}^{ff}(t) = -F^a\tilde{\mathrm{x}}^a(t) - F^n(t)\tilde{\mathrm{x}}^n(t)\,, \tag{4}$$

$$\tilde{\mathrm{u}}^{fb}(t) = -F^s\tilde{\mathrm{x}}^s(t) \tag{5}$$

and thus (1) can be written as:

$$\tilde{\mathrm{u}}^s = \tilde{\mathrm{u}}^{ff} + \tilde{\mathrm{u}}^{fb}\,. \tag{6}$$

The difference is in the closed-loop implementation of $\tilde{\mathrm{u}}^{fb}$, which uses the actual measured states[1]. The difficulty associated with the closed-loop implementation is that if the model for S^s, used in the design (to calculate the feedback gains F^s) was not sufficiently accurate or the sampling rate was too low, then the closed-loop system S^s may become unstable. On the other hand, there are some substantial advantages beyond the two mentioned before. A third advantage is that the closed-loop OWS has all the usual advantages of a closed-loop control system, which imply here the reduction of the motion-base rumble, drift and nonlinearities. A fourth advantage is that we have already designed the full motion-base control system to the actually required specifications and not beyond.

The third form—a *Model Following* OWS—is an *Open-Loop* OWS, used as the reference model, that is augmented with an error feedback to control a second open-loop OWS, which uses the actual motion-base to follow the reference model. This form of solution was suggested for flight simulator use by Sturgeon [Sturgeon81]. This form has the first advantage of the closed-loop OWS and to some extent the third one too, but is easier to design so that it is stable; which is the main disadvantage of the closed loop OWS. The main drawback of this form is that the ease in stable design resulted from losing most of the closed-loop design advantages. Another essential drawback is that the *model following* OWS requires the most computations and thus is the most complex to implement.

[1] May need estimates for some states which are not directly measurable.

Whenever possible, we recommend the use of the *closed-loop* OWS implementation form which offers the least complex implementation and all the benefits. Note that for stability of a *closed-loop* OWS only the sampling in the feedback path (of $\tilde{\mathbf{x}}^s(t)$) is important. Further discussion of the sampling problem is given in Section 6.

2. Merging the Deterministic and Stochastic Solutions

The merging of the *deterministic* and the *stochastic* solutions using the *open-loop* OWS implementation is shown in Chapter III Figure III.1 and in (III.27). There are two problems that occur when using an open-loop implementation:

(i) The nonlinear limiting logic cannot be considered (the stochastic $\tilde{\mathbf{u}}^s(t)$ is not known in advance).

(ii) The steady state (or approaching it as $t \to \infty$) states are not always finite as demonstrated in the following example.

Example

This example is taken from the implementation of the OWS on the Link GAT-1 flight simulator (Chapter VII). Let us augment the motion base states by the state I_{θ^s}, which is the integral of the simulator pitch angle, defined by:

$$I_{\theta^s}(t) \triangleq \int_0^t \theta^s(\tau)\, d\tau. \tag{7}$$

Let us further assume that the required steady state pitch angle found from the deterministic solution is nonzero, $\bar{\theta}^s \neq 0$. The corresponding deterministic state $\bar{I}_{\theta^s}(t)$ becomes infinite since:

$$\bar{I}_{\theta^s}(t) = \int_0^t \bar{\theta}^s\, d\tau = \bar{\theta}^s t \tag{8}$$

where we assumed for simplicity that $\bar{\theta}^s(t)$ is a constant. Thus we have a difficulty using an open-loop OWS. Furthermore, even in the closed-loop OWS implementation the stochastic state $\tilde{I}^s_\theta(t)$ has to be computed by:

$$\tilde{I}_{\theta^s}(t) = \int_0^t \theta^s(\tau) - \bar{\theta}^s\, d\tau \tag{9}$$

and not by the mathematically equivalent:

$$\tilde{I}_{\theta^s}(t) = I_{\theta^s}(t) - \bar{I}_{\theta^s}(t).$$

The "correct" implementation of this example is demonstrated in the OWS implementation on the GAT-1 (chapter VII). Note that the deterministic solution used in the GAT-1 implementation is:

$$\bar{\theta}^s(t) = k\theta^a(t) \tag{10}$$

where we choose $k = \frac{1}{2}$ since $\max(\theta^s) = \frac{1}{2}\max(\theta^a)$ and also $\min(\theta^s) = \frac{1}{2}\min(\theta^a)$.

Closed-loop implementation

The final result of combining the deterministic solution with the stochastic closed-loop implementation (Figure 1), is the addition of a deterministic command $\bar{\mathrm{u}}^{sc}$ (Figure 2). In the *closed-loop* OWS implementation we need to use (III.15) for $\mathrm{x}^a(t)$, and similar equations for $\mathrm{x}^s(t)$ and $\mathrm{x}^n(t)$:

$$\tilde{\mathrm{x}}^s(t) = \mathrm{x}^s(t) - \bar{\mathrm{x}}^s(t)\,, \tag{11}$$

$$\tilde{\mathrm{x}}^a(t) = \mathrm{x}^a(t) - \bar{\mathrm{x}}^a(t)\,, \tag{III.15}$$

$$\tilde{\mathrm{x}}^n(t) = \mathrm{x}^n(t) - \bar{\mathrm{x}}^n(t)\,. \tag{12}$$

Thus, in the closed-loop implementation we need to have beyond $\bar{\mathrm{u}}^a(t)$ the whole input state $\bar{\mathrm{x}}^n(t)$, and the deterministic solution has to provide the expected time functions of the states $\bar{\mathrm{x}}^s(t)$, $\bar{\mathrm{x}}^a(t)$ beyond $\bar{\mathrm{u}}^s(t)$. This state information can be combined with $\bar{\mathrm{u}}^s$ to give a single deterministic command $\bar{\mathrm{u}}^{sc}$:

$$\begin{aligned}\bar{\mathrm{u}}^{sc}(t) &\overset{\Delta}{=} \bar{\mathrm{u}}^s(t) - (-F(t)\bar{\mathrm{x}}(t)) \\ &= \bar{\mathrm{u}}^s(t) + F^n(t)\bar{\mathrm{x}}^n(t) + F^a\bar{\mathrm{x}}^a(t) + F^s\bar{\mathrm{x}}^s(t)\end{aligned} \tag{13}$$

where the F's are the stochastic solution gains and $\bar{\mathrm{x}}(t)$'s are the expected system's states. Now from (III.27), (13) and (1) the optimal command $\mathbf{u}^s(t)$ is given by:

$$\mathrm{u}^s(t) = \bar{\mathrm{u}}^{sc}(t) - F^n(t)\mathrm{x}^n(t) - F^a\mathrm{x}^a(t) - F^s\mathrm{x}^s(t) \tag{14}$$

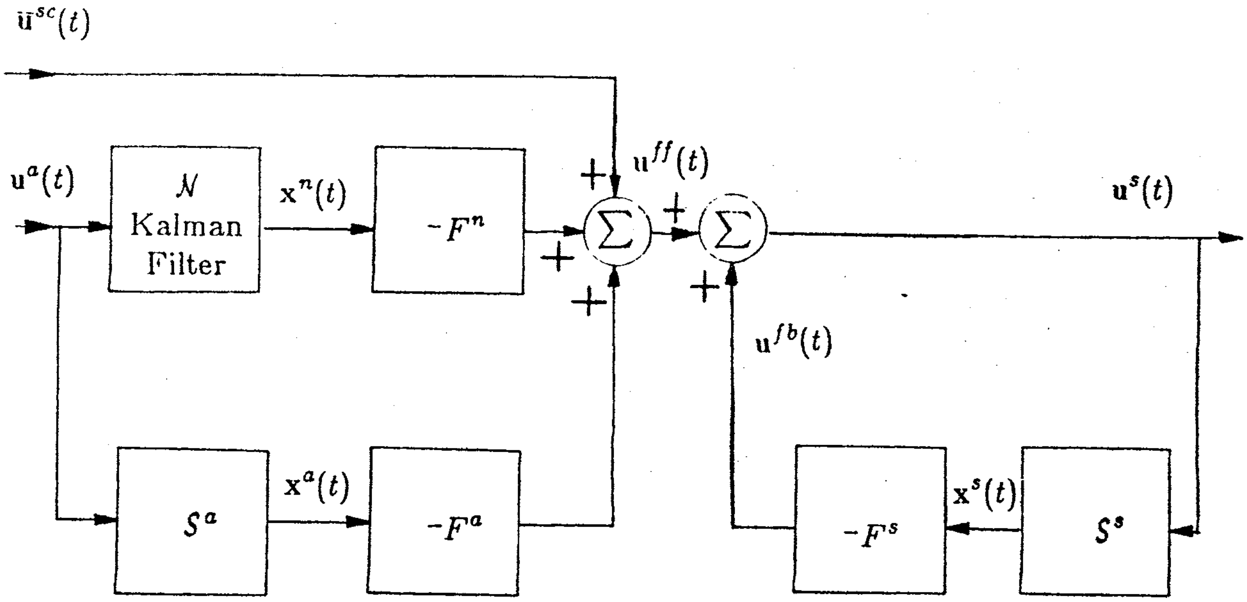


Figure 2. General block diagram of a CLOSED-LOOP OPTIMAL WASHOUT SYSTEM.

as seen in Figure 2.

Note that $\bar{\mathbf{x}}^{n}$ can be computed from an ensemble of $\mathbf{x}^{n}$ recorded from previous simulator flights, done for the same task. Furthermore, as the "simulator gains experience" the estimate of $\bar{\mathbf{x}}^{n}$ improves and so does the OWS.

Overall the closed-loop implementation is recommended. From here on the discussion refers to a closed-loop implementation unless otherwise stated.

3. Axis Transformation

So far we used the L.Q. approximations discussed in chapter II to obtain an OWS in the inertial axis system. This solution can be improved by reinserting the "proper" axis transformations as shown in Figure 3, called an OWSI (OWS Improved). In order to obtain the OWSI we first derive an alternative OWS implementation (Figure 3). This implementation has the intuitive "expected" structure an OWS "should have", which leads to the "proper" way to insert the axis transformations to obtain the OWSI based on their extraction done in section II.9. Next, several notes about the OWSI are given and an example of the improvement the OWSI offers is shown.

In the next section, pilot head movements are considered, which further improve the OWSI.

Recall from Figure II.6 that the system $\mathcal{S}^s$ is a cascade of the motion-base dynamics, $\mathcal{M}$, and the vestibular model, $\mathcal{V}^s$. Thus let us partition the states of $\mathcal{S}^s$ into two parts, one that corresponds to the motion-base states, x^m, and the other that corresponds to the simulator pilot vestibular model, x^{vs}:

$$\mathrm{x}^s = \begin{pmatrix} \mathrm{x}^m \\ \mathrm{x}^{vs} \end{pmatrix}. \tag{15}$$

Similarly, we partition the optimal gain matrix F^s:

$$F^s = (F^m, \quad F^{vs}). \tag{16}$$

Substituting (15), (16) into (14) and reordering the terms we obtain:

$$\mathbf{u}^s(t) = \bar{\mathrm{u}}^{sc}(t) - F^n(\tilde{t})\mathrm{x}^n(t) - F^{va}\mathrm{x}^{va}(t) - F^{vs}\mathrm{x}^{vs}(t) - F^m\mathrm{x}^m(t), \tag{17}$$

where we used the notation $F^{av} \equiv F^a$. The output equations for the two identical vestibular systems are given by:

$$\mathrm{y}^a(t) = C^v\mathrm{x}^{va}(t) + D^v\mathrm{u}^a(t) \tag{18}$$
$$\mathrm{y}^s(t) = C^v\mathrm{x}^{vs}(t) + D^v\mathrm{u}^{s\prime}(t) \tag{19}$$

where $\mathrm{u}^{s\prime}$ is the input to the simulator pilot's vestibular system that considers the flight simulator motion-base dynamics. Now let us augment the vestibular outputs y^a and y^s, symbolized by ${}'\mathrm{y}^a$ and ${}'\mathrm{y}^s$, so that ${}'C^v$ is full rank and ${}'C^{v-1}$ exists. Thus we can then multiply (18)-(19) by ${}'C^{v-1}$ to obtain:

$$\mathrm{x}^{va}(t) = {}'C^{v-1}\,{}'\mathrm{y}^a(t) - {}'C^{v-1}\,{}'D^v\mathrm{u}^a(t), \tag{20}$$
$$\mathrm{x}^{vs}(t) = {}'C^{v-1}\,{}'\mathrm{y}^s(t) - {}'C^{v-1}\,{}'D^v\mathrm{u}^{s\prime}(t). \tag{21}$$

Substituting (20)-(21) into (17), we obtain our alternative OWS implementation:

$$\mathrm{u}^s(t) = \bar{\mathrm{u}}^{sc}(t) - {}'F^n(t)\,\mathrm{x}^n(t) - {}'F^v\,{}'\mathrm{e}(t) - {}'F^m\,{}'\mathrm{x}^m(t) \tag{22}$$

where

$${}'\mathrm{e}(t) \triangleq {}'\mathrm{y}^a(t) - {}'\mathrm{y}^s(t), \quad F^v \equiv F^{va} = -F^{vs}, \qquad {}'F^v \triangleq F^v\,{}'C^{v-1}, \tag{23}$$

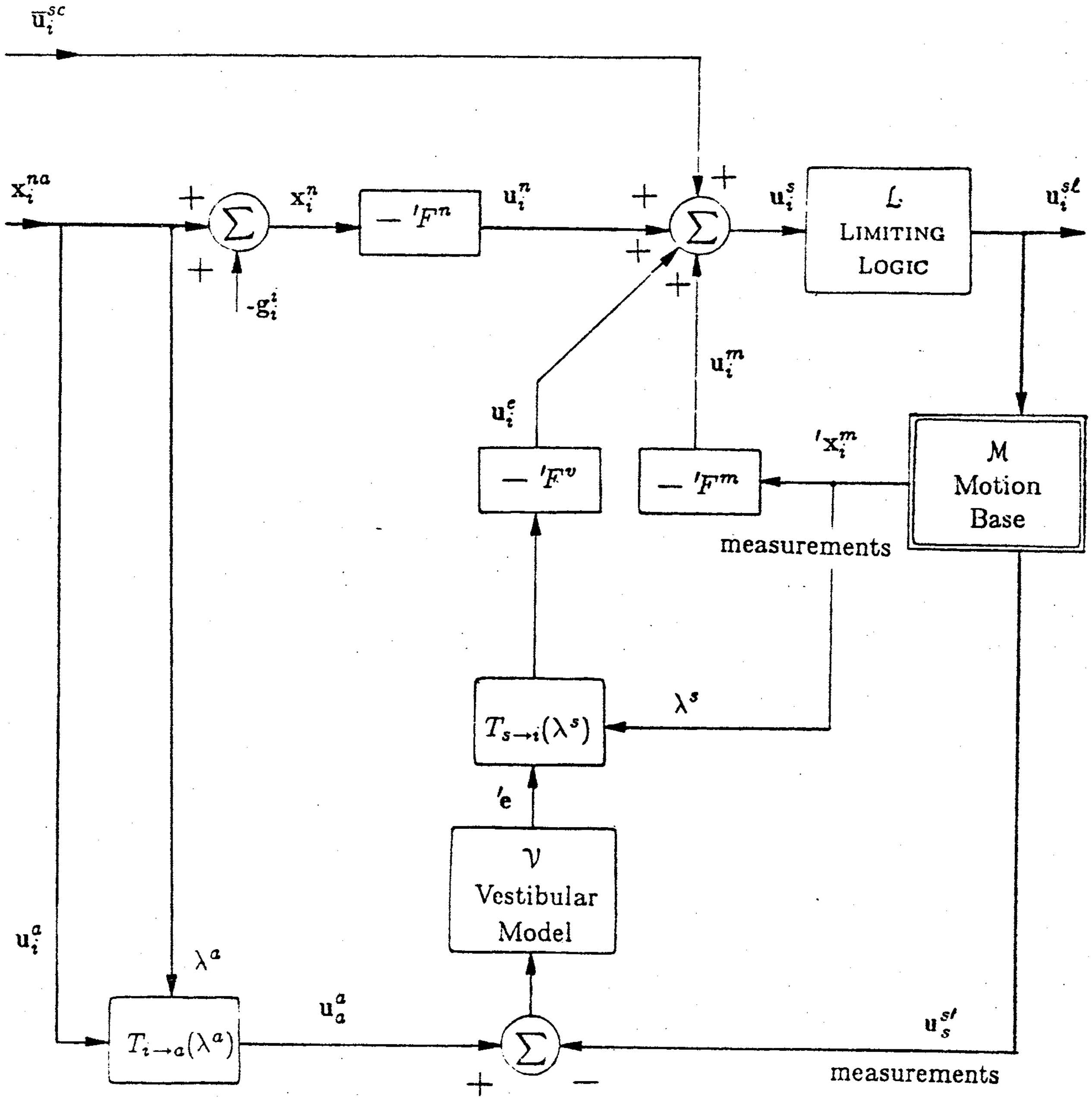


Figure 3. General Block Diagram of an IMPROVED CLOSED-LOOP OPTIMAL WASHOUT SYSTEM. Subscripts denote the axis system in which the vector is represented: i—initial, a—actual airplane, s—simulated airplane. $T_{i\to a}(\lambda^a)$ denotes the Euler axes transformation matrix from i to a axes according to Euler angles λ^a. $T_{s\to i}(\lambda^s)$ is defined similarly.

$$'\mathrm{x}^m(t) \triangleq \begin{pmatrix} \mathrm{x}^m(t) \\ \mathrm{u}^{s'}(t) \end{pmatrix}, \qquad 'F^m \triangleq \begin{pmatrix} F^m & 0 \\ 0 & 'F^v\,'D^v \end{pmatrix}, \tag{24}$$

$$\mathrm{u}^a(t) = K(t)\,\mathrm{x}^n(t), \qquad 'F^n(t) \triangleq F^n(t) - 'F^v\,'D^v K(t), \tag{25}$$

and where we further assumed that $\mathrm{u}^a(t)$ is a linear transformation of $\mathbf{x}^n(t)$; and used the symmetric case result ($\underline{\mathrm{III}}$.143) for F^v.

Intuitive view of the OWS

Let us interpret the four components that compose the optimal computed command $\mathbf{u}_i^s(t)$ (22):

$$\mathbf{u}_i^s = \bar{\mathrm{u}}_i^{sc} + \mathrm{u}_i^n + \mathrm{u}_i^e + \mathrm{u}_i^m \tag{26}$$

1. *Deterministic* command $\bar{\mathrm{u}}_i^{sc}$ (explained before).
2. *Input* $\mathrm{u}_i^n = -'F^n\,\mathrm{x}_i^n$.
   It is a weighted sum of the flight simulator computation inputs, $\mathbf{x}_i^n$.
3. *Error* $\mathrm{u}_i^e = -'F^v\,'\mathrm{e}$.
   It is a weighted sum of the augmented vestibular error, $'\mathrm{e}$. This component corresponds to a command that should "correct" the simulator motion to reduce the vestibular (model's) error. This is the component that is effected by the introduction of the axes transformations $T_{i\to a}(\lambda^a)$ and $T_{s\to i}(\lambda^s)$ (Figure 3). Thus the falsely called "Coriolis" interactions between two or more rotations are simulated "correctly" (according to the vestibular models used). The basic reason for this effect are the *"different initial conditions"* of the two vestibular models of the actual and simulator pilots. More detail in the example to come.
4. *Motion-Limitation* $\mathrm{u}_i^m = -'F^m\,'\mathrm{x}_i^m$.
   It is a weighted sum of the motion states, $'\mathrm{x}_i^m$, which introduce a "negative" feedback that forces the simulator cab to stay within the motion-base limitations (such as position, velocity and acceleration).

Notes on the OWSI (Figure 3)

(i) Only one model has been used for both vestibular systems (of the actual and simulator pilots) since the models are linear and only the models output difference, $'\mathrm{e}$, is needed. Nevertheless, it is as if each model's input is situated in its own axis system, namely, the actual pilot in the a-axes and the simulator pilot in the s-axes. Note that the transformation back into the inertial axes is identical for both vestibular outputs, **y**, which is unexpected (see section $\underline{\mathrm{II}}$.9 for why).

(ii) It is suggested that $\mathrm{u}^{s'}$ be measured directly in the s-axes by mounting three angular rate sensors and three linear accelerometers at a position which is as close as possible to the average simulator pilot's head position. This simplifies the computation (no axes transformation needed) and should improve the overall motion system, by measuring directly the physical variables of interest.

(iii) Similarly the motion-base state, $'\mathbf{x}_i^m$, is measured in the inertial axes.

(iv) The computed airplane motion input, $\mathbf{x}_i^n$, is given in inertial axes for simplicity (it is usually computed in airplane body and wind axes). It is assumed that all the states $\mathbf{x}_i^n$ are available and no state estimation is required.

(v) $\bar{\mathbf{u}}_i^{sc}(t)$ is the deterministic input command (13).

(vi) The vector $\mathbf{g}_i^i$ corresponds to one gee in the $-\mathbf{1}_z$ direction.

(vii) The software/hardware Limiting-Logic, $\mathcal{L}$, corresponds to the safety programming and hardware, which limits the motion to its safe operating region. Its input, $\mathbf{u}_i^s$, is the OWS computed command, and its output, $\mathbf{u}_i^{s\ell}$, is the actual command used to control the motion-base.

(viii) The closed-loop system feedback includes both $\mathbf{u}_i^m$ and $\mathbf{u}_i^e$. The closed-loop system may be unstable if one of them is disconnected.

(ix) The gain matrices F^m, F^v, $'F^m$, $'F^v$ are block diagonal and do not have cross terms between the four physical dimension groups:

1. *Longitudinal*: *Surge* linear and *Pitch* angular.
2. *Lateral*: *Sway* linear and *Roll* angular.
3. *Heave* linear.
4. *Yaw* angular.

which were discussed in Subsection III.5.2.1. This reduces significantly the number of additions and multiplications required to implement the OWS and the OWSI. The interaction between these physical dimension groups in the OWSI is through the axes transformations as demonstrated in the example that follows.

(x) The gain matrix $F^n(t)$ ($'F^n(t)$) is not necessarily block diagonal and the cross terms represent interactions between the four physical dimension groups due to the airplane aerodynamics (e.g. airplane yaw is caused by airplane roll).

(xi) The gain matrix $F^n(t)$ ($'F^n(t)$) can be time-varying in order to accomplish the following (see Section 5):

1. Better match of a time-varying or a nonlinear airplane dynamics computation model. Thus the states of $\mathbf{x}^n$ will approximately match those of the available airplane dynamic computation and no state estimation will be required.
2. Reduce the number of states in the system $\mathcal{N}$; which simplifies the design procedure and reduces the number of

multiplications and additions required in the implementation.

(xii) To accommodate expected changes in the airplane motion magnitude, the gain matrices F^m, F^v, $('F^m$, $'F^v)$ can be time-varying, as discussed in Section 5.

Example of OWSI improvement

This example discusses how we can obtain a "correct" simulation of the so falsely called "Coriolis effect" with less then full 360 degrees rotation capability on any of the rotation axes. Since such limited motion simulators cannot "naturally" simulate such a condition, special simulators, such as the "Vertigon", were built to demonstrate and train pilots for this vertigo effect.

Let us assume that the airplane has a constant yaw rate r^a. In the steady state the simulator motion-base will have zero yaw rate, $r^s = 0$, due to two facts: (i) the limited yaw rotation; (ii) in the steady state condition, both pilots' vestibular outputs y will be zero (current semicircular models are basically a high pass filter with a zero at the origin). Now let us add a step pitch input from 0 to 90 degrees, then the actual pilot would feel a roll motion at rate r^a (which is considered vertigo since the airplane did not have any roll motion) and the input pitch motion. On the other hand the simulator pilot would feel, without the OWSI, only pitch rotation; but with the OWSI the motion-base will **also roll** to give him the same roll sensation as in the real airplane. The OWSI adds the extra motion-base roll command due to the none-zero vestibular state which initially combined with the constant yaw rate gave a zero vestibular output; but after the pitch step input the vestibular state and the yaw rate are at right angles (90 degrees pitch) and do not cancel, which gives rise to a vestibular error in the simulator roll axis, which causes the motion-base to roll.

4. Head rotations

Head rotation is considered in a similar way as the axis transformations, just replace the vestibular model $\mathcal{V}$ in Figure 3 by a cascade of the transformation $T_{s\to h}(\lambda^h)$ from the cab axes to the pilot head axes, then the vestibular model $\mathcal{V}$

and finally, an axis transformation back to the cab axes $T_{h\to s}(\lambda^h)$ Figure 4 (Figure II.13). Adding these head axis transformations will cause realistic simulation of vertigo (the falsely called Coriolis effect) due to head movement, which requires a special simulator as mentioned before. Further analysis has to be done before actual implementation; since rapid head movements may cause very large motion-base commands. There are two solutions suggested:

(i) Scale down the measured head movements λ^h, that is to use $T_{h\to s}(\alpha\lambda^h)$ and $T_{h\to s}(\alpha\lambda^h)$ with $0 \geq \alpha \geq 1$ instead.

(ii) Limit the rate of head rotation, $\dot{\lambda}^h$, used for λ^h in the transformations by a nonlinear "slop limiting filter".

5. Time-Varying Optimal Washout System

We should distinguish between two sets of gains, feed-forward F^n and feedback F^m and F^v (F^s). The solution for the feed-forward gains is shown in Section III.6 and involves only a solution of a time-varying linear equation (III.157) which can be done in real time. It represents the dependence of the OWS on the stochastic modeling, $\mathcal{N}$ of the airplane motions, $\tilde{\mathbf{u}}^a$, and can improve the OWS by:

(i) A better match of a time-varying or a nonlinear airplane dynamics computation model. Thus the states $\mathbf{x}^n$ will approximately match those of the available airplane dynamic computation and no state estimation will be required.

(ii) Reduce the number of states in the system $\mathcal{N}$; which simplifies the design procedure and reduces the number of multiplications and additions required in the implementation.

On the other hand, the feedback gains solution depend on a gross measure of the expected airplane motions, through the cost parameters ρ and R (mainly ρ) and do not depend on $\mathcal{N}$ (Subsection III.5.1). Using time-varying parameters $\rho(t)$ and $R(t)$ enables use to take into account expected variations in the simulator required motion so as to optimize the us of the available motion-base. But the gains computation requires a solution of a differential Riccati equation that influences the stability of the closed-loop system. The basic reason for these after-thoughts is the incorrect ergodicity assumption used in the solution of the OWS. If we assume slow changes in $\rho(t)$ and $R(t)$, as compared to the eigenvalues of the systems $\mathcal{M}$, $\mathcal{V}$, then we can use a succession of time-invariant solutions as a good approximation to the time-varying solution (assuming $\dot{P}^s \approx 0$). Furthermore, we can use the steady state gains F^m and F^v, which are solutions of an ARE only, and are originally time-invariant. Thus the time variation of $F^m(t)$ and $F^v(t)$ can be implemented as an interpolation between a relatively small numbers of pre-computed gains, making a real time implementation of a time-varying OWS possible. This method was implemented in the OWS built for the GAT-1 flight simulator (chapter VII), where it was sufficient to interpolate between four gain matrices for successively larger ρ values. I Chapter IV, we extended this notion to include dependence of ρ and R on the simulator limiting variables $\mathbf{u}^\ell$, this new control design is termed PLQ.

6. Computation Delay

In many systems the aerodynamics computation is done on one computer and the "motion-base drive logic"—the washout system is calculated on another computer, based on the results of the previous aerodynamic calculations. Thus we are dealing with a delay of two computation cycles which can be as long as 0.14 seconds (the shortest 0.02 seconds). Thus, this delay cannot be overlooked. The solution is to use the current pilot input controls as part of the airplane input states $\mathbf{x}^n(t)$ which can compensate for the computation lag of the other airplane states. One thing that helps the situation is that even the short-period mode is of the order of one second so that the airplane states computation update should not cause too much of a problem if the computation cycle is less than 0.1 seconds.

7. Sampling

When using the closed-loop OWS, it is very important to have a short computation cycle time for the feedback loop. The feedback loop includes the calculation of (Figure 3): $\mathbf{u}_i^e$, $\mathbf{u}_i^m$ (26) and the limiting-logic, $\mathcal{L}$. If the implementation of Figure 2 is used (state feedback), then only the $\mathbf{u}^m$ part of $\mathbf{u}^{fb}(t)$ and $\mathcal{L}$ have to be computed within a short time (the latter since the vestibular system states have very slow eigenvalues, time constants of the order of several seconds). In the case where the vestibular models include fast modes of less than 1 second, their feedback contribution has to be calculated within a short time.

One way to reduce the sampling problem is to discretize the systems in the problem formulation stage. This does not solve the sampling problem since if the sampling rate is too low, then high frequency modes of the motion-base will not be sampled at a high enough frequency and the closed-loop OWS will be unstable.

8. Implementation With a Sign Sensitive Cost

Using the *sign sensitive cost* developed in Chapter IV Section IV.3 we obtain different gains values for F^{va} and F^{vs} (17) thus we need to use two vestibular models in the implementation as to compared with one before (Figure 3). The new implementation including the axis transformations and head movement consideration is shown in Figure 4.

9. Pseudo Linear Quadratic System Implementation

Using the derivation of *Pseudo Linear Quadratic* control (PLQ) in Chapter IV we can design an OWS using a nonquadratic cost function, such that the hard boundaries of the simulator travels can be better accounted for. This results in a nonlinear OWS where the resulting gains are functions of the limiting variables $\mathbf{u}^{\ell}(t)$, i.e. $F^{s}(\mathbf{u}^{\ell})$, $F^{a}(\mathbf{u}^{\ell})$ and $F^{n}(\mathbf{u}^{\ell})$. Thus in the PLQ implementation of Figure 4 the following optimal gains are **not constant** but functions of $\mathbf{u}^{\ell}(t)$, $'F^{n}(\mathbf{u}^{\ell})$, $'F^{va}(\mathbf{u}^{\ell})$, $'F^{vs}(\mathbf{u}^{\ell})$ and $'F^{m}(\mathbf{u}^{\ell})$. Furthermore, we cannot combine the deterministic state $\overline{\mathbf{x}}(t)$ and the deterministic solution $\overline{\mathbf{u}}^{s}(t)$ using (13) to obtain a lower dimensional deterministic input since the gains are not constant. Thus we have to store both $\overline{\mathbf{u}}^{s}(t)$ and $\overline{\mathbf{x}}(t)$ and use those as inputs instead of $\overline{\mathbf{u}}^{sc}(t)$.

Currently the calculation in real time of the optimal feedback gains $'F^{vs}(\mathbf{u}^{\ell})$ and $'F^{m}(\mathbf{u}^{\ell})$, is a considerable challenge (if possible at all) and requires the solution of an ARE in real time. Furthermore, even the calculation of the optimal feed-forward gains $'F^{n}(\mathbf{u}^{\ell})$ and $'F^{va}(\mathbf{u}^{\ell})$, is not easy and requires the solution of several Sylvester equations. Thus a table lookup and interpolation method is suggested. In Chapter IV an example is shown (Subsection IV.2.3) were we used a table with four entries for $F(\cdot)$ and interpolation between them, which worked very well. Being more conservative, we would need for a six-degree-of-freedom simulator a table of 15,120 numbers, assuming the matrix $F(\cdot)$ has 252 elements (6 controls times 42 states,

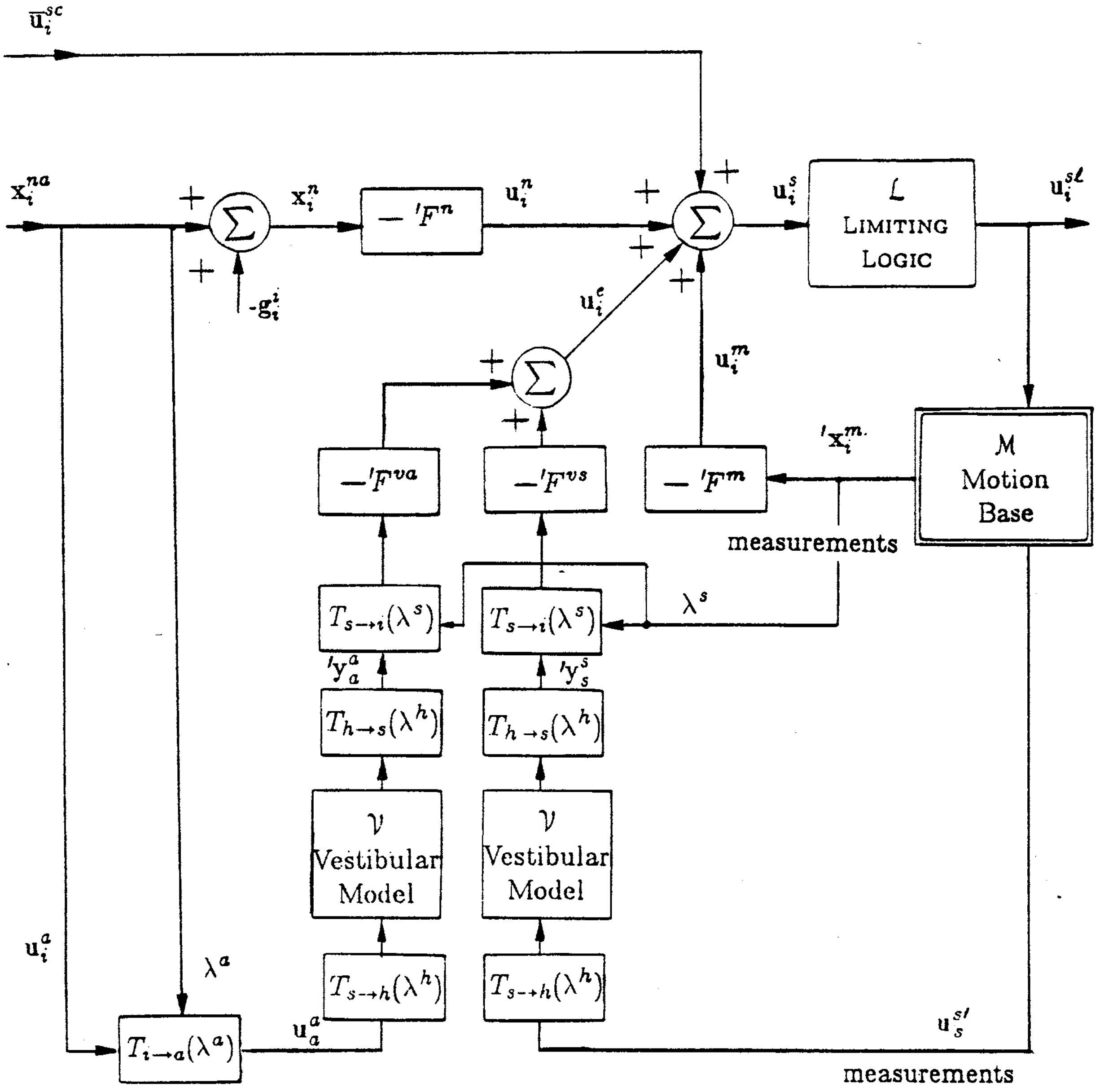


Figure 4. General Block Diagram of an IMPROVED CLOSED-LOOP OPTIMAL WASHOUT SYSTEM which includes consideration of pilot's head motions (λ^h). Subscripts denote the axis system in which the vector is represented: i—initial, a—actual airplane, s—simulated airplane, h—head. $T_{i\to a}(\lambda^a)$ denotes the Euler axes transformation matrix from **i** to **a** axes according to Euler angles λ^a. $T_{s\to i}(\lambda^s)$, $T_{s\to h}(\lambda^h)$, $T_{h\to s}(\lambda^h)$ are defined similarly. Note that since $T_{a\to h}(\lambda^h) \equiv T_{s\to h}(\lambda^h)$ and $T_{h\to a}(\lambda^h) \equiv T_{h\to s}(\lambda^h)$ then the symbols $T_{a\to h}(\lambda^h)$ and $T_{h\to a}(\lambda^h)$ where not used.

Section III.5) and we use 10 interpolating points for each of the 6 limiting variables in $\mathbf{u}^{\ell}$. This size of table is very reasonable to implement using even a small microcomputer.

References

[Sturgeon81] Sturgeon, W. R., "Controllers for Aircraft Motion Simulators," *Journal of Guidance and Control*, 4, no. 2, pp. 184-191, April, 1981.

References

[Sturgeon81] Sturgeon, W. R., "Controllers for Aircraft Motion Simulators," *Journal of Guidance and Control*, 4, no. 2, pp. 184–191, April, 1981.

Chapter VII

Experimental Evaluation

This chapter describes the implementation and testing of the optimal washout design methodology for the longitudinal mode (pitch and surge axes) for the LINK GAT-1 flight simulator. Twenty pilots were used as subjects in these experiments. Their experience ranged from non-pilots with no experience on a flight simulator to expert simulator pilots to student airplane pilots, to light airplane pilots with one hundred hours to those with a few thousand hours with and without instrument flight ratings, to fighter pilots with little experience with light airplanes but with a few thousand hours, to airline pilots with over ten thousand flight hours on large passenger jet airplanes such as the DC8. In spite of the large variety, there were no test pilots. The experiments performed are of two categories: a preliminary one which led to a quantitative one.

This chapter is organized as follows. In the introduction, several types of possible experiments and experimental conditions are presented and discussed. In the materials and methods section, the LINK GAT-1 flight simulator and the designed washout are first described, second, four experiments are outlined: (i) a blind test of a change of washout (ii) detection of a washout change (iii) detection of a random experimenter controlled flaps down during level flight (iv) the effect of washout change during take-off. The results section mainly describes the "flaps down" experiment. The discussion section points out the main conclusions from the experiment and their relation to the rest of the thesis.

This chapter includes the following appendices:

A. Suggested experiments on the VMS flight simulator at NASA AMES Research

Center.

B. Washout design program.

C. Operating instructions for the implemented GAT-1 pitch optimal washout system.

D. Pitch motion-base modeling.

E. Circuit testing of the pitch optimal washout.

F. GAT-1 modifications to adapt it for use with the pitch optimal washout system.

G. Design parameters and simulations of the optimal washouts #0 and #2.

H. Parameters of the OWS used in the experiments.

I. Flaps down detection, experimental results of T_d, T_c, T_p and Δh.

J. Experience of pilots used.

K. Experiments to demonstrate the Optimal washout system (OWS). Used in the M.I.T. *Flight Simulation* course 16.36, spring, 1982.

1. Introduction

The washout evaluation considered here is based on the expected value of motion in research simulators rather than in training simulators. The following objectives are used as the guidelines in the design of the washout experimental evaluation:

1. Are the changes in our design parameter of the pitch washout system that was implemented in the GAT-1 noticeable by the pilot and/or by the recorded simulator signals? How noticeable are these changes? In other words, how far apart are these different washout systems?
2. Which washout is best? In what sense is it best?
3. Can these results be extrapolated to the design of washout systems for other simulators and/or for other maneuvers not tested?

In order to address these objectives, the following four experimental approaches were considered, each of which relates to one or more of these objectives:

1. Pilot's performance and control.

2. Pilot's opinion.
3. Detection of washout changes.
4. Detection of airplane changes.

Next, each of these divisions and further subdivisions are discussed.

1.1. Pilot's Performance and Control

Looking for changes in task performance due to washout changes is the first thing that comes to mind. The problem is that well trained pilots can adjust to the changing environment (motion changes), and maintain the same level of performance almost independently of the motion [Parrish76], for tasks that involve "standard, state airplanes", as is also demonstrated in the experiments performed.

However, the pilot's control technique required to obtain the same performance can vary significantly [Puig78]. Thus, the combination of these two types of criteria can be used to evaluate the simulator's motion. The way to actually use these criteria and combine them is not simple, since the pilot's control varies considerably even under the same external conditions. One analysis method, used by [Sinacori77S] was to look at the resulting power spectra and probability density distribution of the stick position, but even these do not render a simple measure.

1.2. Pilot Opinion

The most common evaluation approach is to ask the pilot's opinion or rating of the simulation. This can be done for a specific set of maneuvers or for the pilot's choice of maneuvers. With proper training, these reports can give consistent ratings that are highly sensitive to the airplane environment [Puig78]. The pilot's opinion can be obtained in two ways:

(i) Absolute.
(ii) Relative—comparison of pairs of washouts.

Pilot's ratings obtained by the relative approach seem to be more consistent and sensitive. The pilot's opinion can be based on a reference to:

(i) Experience in the real airplane.
(ii) The correspondence of the simulator's motion to other displays: instruments, sound effects and the out of the window visual displays.

The second choice was recommended as the only valid choice by Bray and was used experimentally by Sinacori [Sinacori77S]. Furthermore, the pilot's opinion can be questioned in the following ways:

(i) Which washout gives the motion that corresponds best to one of the above references?

(ii) In which case does the simulator feel more like the real airplane (not referring to the simulator motion specifically)?

From preliminary exploratory experiments, it seems that the group of *light airplane pilots* tested (four pilots with more than one hundred hours of flight experience) tended to include the simulator's motion fidelity as part of the simulated airplane characteristics. Beyond that motion sensations and self orientation are usually non-cognitive sensations and thus are hard to report. For these reasons, it would be expected that the pilot's opinion will be difficult to use as a criterion for experiments using a small three-degree-of-freedom simulator flown by light airplane pilots with at most a few hundred flight hours. (A requirement of Sinacori's study [Sinacori70] is that pilots have recent experience and we would further require that they have at least 500 hours of flight experience.) Nevertheless, pilot opinion was checked and gave some unexpected results.

The other methods to be discussed will hopefully give more consistent and objective data. In the case where a larger flight simulator (more motion and degrees of freedom) is used, it is expected that the pilot's opinion will be a more valid measure. An experimental design using relative comparison between pairs of washout systems was designed for use on the VMS flight simulator at NASA Ames. This experiment was planned to evaluate the minimum required linear travels and is given in Appendix A.

1.3. Detect Washout Changes

In this experiment, we tested whether the pilot can "detect" one of the following:

(i) When and if a washout was changed?

(ii) Which washout was used (obviously after some learning period for the characteristics of the washout in question)?

The objective of these experiments is to test whether the pilot notices changes in the washout system and if he can recognize them. The success of such an experiment is a prerequisite for the evaluation of how good the simulator motion is for a given washout system. These experiments can also establish "just noticeable differences (JNDs)" between different washout systems, which are interesting results by themselves and can be used to improve the design criteria (cost) to obtain an optimal washout system.

This detection test can be performed for

1. Preselected maneuvers.
2. General maneuvers of the pilot's choice.

The preselected maneuvers have the advantage of being more reproducible and thus give more consistent results; but they may completely miss a whole domain that was not anticipated by the experimenter. Furthermore, they may take more time since the maneuver's protocol does not depend on the current experimental results. The relation between these two types of experiments is like the relation between "preset" and "adaptive" experiments to test digital finite-state machines [Kohavi70]. In most cases, preset maneuvers are used to intensively probe a narrow flight domain (that is considered important) and general maneuvers are used to obtain a general view of the overall quality of the washout system [Harrington79].

The final step in this type of experiment is how to judge whether the pilot detected the change in the washout. This can be done by one or more of the following:

1. Ask the pilot.
2. Detect any noticeable changes in the pilot's control and performance.
3. Physiological changes—eye movements, head movements, respiration, etc.

Asking the pilot is so simple, why bother with the others? The reason is that this method does not always work. In one preliminary experiment, the "pilot's control" always corresponded perfectly to the washout system used, but his reports (two choices) corresponded only to chance detection.

Concerning the physiological changes, very consistent head movements were found to correspond to airplane roll as reported by Sinacori [Sinacori77M]. These could possibly be used for objective detection of the washout's influence in the pilot. Other possible methods are eye movements, respiration, heart rate, galvanic skin resistance, pallor (a motion sickness symptom), urine test (to check changes in stress), and possibly others. These can give interesting results, but require a large amount of instrumentation connected to the pilot, which is not generally available. Beyond that, interpretation of these physiological results is not very easy. After establishing that the washout changes introduced are noticeable and or detectable, a further examination can be made to test how good a given washout is.

The next section describes another type of detection experiment to further evaluate the washout.

1.4. Detect Airplane Changes

In these experiments, we test whether the pilot can "detect" changes in the airplane simulation and displays. The objective here is to determine the washout that best simulates the pilot's actual performance in a real airplane. Possible changes in the airplane are:

1. Aerodynamics (equations of motion).
2. Flight instrument malfunction.
3. Other display changes (visual, for instance).
4. External disturbances.

The aerodynamics can be changed in a "natural" or "unnatural" way. Examples of natural aerodynamic changes are flaps up/down, landing gear up/down, or, a malfunction of the airplane such as engine power down. Less natural ones are motion of the center of gravity of the airplane or weight changes. Unnatural ones can involve any other changes in the parameters of the aerodynamical model used in the flight simulation including changes in the effects of the pilot's controls.

An example of flight instrument malfunction would be if the artificial horizon became stuck, which occasionally happens in the real airplane. The set of flight instruments is one type of display, other displays include the visual out-the-window display which can be changed to have different characteristics that could be detected with the aid of the simulator motion (a display in its own right). Examples of external disturbances are wind changes, external changes, or, changes in turbulence characteristics or intensity.

Overall, there are many possible experiments and only the ones that best fit the available simulator equipment and pilots will be pursued. In general, it seems best to use naturally occurring airplane changes which could be compared to measurements and real situations in the actual airplane.

2. Materials and Methods

In this section we first describe the Link GAT-1 flight simulator used in the washout testing experiments. Second, we describe the implemented pitch optimal washout system and its circuit. Third, we describe briefly the features of the circuit that implements the nonlinear washout using PLQ (Pseudo Linear Quadratic) and the interpolator circuit used to obtain approximations to intermediate washouts not implemented. Fourth, we describe two preliminary experiments which lead to the "flaps down" experiment. Furthermore, a take-off experiment is described which indicates the advantage of a washout system that uses an additional "integrator state" which is the integral of the motion-base pitch angle minus some fraction of the simulated airplane pitch angle. This is an implementation where we combine a very simple deterministic washout and our stochastic washout (see the example in section VI.2 and take-off example in Figure 32).

2.1. The Link GAT-1 Flight Simulator

In order to test the concept of the optimal washout, we used the Link GAT-1 flight simulator that is in the Man Vehicle Laboratory (Figure 1). This is a General Aviation simulator Trainer that resembles a Cessna 150/152 light aircraft and has three degrees of rotational freedom: yaw, roll and pitch, in the order of outer to inner gimbal. It was modified to have a display of horizontal strips on the two side windows. These strips were pitched and rolled to give the pilot pitch angle and roll angular velocity visual out-the-window sensation. The GAT-1 is 14 years old and is still being manufactured with only slight modifications. It has a very clever design which takes advantage of the simple washout it used in order to simplify its circuits. The washout it uses has the following descriptive characteristics:

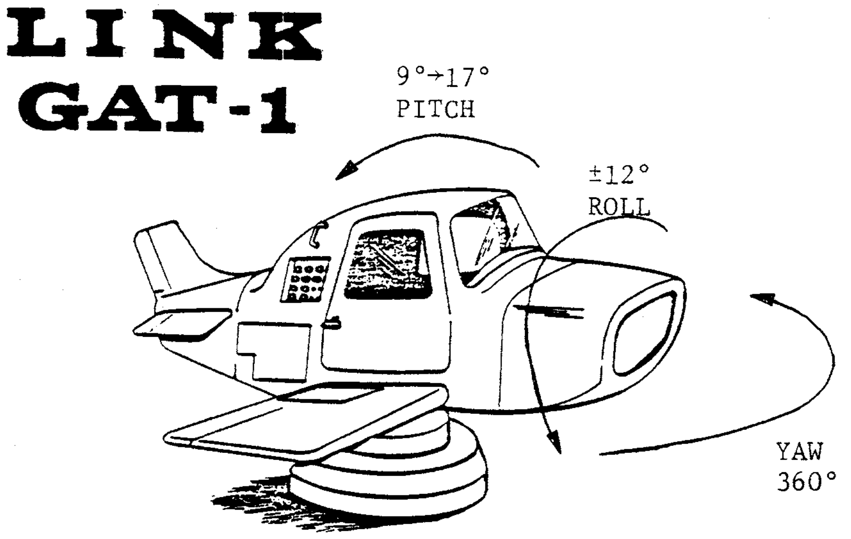


Figure 1. The 3 degrees-of-freedom excursion limitations of the GAT-1 flight simulator.

1. The pitch angle is 1/2 of the computed airplane pitch angle and is limited to +16 *degrees* pitch up and —8 *degrees* pitch down.
2. The roll angle is 1/6 of the computed airplane roll angle and is limited to ±12.5 *degrees* off the erect position.
3. The yaw angle is not limited, but its rate is limited to 30 *deg/sec*. There are no specifications for the pitch or roll rate limits.
4. The GAT-1 simulator has **no** linear motion capabilities. Furthermore, the computed linear motion of the airplane has no influence on the simulator motion. Thus the optimal washout is an improvement in that respect.

The GAT-1 design takes advantage of the simple washout by using the motion-base itself as the last step of the integration of the airplane equations of motion. This is done by giving an angular velocity control input to the motion-base. Thus the motion-base is in a closed-loop, since its angles are fed back into the airplane computations. This has the advantage of improving the motion quality as usually considered for feedback systems, but does not enable one to fly the simulator with the motion off. Also, all the flight instruments in the cab are properly scaled measurements of the motion base angles. This makes it very difficult to modify the current GAT-1 washout and thus only the simplest axis to modify, the pitch axis, was adapted for use with the optimal washout. The other point to note is that the pitch and roll rotation axes are below the center of mass of the simulator cab, which makes them open-loop unstable. This also implies that the motion-base integration of angular rate commands is only approximate, as will be discussed later.

2.2. Pitch Optimal Washout Implementation

A computer program using optimal control algorithms was written to design the optimal washout gains and simulate the resulting washout (Appendix B). This design has been implemented as one plug-in board for the GAT-1 flight simulator and another small card for the extension to a nonlinear washout. It also required a major irreversible change in the pitch attitude computation board, minor changes in three other boards (TIME DIVISION, ALTITUDE and RELATIVE WIND) and some additional wiring on the back plane of the simulator. The overall implementation included five functional elements: pitch angle computation decoupling, adding surge linear acceleration computations, a^a_{xa} (along the x airplane body axis), pitch axis motion-base modeling, pitch optimal washout system and a nonlinear extension of the washout system. Detailed documentation of these themes appears in the following appendices:

B. Washout design program.

C. Usage instructions and operation of the GAT-1 pitch optimal washout system.

D. Pitch motion-base modeling.

E. Circuit performance testing of the pitch optimal washout.

F. GAT-1 modifications to adapt it for use with the pitch optimal washout system.

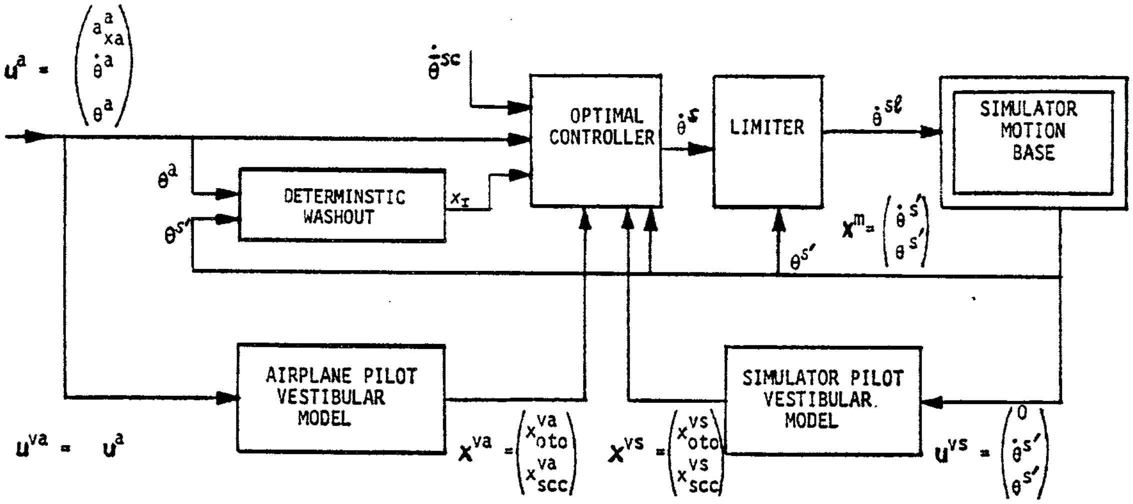
H. Parameters of the OWS used in the experiments.

Next a system and circuit level description of each of the five functional elements is given.

The first step in the washout implementation was to decouple the computation of the pitch angle from the simulator pitch motion by modifying the pitch axis computation and adding an integrator to compute the pitch angle from the available pitch rate signal. This modification changes somewhat the longitudinal characteristics of the simulated airplane. The most notable change is the reduction of the phiogoid period by about 20 percent which seems to improve the realism of the simulator and corresponds to a more exact implementation of the equations of motion provided by the Link (Appendix A of chapter II). In the original implementation, the integration is approximate due to the motion-base's "inverted pendulum" dynamics. One discrepancy in the current implementation (not in the original GAT-1) is the use of the approximation $\sin\theta^a = \theta^a$ for pitch angles $-16^o \leq \theta^a \leq 32^o$. The quoted range of θ^a indicate hard saturation limits, implemented to replicate the original GAT-1 pitch angle limitations. After these modifications were made, the GAT-1 could "fly" with pitch motion off. The second step is the computation of the forward acceleration which was added onto the RELATIVE WIND card as shown in Appendix F.

The closed-loop washout system design requires a model for the GAT-1 motion-base pitch axis. A second order inverted pendulum model with experimentally fitted parameters was used (Appendix D). This model was also implemented as a circuit on the pitch optimal washout board so as to enable a safe and easy testing of the implemented washout. Finishing the third step we could continue to the core implementation of the pitch optimal washout.

A general block diagram of the pitch optimal washout system is given in Figure 2 and a general circuit diagram is given in Figure 3. This is a closed-loop washout system which has, among others, the advantage of being relatively simple to implement. The implementation basically requires the implementation of two vestibular models, (Figure 4) one for the reference airplane pilot and one for the simulator pilot; beyond that, the washout itself is merely a summation of all the states of the system using the precomputed optimal gains. An additional circuit (Figure 4) is an integrator which is used in some washout designs to obtain a simulator pitch angle that is independent of the surge acceleration input a^a_{ax}, at the steady state. This is expected to improve the take-off simulation by reducing the motion-base pitch angle due to the "g tilt" (simulation of linear acceleration by using gravity) before lifting the nose in take-off. Another supporting circuit is the reset circuit which starts the simulation by zeroing all the internal system states;


$u^a = \begin{pmatrix} a^a_{xa} \\ \dot{\theta}^a \\ \theta^a \end{pmatrix}$
$\dot{\bar{\theta}}^{sc}$
OPTIMAL CONTROLLER
$\dot{\theta}^s$
LIMITER
$\dot{\theta}^{s\ell}$
SIMULATOR MOTION BASE
θ^a
DETERMINSTIC WASHOUT
x_I
$\theta^{s'}$
$\theta^{s'}$
$x^m = \begin{pmatrix} \dot{\theta}^{s'} \\ \theta^{s'} \end{pmatrix}$
AIRPLANE PILOT VESTIBULAR MODEL
SIMULATOR PILOT VESTIBULAR. MODEL
$u^{va} = u^a$
$x^{va} = \begin{pmatrix} x^{va}_{oto} \\ x^{va}_{scc} \end{pmatrix}$
$x^{vs} = \begin{pmatrix} x^{vs}_{oto} \\ x^{vs}_{scc} \end{pmatrix}$
$u^{vs} = \begin{pmatrix} 0 \\ \dot{\theta}^{s'} \\ \theta^{s'} \end{pmatrix}$

KEY:

u^a airplane motions

a^a_{xa} - longitudinal linear acceleration in airplane body axes

$\dot{\theta}^a$ - Euler pitch rate

θ^a - Euler pitch angle

$\dot{\bar{\theta}}^{sc}$ extra input for future use for a predictive washout

u^{va} airplane pilot vestibular input

u^{vs} simulator pilot vestibular input

$\dot{\theta}^{c\ell}$ simulator pitch axis command

x^{va} airplane pilot vestibular model states

x^{va}_{oto} of the otolith

x^{va}_{scc} of the semicircular canal

x^{vs} simulator pilot vestibular model states (components similar to those of x^{va})

x^m motion base states

$\dot{\theta}^{s'}$ - Euler pitch rate

$\theta^{s'}_t$ - Euler pitch angle

OPTIMAL CONTROLLER:

$$\dot{\theta}^s = -f^n u^a(t) - f^{va} x^{va}(t) - f^{vs} x^{vs}(t) - f^m x^m(t) - f_I x_I(t) - k_{sc}\dot{\bar{\theta}}^{sc}(t)$$

$$x_I = {}_{t_0}\int^t \theta^{s'}(\tau) - \mu\theta^a(\tau)d\tau \qquad 0 \le \mu \le 1$$

In the PLQ controller, the gains are not constant but are functions of $\theta^{s'}$, i.e. $f^n(\theta^{s'})$, $f^{va}(\theta^{s'})$, $f^{vs}(\theta^{s'})$, $f^m(\theta^{s'})$, $f_I(\theta^{s'})$

Figure 2. General block diagram of the pitch-surge optimal washout system.

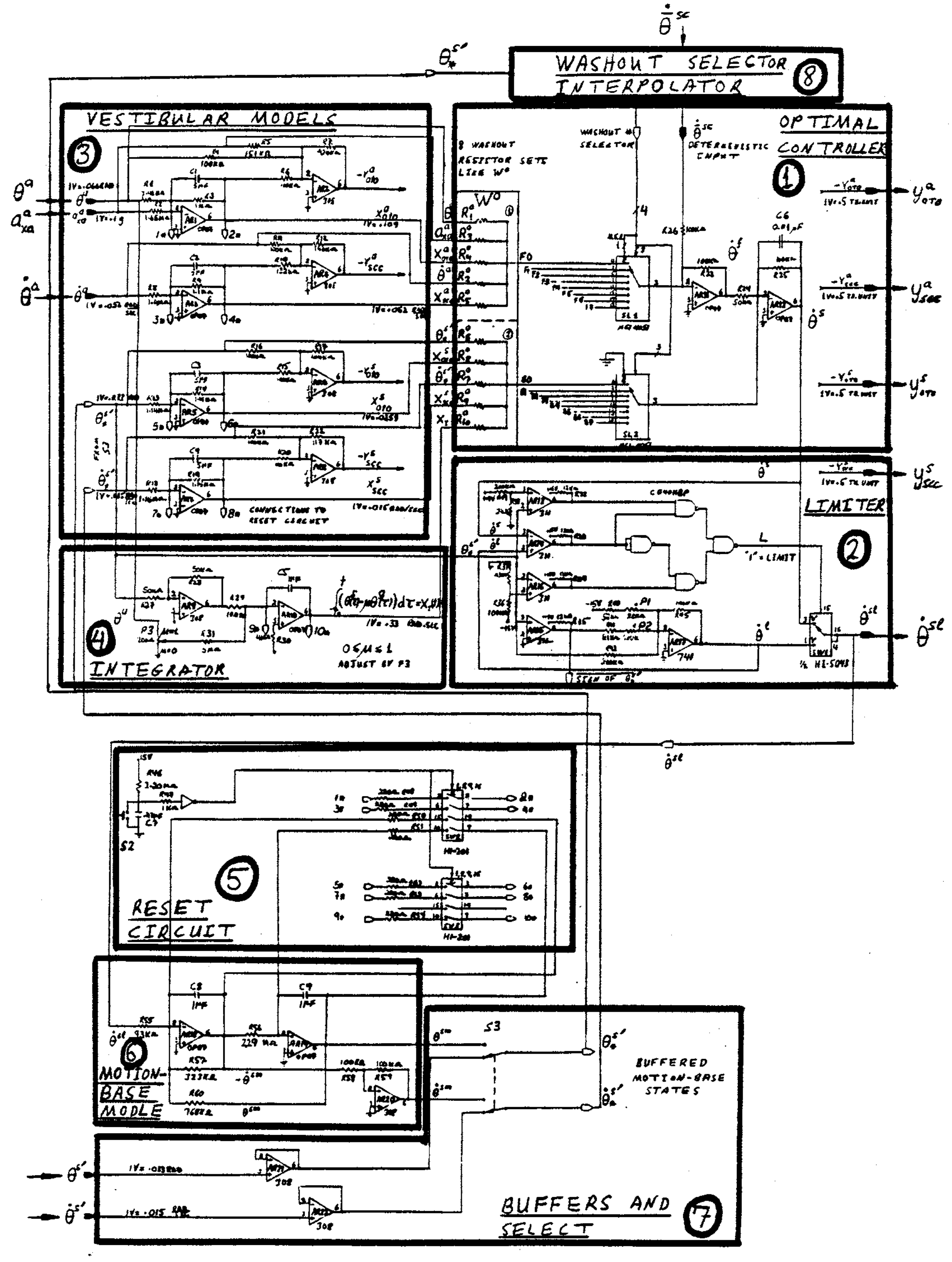


Figure 3. Overall circuit and wiring of the pitch optimal washout system.

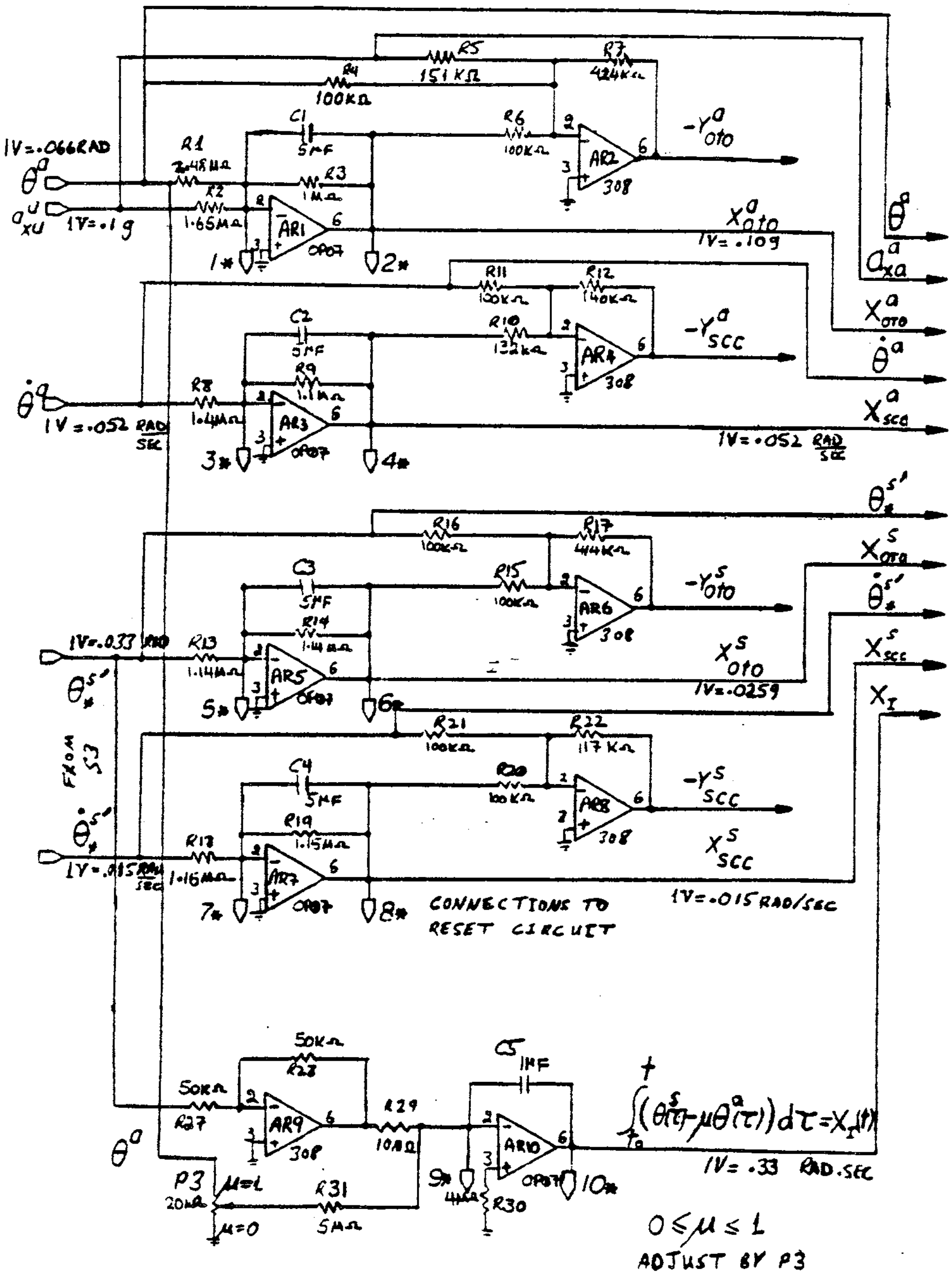


Figure 4. The two sets of vestibular models; each for the "surge" otolith and the "pitch" semicircular canal. At the bottom (AR9, AR10) is the deterministic washout including an integrator.

of the integrator, the vestibular models and the pitch motion-base model.

The outputs of these vestibular models are in terms of threshold units, where an output of one is the minimum input the pilot can perceive under the expected pilot workload. The circuits implementing these four models are given in Figure 4 and there equations are given below.

Otolith: u^o= linear acceleration input, y^o= otolith output.
Transfer function form

$$y^o(s) = G_o \frac{s+b_o}{s+a_o} u^o(s) \tag{1}$$

State space form

$$\dot{x}_{oto}(t) = -a_o x_{oto}(t) - (a_o - b_o) u^o(t) \tag{2}$$

$$y^o(t) = G_o x_{oto}(t) + G_o u^o(t) \tag{3}$$

Semicircular Canal: u^s= angular velocity input, y^s= semicircular canal output.
Transfer function form

$$y^s = G_s \frac{s+b_s}{s+a_s} u^s(s) \tag{4}$$

State space form

$$\dot{x}_{scc}(t) = -a_s x_{scc}(t) - (a_s - b_s) u^s(t) \tag{5}$$

$$y^s(t) = G_s x_{scc}(t) + G_s u^s(t) \tag{6}$$

The values used are

Otolith	Semicircular Canal
G_o= 21.17 $1/g$	G_s= 40. sec/rad
a_o= 0.19 rad/sec	a_s= 0.169 rad/sec
b_o= 0.076 rad/sec	g= 9.8 m/sec^2 (acc. of gravity)

The pitch optimal washout (Figure 3) includes five parts: controller, limiter, vestibular models, deterministic washout and integrator and reset circuit. The controller itself includes four parts: two summers and two gain selectors (Figure 5):

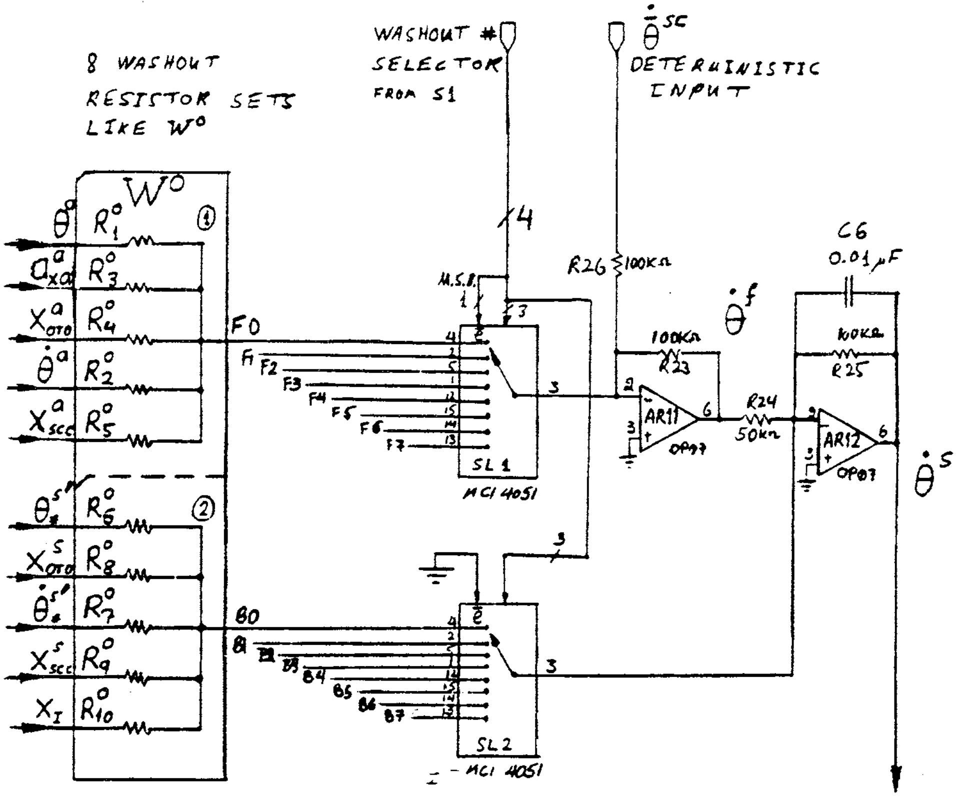


Figure 5. The optimal controller and washout selector.

1. The first summer is used to compute the feed-forward command, $\dot{\theta}^f$, using the positive gains set by R_1^i to R_5^i of the $i^{\underline{th}}$ washout, W^i. The inputs are the simulated airplane states θ^a, $\dot{\theta}^a$ and a_{xa}^a, the reference airplane pilot vestibular states x_{oto}^{va} and x_{scc}^{va} and finally the deterministic input command $\dot{\bar{\theta}}^{sc}$.

2. The second summer sums the feed-forward command, $\dot{\theta}^f$, to the feedback signals; motion base states $\theta^{s\prime}$ and $\dot{\theta}^{s\prime}$, and the simulator pilot vestibular model states x_{oto}^{vs} and x_{scc}^{vs}. The gains at the second summer are all negative, which turns out to be the appropriate sign for all the feedback signals.

3. The first gain selector chip, CD4051B, chooses one of the eight sets of feed-forward gains, which are determined by the five resistor values R_1^i through R_5^i. It has a four bit select control; three bits select which of the eight gain settings will be connected to the first summer, while the fourth bit, $\bar{e}$, enables us to disconnect the input to the washout system from the airplane computed states. Thus, when the washout selector switch is set at position 8 or 9, the $\bar{e}$ bit is set to "1" and the feed-forward signals from the simulated airplane are set to zero; however the $\dot{\bar{\theta}}^{sc}$ remote input is left connected. The feedback gains are not set to zero, so the originally

unstable motion base will not fall over.

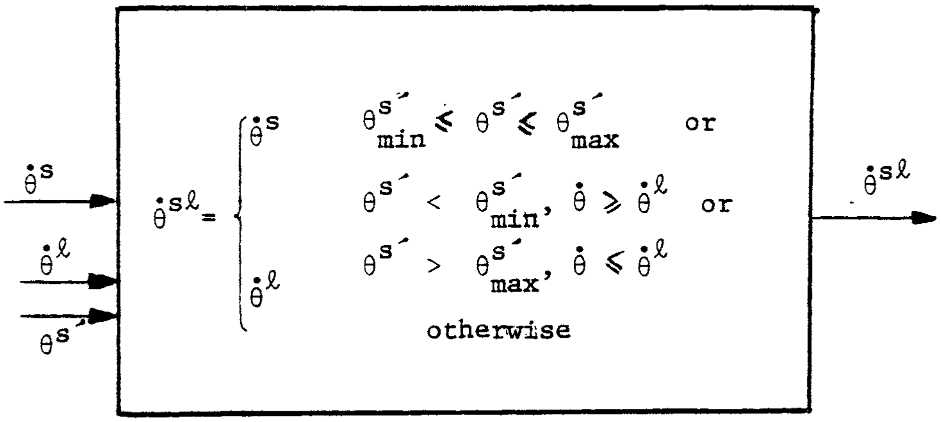
4. The second gain selector chip, CD40051B, chooses one of the corresponding eight sets of feedback gains which are determined by the five resistors R_6^i through R_{10}^i. The $\bar{e}$ bit is always set to "0" so that in all cases, one set of gains is used. The other three selector bits are connected in parallel to those of the first gain selector, so that one of the eight washouts is chosen in positions 0 through 7 and in position 8 and 9 feedback gain sets 0 and 1 are chosen respectively.

The limiter circuit limits the motion-base pitch angle $\theta^{s'}$, so that the simulator cab will not reach the mechanical end stops and will remain within $\theta^{s'} = -9\,degrees$ (lower limit) and $\theta^{s'} = 17\,degrees$ (upper limit) (Figure 6 and 7). This is achieved by switching between the optimal washout command $\dot{\theta}^s$ and the limiting command $\dot{\theta}^\ell$ by the analog switch HI-5043. The logic that governs this switching is based on a test of whether the pitch angle $\theta^{s'}$ is in the above range and if the optimal washout command $\dot{\theta}^s$ will command the pitch angle to return to within its bounds Decided on the basis of the relative sign of the limiting command $\dot{\theta}^\ell$ and $\dot{\theta}^s$, see Table 1.

Finally, there is a reset circuit (Figure 8) that enables us to zero all the internal states of the washout system (vestibular model, integrator, motion-base model). This circuit automatically resets the washout system on power on so that the motion-base will start up in the exact, level position (zero state) (Appendix C). In addition Figure 8 includes two buffers for the motion-base states.

2.3. Nonlinear washout and interpolator

The purpose of the nonlinear washout controller and interpolator circuit is two fold: (i) enable the choice of intermediate washouts that are approximated by interpolation between two of the eight implemented washouts; (ii) implement a nonlinear washout using PLQ (Subsection IV.2.3). The interpolation is achieved by modulating the duty cycle of the switching between two successive washouts. The switching frequency used is $400\,Hz$ and the interpolation is based on a 4 bit binary number selected manually by a thumb-wheel switch or by the nonlinear washout control circuit. The nonlinear washout control circuit uses a 6 bit A/D conversion of the motion-base pitch angle. The two most significant bits select one out of four washout gain settings and the other four bits are used to interpolate between that washout and the next higher number washout. The system is designed with a


$\dot{\theta}^{s}$
$\dot{\theta}^{\ell}$
$\theta^{s'}$
$\dot{\theta}^{s\ell} = \begin{cases} \dot{\theta}^{s} & \theta^{s'}_{min} \leqslant \theta^{s'} \leqslant \theta^{s'}_{max} \quad \text{or} \\ & \theta^{s'} < \theta^{s'}_{min},\ \dot{\theta} \geqslant \dot{\theta}^{\ell} \quad \text{or} \\ & \theta^{s'} > \theta^{s'}_{max},\ \dot{\theta} \leqslant \dot{\theta}^{\ell} \\ \dot{\theta}^{\ell} & \text{otherwise} \end{cases}$
$\dot{\theta}^{s\ell}$

$$\dot{\theta}^{\ell} = -f_1\theta^{s'} - f_2\dot{\theta}^{s'} + k\theta^{\ell\ell}(\theta^{s'})$$

where

$$\theta^{\ell\ell}(\theta^{s'}) = \begin{cases} \theta^{s'}_{max} & \theta^{s'} \geqslant 0 \\ \theta^{s'}_{min} & \theta^{s'} < 0 \end{cases}$$

Inputs: $\dot{\theta}^{s}$ - optimal pitch rate command

$\dot{\theta}^{\ell}$ - limitter loop pitch rate command

$\theta^{s'}$ - motion-base pitch angle

Constants: $\theta^{s'}_{max}$ - maximum allowable motion-base pitch angle

$\theta^{s'}_{min}$ - minimum allowable motion-base pitch angle

$\theta^{s'}_{max}$ = 17 degrees, $\theta^{s'}_{min}$ = -9 degrees

Figure 6. Block diagram of the pitch angle limiter.

$\theta^{s'}$	$\dot{\theta}^{l}-\dot{\theta}^{s}$	V_{max}	V_{min}	V_c	L
$\theta^{s'}_{min}\leqslant\theta^{s'}\leqslant\theta^{s'}_{max}$	$\leqslant 0$	"0"	"0"	"0"	"0"
	>0	"0"	"0"	"1"	"0"
$\theta^{s'}>\theta^{s'}_{max}$	$\leqslant 0$	"1"	"0"	"0"	"0"
	>0	"1"	"0"	"1"	"1"
$\theta^{s'}<\theta^{s'}_{min}$	$\leqslant 0$	"0"	"1"	"0"	"1"
	>0	"0"	"1"	"1"	"0"

L = "0" ⇒ No limit $\dot{\theta}^{sl}=\dot{\theta}^{s}$

L = "1" ⇒ Limit

Table 1: Logic truth table for the limiter

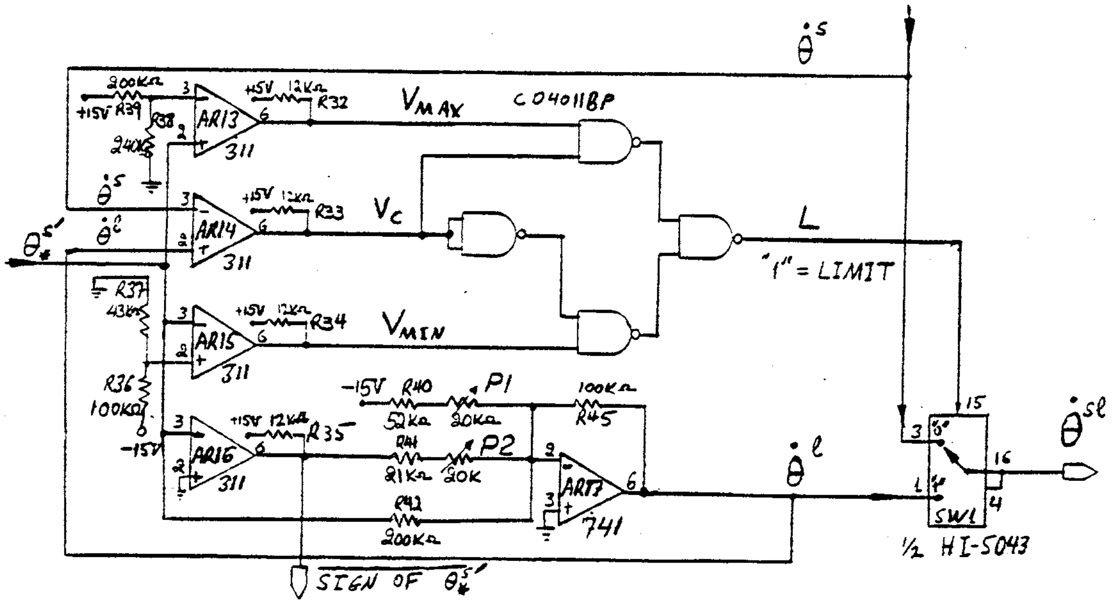


Figure 7. The pitch angle limiter.

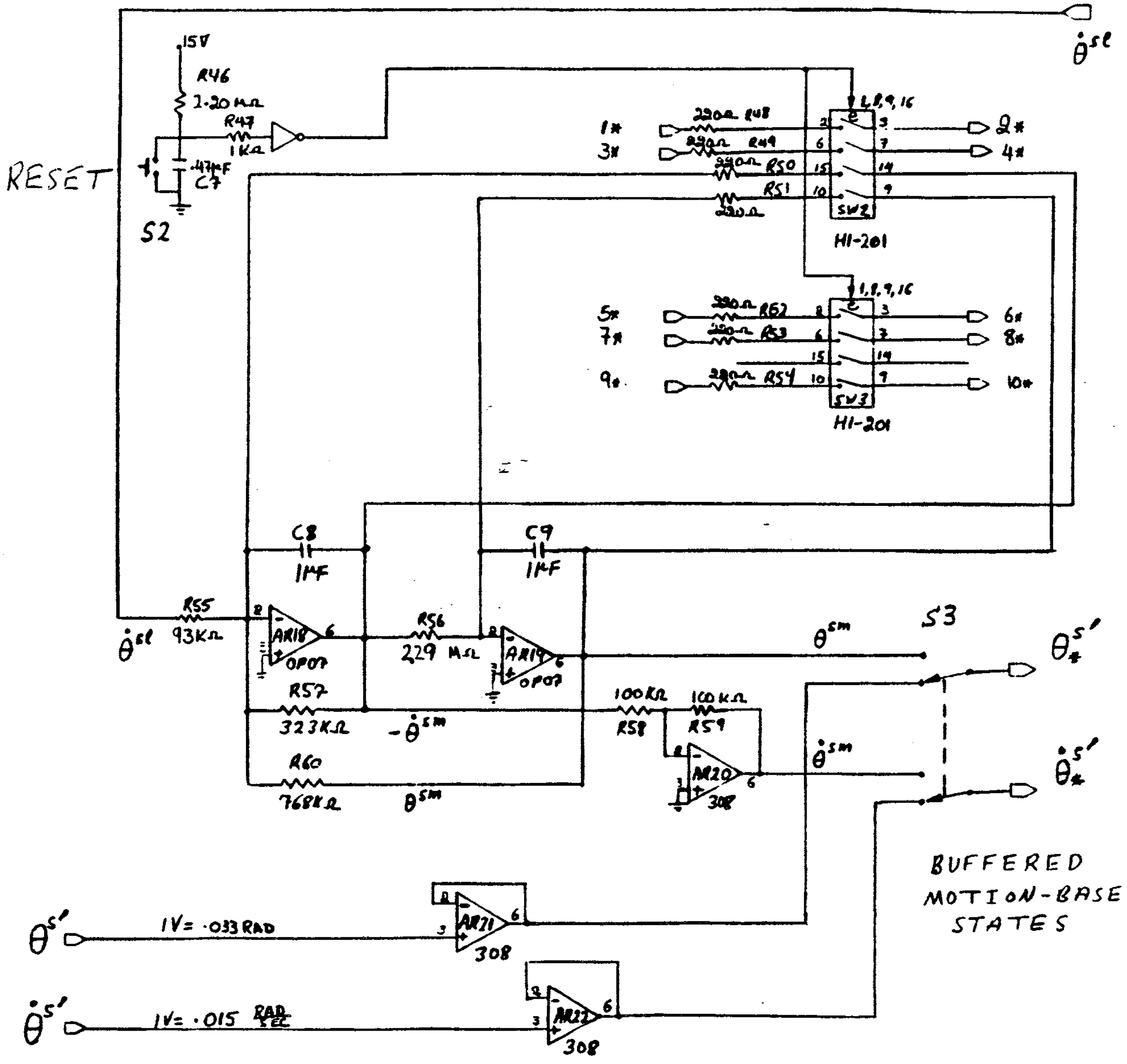


Figure 8. Reset circuit, motion-base buffers and the motion-base model used only for testing.

saturation that does not allow a rap around of the interpolation after selection of the highest number washout binary "11".

2.4. Experimental Design

Four experiments were used to evaluate the optimal washout system:

1. Blind test.
2. Washout detection.
3. Flaps-down detection.
4. Take-off.

The experiments 1, 2 and 4 were performed in order to get an insight to the effects of the optimal washout, while experiment 3 lead to data that could support statistical analysis. The insight found in these experiments was much more interesting then the statistically verifiable results found in experiment 3.

Beyond the above experiments several other experiments were done too, their conclusions follow. It was found that the visual display made the pilot feel that the simulation was more real but it did not seem to effect his performance. The pilot associates changes in simulator motion as part of the airplane aerodynamic characteristics. The pilot was required to strap himself using two belts during the experiment. One was a regular waist safety belt and the other was a chest belt that made sure that the pilot had his back leaning against the back of the seat. Both belts were not tightly attached so the pilot would be as comfortable as possible. The reason for the chest belt was that pilots tend to lean forward when the motion-base is pitching up and thus we would loose a large portion of the "g-tilt" effect that gives the pilot a sensation of surge linear acceleration (it would be interesting to measure this effect).

Appendix G includes the designs and simulation of two of the optimal washouts used in the experiment (named #0 and #2); the design parameters of the other washouts used are given in Appendix H. In washout #0, the design uses equal weights in the cost for one threshold unit error of the semicircular canals and the otoliths i.e. $Q_o/Q_s = 1$. In washout #2, it is assumed that the semicircular canals are three times more sensitive than in washout #0 (have one third the threshold) $(Q_o/Q_s = 0.1)$.

2.4.1. Experiment 1—Blind Test

Five pilots (JW, IM, JF, JN, SU) were tested to obtain their impression of four washout systems–the original GAT-1, no pitch motion, and two choices of optimal washouts #0 and #2. This was a blind test in the sense that the experimenter did not explain anything about the washout system and the possible motions of the simulator to the pilots.

Pilot Instructions

The only information that the pilot had was that he was about to use a Link GAT-1 flight simulator that resembled a Cessna 150/152 type airplane. The pilots were asked to evaluate the simulator, especially its motion, while flying their choice of maneuver for 30 to 60 minutes. Beyond that the pilots were asked to notify the experimenter of any noticeable changes in the simulator. Although not specifically explained, it was clear to the subjects that the experimenter's main interest was in the simulator's motion.

Experimental Set Up

(i) No visual out-of-the-window display was used.

(ii) During flight, the simulator's pitch motion washout could be changed between the two optimal washouts and the no pitch motion "washout". To change back to the original GAT-1 washout, the simulator had to "land" and be switched off so that the ALTITUDE card could be changed. There were some differences in the aerodynamic characteristics of the longitudinal modes between the GAT-1 original washout and the others. In the GAT-1 case, the phiogoid mode was about 55 seconds compared to 40 seconds for the other washouts (including the no pitch motion washout).

(iii) Due to a simulator problem most subjects reported that they had to depress the right pedal (rudder) and/or ailerons to compensate for the tendency of the simulator to roll left.

2.4.2. Experiment 2—Washout Detection

Three pilots (JW, IM, JH) were subjects in an experiment to detect washout change. The purpose of this experiment was threefold:

1. To determine if the changes in the pitch washout (motion) produced by two different optimal washouts are detectable.
2. To determine the applicability of using a coordinated turn and a flaps down/up maneuver for a washout change detection experiment.

3. To determine if a flaps down/up can be used in an airplane change detection experiment.

The experiment included the two washouts #0 and #2. Based on the pilot's report and the simulated airplane pitch angle time history $\theta^a(t)$, the washout used was deduced.

About three quarters of the experiment time was devoted to training. The first pilot JW flew for 70 minutes. The second pilot IM flew for 150 minutes in two sessions, 45 minutes and 105 minutes with a 120 minute break. The third pilot JH flew for 120 minutes.

The experiments were supplemented by an actual flight in a Cessna 172 by the experimenter, JI. This was to achieve objective 3 and to compare the feeling of flaps down/up in an actual airplane to that achieved in the simulator.

Pilot Instructions

A general explanation of how the optimal washout works and the objective of the experiment was given to the two pilots before the experiment. The pilot was asked to comment on his experiences during the experiment.

The washouts were changed while the pilot was at level flight, so as to minimize the transition motion due to washout change (under these conditions the washout change can not be detected by the pilot). After that, the pilot was asked to do an approximately 360 degree, two minute coordinated turn (30 degree bank angle). Sometimes the pilot would do two turns, one to the left and one to the right at the end of which the pilot was asked to tell which washout was used. The same type of experiment was repeated for the condition of level flight when the pilot was asked by the experimenter to lower the flaps and after stabilizing to put the flaps back up. The pilot was instructed to primarily maintain altitude and secondarily to maintain heading. The pilot reported the maximum altitude deviation he achieved during each experimental run The experimenter informed the pilot after each run whether his washout detection was correct.

Experiment Set Up

There was a visual out-the-window display for roll and pitch (horizontal stripes).

Roll display was off for the second half of the experiment with pilot JW, due to motion sickness symptoms reported by the pilots. There was continuous voice communication between the experimenter and the subject. During the experiment, the following time histories were recorded:

1. $\theta^a(t)$ - simulated airplane pitch angle
2. $a^a_{xa}(t)$ - simulated airplane forward acceleration in body axes (surge axis acceleration).
3. $\theta^s(t)$ - motion base pitch angle.
4. $\dot{\theta}^a(t)$- simulated airplane pitch angular rate, only for pilot IM for 10 *seconds* after change in flaps.

2.4.3. Experiment 3—Flaps-down Detection

In this experiment the pilot was required to hold level flight and to detect when full flaps were put down randomly by the experimenter. Eleven pilots were tested (CO, IM, JW, JH, DM, AE, LH, PM, GO, EA, YM), the first seven for 90 minutes each and the latter four for 20 minutes each. The experimenter and a few other non-pilots were also tested for 30 minutes each.

The pilot flew the airplane straight and level as best he could and told the experimenter when the airplane was stabilized and he was ready for the flaps down transition. The experimenter verified that the airplane was stabilized and at level flight by looking at a storage scope trace of the airplane pitch angle; then after a subjective random time of a few seconds the run was started. The chart recorder was started and flaps were put down one second later. The flaps were put back up after 5 more seconds and the chart recording was stopped after a total time of eight seconds. When flaps are put down the airplane tends to "balloon" i.e. nose up and start gaining altitude. In order for the pilot to achieve his task his proper elevator control is to push the nose down (pitch down). The initial sensory signals the pilot perceives are a very small pitch up rotation (seems below the pilots threshold) and a quite strong deceleration cue of the airplane (approximately 0.2 g). This deceleration gives the pilot the false sensation that the airplane pitched down which if followed leads to an incorrect airplane elevator control.

Post experiment, runs that were judged to have non-standard initial conditions (labeled I.C. not 0) were rejected. These judgments were based on non-zero preflaps-down of y^a_{scc} and a^a_{xa} which correspond approximately to pitch rate and surge acceleration. Note that since the GAT-1 has no linear motion capability then actual pitch down was used by the optimal washout system to simulate the airplane deceleration due to the flaps-down transition.

A preliminary experiment was done to test the effect of asking the pilot to press the "detection button", on his control during the flaps-down experiment. It was found.that it did not have any substantial effect on the pilot controls.

Four numbers were used to summarize the data recorded from each run (Figure 9): T_d pilot detection time, T_c time till pilot first control after flaps-down, T_p time till pitch angle started to drop, ΔH how much did the airplane "balloon" after flaps-down transition as reported by the pilot.

The objectives of this experiment are:

1. To find the relation between the pilot flaps-down detection time and the washout used.
2. Is the pilot performance measured by the "ballooning" (increase in altitude, ΔH, right after the flaps down transition) affected by the different washouts tested.
3. Can the linear acceleration motion cue confuse the pilot to the extent of making an initial control in the wrong direction, i.e. initially pull up the nose, using the elevator control rather than push it down? The answer to this question is possibly the reason for many general aviation accidents that occur due to a stall during a landing approach. During this maneuver, flaps are put down and the airplane is both at low speed and in a turn (airplane is banking). Thus, it is very close to a stall and small incorrect judgment of the nose position (pitch angle) due to the linear acceleration cue associated with flaps down can cause an incorrect control which would stall the airplane at the low altitude and cause an accident.
4. One wishes to find which washout generates the best match to the pilot's behavior in the real airplane. This is not achieved due to lack of comparison tests in the real airplane.

Pilot Instructions

The pilot was briefed on the experimental objectives 1 and 4. The pilot was told to press a small microswitch when he detected the flaps-down transition. This switch was mounted on his control yoke next to his left thumb (for pilot CO, next to his right index finger). The pilot was instructed to fly level toward direction 030

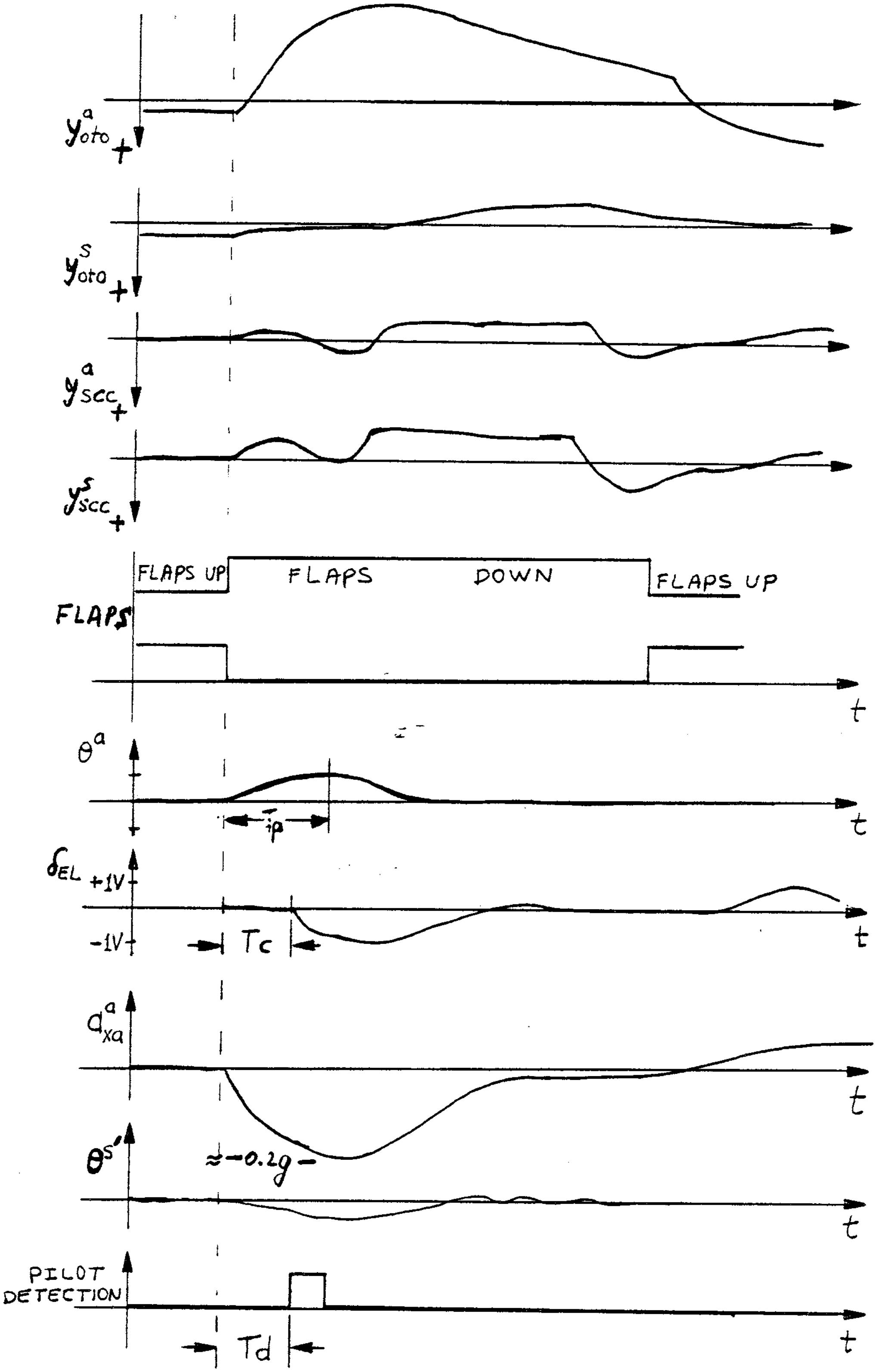


Figure 9. Recorded data for each *flaps-down* experiment and the definitions of T_d—detection time, T_c—time of first control, T_p—time to peak of pitch angle.

at an altitude of 1000 $feet$ with the engines at 2300 rpm. This rpm corresponds to 100 mph (sometimes up to 104 mph) which is the maximum speed at which flaps may be lowered. The pilot's main task was to hold altitude (independent of the flaps transition), second he was to indicate when he noticed the flaps-down transition and third to hold heading. The pilot was also required to report his initial increase in altitude due to the flaps-down transition.

Experiment Set Up

There was a visual out-the-window display for roll and pitch (horizontal stripes). There was continuous voice communication between the experimenter and the subject. During the experiment, the following time histories were recorded:

1. $\theta^a(t)$—simulated airplane pitch angle
2. $a_{xa}^a(t)$—simulated airplane forward acceleration in body axes (surge axis acceleration).
3. $\theta^s(t)$—motion-base pitch angle.
4. $\delta_{EL}(t)$—elevator control input of pilot
5. Flaps position up or down (event marker)
6. Pilot indication switch (event marker)
7. $y_{oto}^a(t)$—reference airplane pilot otolith model output.
8. $y_{oto}^s(t)$—simulator airplane pilot otolith model output.
9. $y_{scc}^a(t)$—reference airplane pilot semicircular canal model output.
10. $y_{scc}^s(t)$—simulator airplane pilot semicircular canal model output.

2.4.4. Experiment 4—Take-off

Figure 10 shows an inflight measured recording of the surge axis specific force of a single engine Cessna 172 during take-off (pilot WH). The dotted line represents an estimate of the pitch angle. The experiment focused on the pitch up during take-off in the time window $t = 20$ to $t = 50$ $seconds$. The interesting point about the pitch up ($t = 29$ to 33 $seconds$) is that the pitch information sensed by the semicircular canal and the otoliths do not correspond to the usual pitch up sensations; the pitch angle increases from approximately 0 to 4.5 $degrees$ (semicircular canal cue) while the linear acceleration sensed in the surge direction decreases from approximately 0.23 g to 0.13 g (otolith cue). The linear acceleration usually increases from 0 to 0.08 g for the given pitch angle. This unusual set of vestibular cues gives the washout

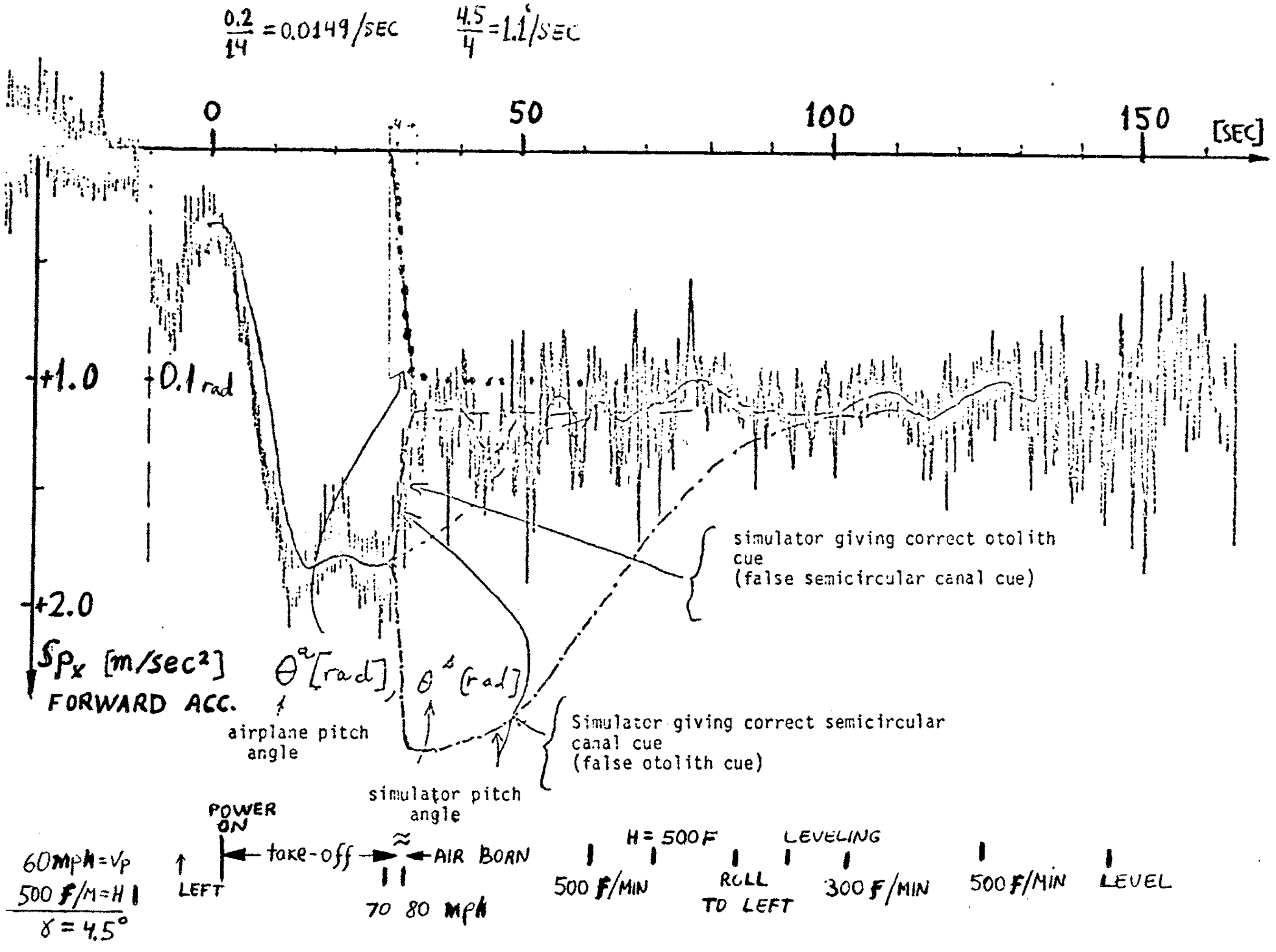


Figure 10. In flight measurement of the take-off surge axis specific force, an estimate of the airplane pitch angle (dotted line) and possible expected simulator pitch motion (dashed line and dash-dot line). Note that the positive direction is down.

design (having only pitch rotation) a difficult decision to make: Give the correct otolith cue by pitching down from, say, 10 *degrees* to 6.5 *degrees* dashed line in Figure 8 (assuming that we use "g-tilt" to simulate the linear acceleration at the time before $t = 29\,seconds$) or give the correct pitch rate cue by further pitching up from 10 to 14.5 *degrees*—dash-dot line in Figure 8. Since we are constrained in only having pitch rotation with no available linear surge motion, the design of the washout has to make an explicit trade-off between the otolith and the semicircular canal cues. It is interesting to discover what his trade-off is in terms of the design of the motion washout. One experimental difficulty is that the pitch rate involved is quite low, 1.1 deg/sec peak, which is on the order of the pilot's rotation threshold. Therefore, a pitch rotation stripe pattern was used to enhance pitch rotation detection.

Pilot Instructions

Put the brake on. Check that the flaps are down and the throttle is at minimum (all the way out). Release the brake and apply full throttle. Run down the runway and take-off at 65 mph with a climb rate of 500 $feet/min$ to an altitude of 300 $feet$. Fly level for 30 *seconds* and then land using flaps as usual. As soon as you have landed pull the brake to stop. During the whole flight maintain the same heading 030.

Experiment Set Up

The set up was the same as in experiment 3—flaps-down detection except that the event marker did not record the flaps position.

3. Results

3.1. Experiment 1—Blind test

All five pilots did not notice any difference between the three washouts tested #0, #2 and the original Link system which has no linear acceleration motion cueing; This is so even though each pilot did several take-offs and landings where the linear acceleration cue given by washout #0 is very noticeable compared to the null cue of the original Link washout. Furthermore, most non-pilots who were given a demonstration of the washouts noticed the acceleration cue during take-off and the braking cue after landing. The pilots did not notice any difference even when specifically asked, after the experiment, about the take-off and braking. One exception was pilot IM who said that he noticed something during the take-off and braking but could not tell what it was, and after a few days thought at home he came up with the right idea of what happened. During IM's experiment it was interesting to observe changes in the simulator motion during a coordinated turn due to interchanging between washouts #0 and #2, which were not noticeable by the pilot, just by looking at the simulator. It is interesting to note that pilot JF had very much experience on a Link GAT-1 flight simulator which did not seem to help.

It seems that unless pilots are instructed as what to look for specifically, it is very difficult for them to be critical about the simulator motion, the situation is too complex. The pilots also interpret changes in the simulator motion as changes in the simulated airplane dynamics. Thus it was concluded that one should not attempt to ask a pilot to distinguish between these two changes but the experimenter should take on the task of what is the reason for the pilot's feeling. The pilot should be simply asked if there was any change in the simulation.

3.2. Experiment 2—Washout Detection

From this experiment it seemed that the change in pitch washout produced by a change of optimal washouts #0 and #2 could not be detected consistently from the pilots reports. It could be detected using the flaps-down transition by looking at the pitch angle recording. The coordinated turn is not a good maneuver to do this detection. It is possible that a dutch-roll maneuver is also a good candidate for a pitch washout detection experiment, although the reason is not clear to the experimenter. The flaps-down can be used for an airplane change detection experiment (experiment 3). The flaps-up transition seems to give less consistent results although the stimulus to the pilot is larger; it is harder to stabilize the airplane with flaps down. Following are the results for each of the three pilots.

Pilot JW

The first 17 washout detection runs were done using a coordinated turn as the underlying maneuver through which the pilot was to detect which washout was used #0 or #2. 12 runs were used as training. Out of the five test runs the pilot was wrong 4 times. During these tests the pilot was given no feedback if his judgment was right or wrong. This part was 33 minutes long.

A schematic drawing of the washout detection during the flaps-down is shown in Figure 11. In this experiment also flaps-up was tested. There were a total of 16 runs, of which 11 were training. Out of the 5 test runs, the pilot was correct in the first 3 and wrong in the last 2. After each run the pilot was informed if his judgment was right of wrong. In contrast with this result, looking at the recorded data and using the following criteria: washout #0 if $T_p > 2.5\,seconds$ and #2 otherwise (Figure 11); it was clear that 14 of the runs followed this criteria while the other 2 were inconclusive. Another conclusion is that the required training for washout #0 is shorter (4 runs) compared to 7 runs for washout #2. This result despite the fact that washout #0 was the first the pilot trained for. The pilot is considered to be in need of more training as long as the airplane pitch angle $\theta^a(t)$ and the surge acceleration $a^a_{xa}(t)$ are not a repetitive response to flaps up and down transition. The pilot reported that he used the rate of climb as his criteria to use the elevator control to push the airplane nose down.

It is concluded that although the pilot can not cognitively detect which washout was used the washouts difference showed up in his control responses which he was not aware of.

Pilot IM

During a total of 2.5 hours (with a 2 hour break in the middle) 22 runs (repeats) of the sequence: flaps-down, stabilize the airplane, flaps-up, stabilize the airplane. 16 of these runs were used as training to familiarize the pilot with the two washouts used #0 and #2. Out of the 6 test runs the pilot detected the washout correctly in 4 runs. The pilot reported that it is easier to do the task (keep level flight) with washout #2, his comment was that #0 led him to over control and that he felt the incorrect pitch motion due to the linear acceleration simulation by the washout system.

In contrast with JW it seems that IM can detect which of the two washouts were used using the flaps up and down transition.

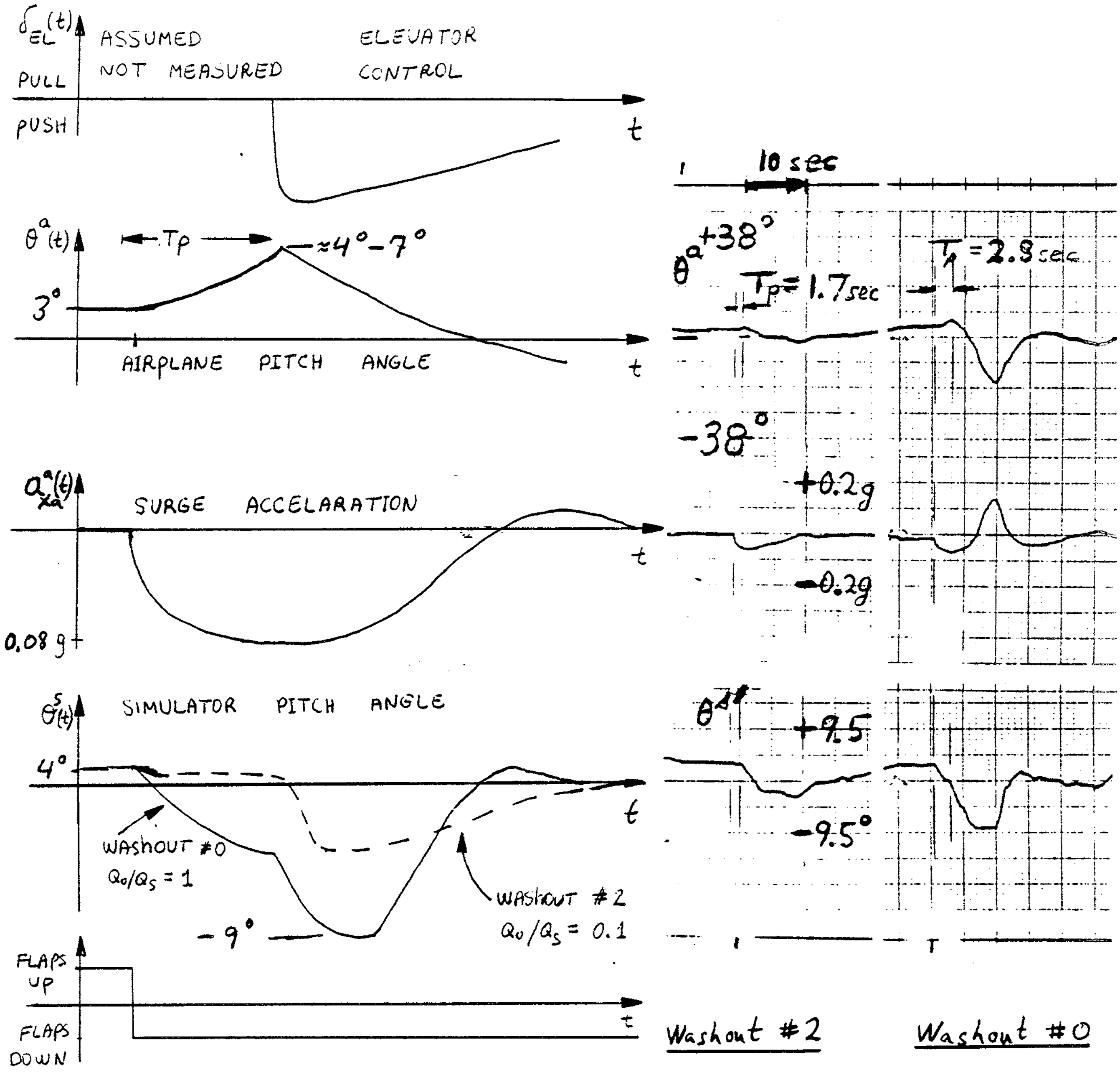


Figure 11. Response of pilot JW to flaps-down for the two washouts #0 and #2. (a) Recordings of a sample of two responses (b) Schematic drawing. Note that the elevator control is only estimated. If $T_p > 2.5$ it was washout #0, otherwise #2.

Pilot JH

The pilot was tested for 2 hours. The experiment included 19 runs of flaps down and up transition out of which 4 were used to test if the pilot could detect which washout #0 or #2 was used. The pilot made a correct detection only in 1 test run. The experiment also included the pilot's choice of a dutch roll maneuver to detect which washout was used. Using this maneuver the pilot detected which washout was used in all 4 test trials. These tests were done before flaps down/up maneuver was used. When repeated twice later in the experiment, to aid the washout detection using the flaps down/up maneuver the pilot made an incorrect detection.

During this experiment the effect of the visual pitch and roll stripe display was tested. The pilot could not detect the switching off of the motion of the visual display, nor did he detect any changes in the airplane characteristic due to that even when specifically asked.

During the first take-off using washout #2, the pilot hit the ground, i.e. he did not pull the nose up enough. This also happened in the 3rd take-off using washout #0, after using the Link washout. It is conjectured that due to the g-tilt used to simulate the acceleration during take-off the pilot judges that his nose is too high (close to stall condition) and thus he hits the runway.

When generally comparing the four washouts #0, #2, Link and no pitch motion, the pilot thought the Link was the best. The main reason seems to be the lower bandwidth of the pitch motion control system which gave the pilot a smoother flight. The pilot's comment was that using washout #0 made the flaps down transition feel "life like" (very real). Also the pilot thought that washout #0 was better then #2.

Other comments the pilot made were: the engine sound has too low a frequency, the yaw axis needs a little more damping, the simulator flies faster then usuall (this was also said by pilot JN). Also when washout #2 was changed to #0 "roll stability was different; roll was smother", "airplane was less sensitive in pitch—better". Washout #0, "problem in pitch, too sensitive to sudden inputs; jerky". "In the real airplane there is a dead zone in the elevator (pitch) controls which is not existent in the simulator". Simulator has "strange coupling between pitch and roll". Pilot like no-pitch-motion over washouts #0 and #2 since the pitch is not jerky; the pilot does not find any changes in airplane dynamic due to no-pitch-motion (not too surprising). It seems that pilot comments should be interpreted with great care.

3.3. Experiment 3—Flaps-down Detection

In this section we describe two aspects of this experiment: the initial pilot response and his response after training. The first is more interesting though the second one is more quantitative.

3.3.1. Initial Response Results

We start by showing two pilots responses, as a demonstration of the outcome present for all pilots tested, of the simulation of the deceleration during flaps-down transition. The main research interest of the first pilot demonstrated, CO, is in vestibular physiology. Furthermore CO has very through understanding of the airplane dynamics, inertial motion sensing, its effect on pilots perception, manual control and the design of motion in flight simulator. Unfortunately CO has only 270 *hours* (on Cessna 150/152) with last flight experience a year back, thought he had more recent experience on flight simulators. To cover up for this misfortune our second pilot EA is a commercial airline pilot with a record of 12,000 *hours* of flight experience, unfortunately mainly on a DC8.

Pilot CO

Four experimental runs of pilot CO are shown in Figure 12. These runs were performed with an optimal washout with $Q_o/Q_s = 1$, with the pilot's air speed indicator covered and with both pitch and roll visual display on. The runs 2, 3, 4 and 5 are in chronological order where 1 was the first run the pilot experienced. The pilot was tested also the day before for 45 *minutes* and had 11 run tests of flaps-down. Before the runs showed in Figure 12 were performed one run, 1, was done with no motion and the initial pilot response was in the correct direction i.e. push the elevator control. The pilot had only general knowledge of what is the motion given to him by the optimal washout system.

We see in Figure 12 that in all the runs the initial control of the pilot was in the wrong direction (pull nose up) except in run 4 were it was initially correct but

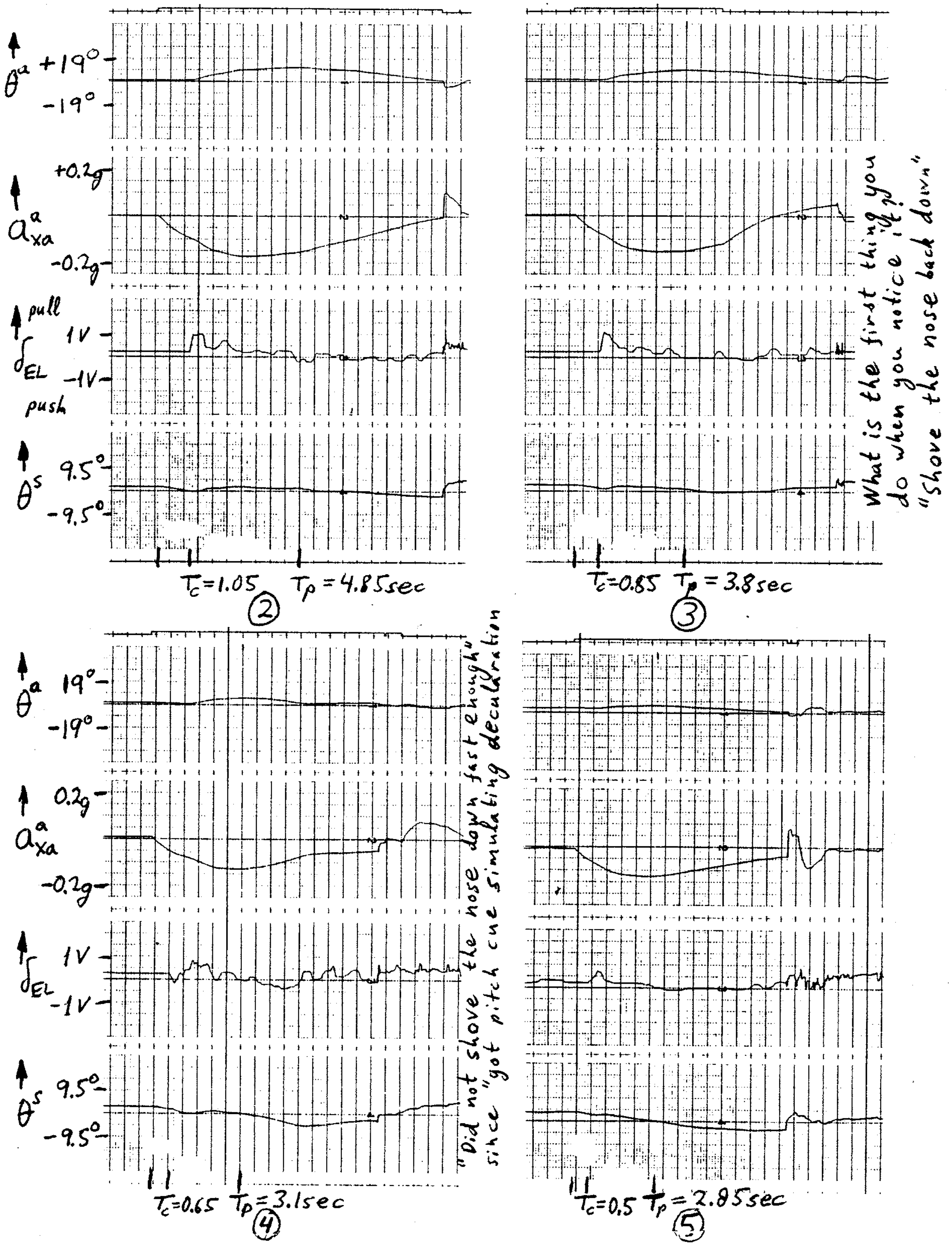


Figure 12. The second to fifth initial runs of the flaps-down experiment with pilot CO. In these runs the optimal washout has $Q_o/Q_s = 1$, the air speed indicator was covered and both the pitch and roll visual display were on. In the first run (not shown here), using No-Motion, the pilot used correct controls i.e. first pushed forward the elevator control δ_{EL}. Positive δ_{EL} is pull nose up.

the pilot immediately realized (0.3 *seconds*) that he is doing the "wrong thing" and went back to the wrong control direction as in the other runs shown. After run 3 the experimenter asked the pilot to describe what was his first control reaction when he noticed the flaps-down transition. The pilot answer was "shove the nose back down" i.e. exactly the opposite of what he was really doing. This question-answer is probably the reason for the very brief initial correct response of the pilot in run 4. I explain this as: following the initial correct response the pilot immediately realized his "mistake" and followed it by an incorrect elevator control as done before. After run 4 the pilot commented that he "did not shove the nose down fast enough" since "got pitch cue simulating deceleration", i.e. again he was not really aware of what his control was. It should be noticed that during the course of these four runs the time of the initial control T_c and the time T_p reduced progressively by a factor of 2.

Pilot EA

Pilot EA was given an explanation of the effects the the optimal washout system and the "ballooning" of the Cessna 150/152 on flaps-down transition. Furthermore he saw a half hour demonstration of these effects using two of his friends, that are airline pilots, which flew the GAT-1. He even saw the other pilots experimental records which showed how the simulation of deceleration can fool a pilot to give a wrong control. The pilot was thus instructed to rely on his instruments. Pilot EA was trained for 13 *minutes* before any flaps-down experiments were run. The first run, not shown, is with no pitch motion and the pilot made the correct control—so he knows what is supposed to do.

Figure 13 shows the pilot responses in the $3^{\underline{rd}}$ and $4^{\underline{th}}$ runs of the flaps-down experiments with a an optimal washout with $Q_o/Q_s = 1$ and with visual pitch and roll display on. The initial incorrect control can be clearly seen in run 3 1.3 *seconds* after the flaps-down transition. In run 4 this initial incorrect response is missing and the response is in the correct direction but after 2 *seconds*. After run 3 the pilot was asked what was the control he applied and his answer was "unless you explained before (I started the experiment) I would swear that I pushed the nose down". Thus I would explain that EA's correct response in run 4 is due to a response to

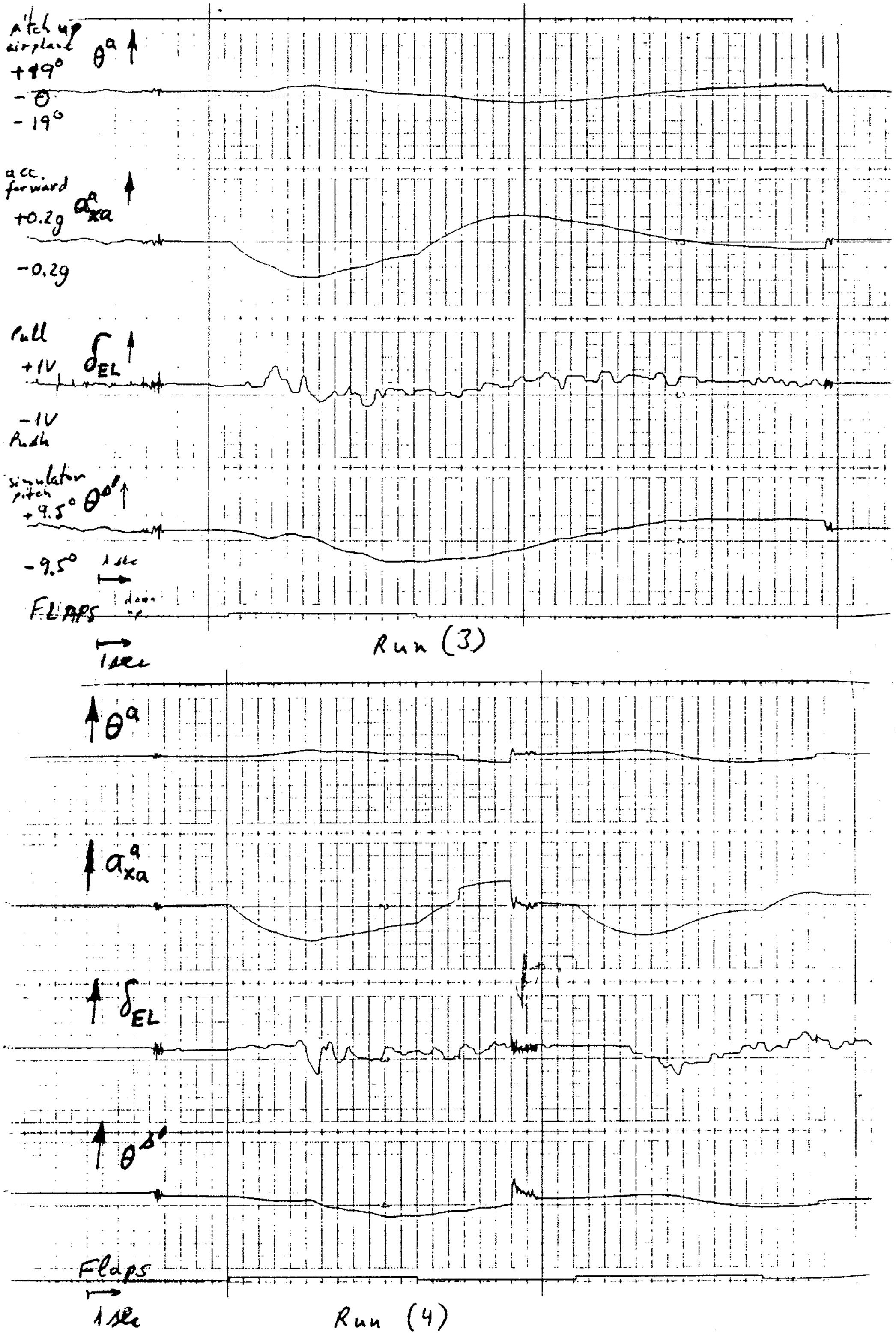


Figure 13. $3^{\underline{rd}}$ and $4^{\underline{th}}$ runs of flaps-down experiment with pilot EA (has $12,000\,hours$), using an optimal washout with $Q_a/Q_s = 1$, visual pitch and roll display were on.

the flight instruments and he is suppressing his response to the motion. In contrast after training other pilots used the motion to there advantage (rather then suppress the reaction to it).

Conclusions

1. The motion has a strong influence on the simulation results.
2. The motion influence cannot be negated initially by verbal explanations. A pilot can be easily trained (within 10 runs) to use the motion correctly.
3. The pilot is not always aware of what his controls are and gives false reports that describe what he thought he should have done.
4. It is not clear if the pilot's reaction is to the incorrect rotation cue (pitch down) or the correctly simulated deceleration cue. To resolve this question, one needs a simulator with linear motion capability and/or a comparison test in a real airplane. A preliminary test done by pilot DM seem to indicate that the deceleration effect in a Cessna 150/152 is smaller (the flaps come down 2 to 3 times slower) but it still causes a pitch down sensation to the pilot.

3.3.2. After Training Results

A sample of 8 runs of pilot JH are shown in Figures 14–16. Figures 14–15 demonstrate how consistent the responses are. Comparing Figures 14–15 to Figure 16 shows the changes in the pilot response due to a change of optimal washout parameter $Q_o/Q_s = 0.32$ to $Q_o/Q_s = 1.0$. Note that this pilot had several hours of preceding training while doing experiments 2—washout detection (included flaps-down training) and experiment 4—take-off. Pilot JH had the most consistent results in these experiments; probably because he had the most flight hours ($2,800\,hours$). The best performance was obtained from pilot IM who has only $150\,hours$; possibly due to his aerobatic experience. The experimental results of parameters T_d, T_c, T_p and ΔH for each run are shown in Figures 17–22 (data in Appendix I and [Ish-Shalom82]). One should note that the data for ΔH is based on the pilot's report, where this report is not his main task. Furthermore he was highly motivated to do well, based on the ΔH performance measure and thus the data may be biased. It should be noted that Figures 17–22 are not conventional in the sense that their ordinate is doubly sorted; first by washout and second by run number. This enables one to see the effect of training and that of the order of washout presentation. Looking at ΔH in Figures 18, 20 and 22 one can see the training effect where the response times T_d, T_c, T_p and the performance measure ΔH decrease. It seems that the main effect is over after the first 5 runs. After a change in washout there seems to be a second training effect that is over within one or two runs. The washout changes during the experiment were designed to reduce this second training by making only changes between "close" washouts during the succession of the experiment. Thus we needed to reject less data points in the computation of the averages for each washout.

Intuitively the times T_d and T_c for each run should be the same since the time of first control T_c, after proper training, should also represent the time of flaps-down detection. Looking at Figures 17, 19 and 21 one can see that the results for T_d are more consistent than the results for T_c. Thus one can assume that the flaps-down detection decision indicated by T_d is "more filtered" then that for T_c. It is surprising to find that in general T_d is shorter then T_c nevertheless.

The average and standard deviation of T_d, T_c, T_p and ΔH for each washout tested excluding the two training effect and "far out points" are shown in Figures 23–25. Looking at the averages, $T_c > T_d$ for pilot JW, $T_c \geq T_d$ for pilot IM where $T_c = T_d$ for No-Motion. For pilot JH $T_c < T_d$ significantly ($\rho = 0.005$) for the Link-like washout, but changes to what we had for the other two pilot i.e. $T_c > T_d$ significantly ($\rho < 0.014$) for the optimal washouts with $Q_o/Q_s \geq 0.32$. For these washouts the time $T_c - T_d \approx 0.15\, \mathit{second}$ for both JH and IM. For JW this time difference is much larger $\approx 0.5\, \mathit{second}$ and is probably the reason for his relatively poor performance. The following conclusions are made from the relation between T_d and T_c:

(i) One of the following is true: (a) The pilot cannot handle both tasks at once and does them sequentially i.e. first the detection and then the control (both are motor reactions and thus there is no reason to believe initial that there is a difference between them). (b) The pilot was not trained or did not "want" to rely to much on the motion and thus delayed his control after his detection. It is shown in Table 2 that the changes in simulator motion do have a significant effect on the pilot recorded statistics.

(ii) From JH results it seems that the motion aided his decision when the flaps were put down since with the Link $T_c < T_d$.

Another related fact is that the time $T_p - T_c$ indicates the difference between an abrupt or gradual (larger time difference) pilot control. For JH $T_p - T_c$ decreases as Q_o/Q_s increases (more motion) by a factor of 2. A likely explanation is that the pilot was more confident in his controls as he had more motion (Q_o/Q_s increases). On the other hand for IM $T_p - T_c$ did not change much with the motion. For JW one cannot make conclusions since the standard deviation of the measurements is too large.

Table 2 shows the significance level of the difference between two "neighbor" washouts using a T-test. The highest significance (lowest ρ) is obtained for pilot JH. In general within a 0.05 level or better there are differences between the washouts using 3 out of the 4 statistics (T_d, T_c and T_p). It is not too surprising to see that the performance does not show significant dependence on the washout used. It is expected that with better pilots and more training the performance dependence on the washout would be even smaller. T_p also shows a smaller dependence on the washout then T_d and T_c for pilot JW and is much more variable then the other two times for all pilots.

Table 3 shows the significance level of the difference between every pair of pilots using a T-test for each statistic and washout tested individually. Combining our four statistics we see that only 3 pilot pairs for a given washout are not significantly different at a $\rho = 0.015$ level; specifically: JH-JW Link, IM-JW Link, JH-IM $Q_o/Q_s = 0.32$. In all except one case the most significant difference between pilots is in one of the performance measures ΔH or T_p. This indicates that training experience and the quality of a pilot is manifested through the performance measures, ΔH, and T_p, while T_d and T_c are more directly related to the inertial sensory input through "less trainable" paths. It is also evident from Table 3 that there is a significant difference between No-Motion and Link (at $\rho < 0.05$). Furthermore there is a significant difference between the Link and the optimal washouts for both pilots JH and IM (at $\rho < 0.01$). What is surprising is that ΔH for the Link is significantly smaller (at $\rho = 0.01$) then that for any of the optimal washouts for pilot IM—although T_d and T_c are significantly ($\rho = 0.01$) higher. This fact can be at least partially accounted to pilot training, since the Link runs were performed in the last part of the experiment.

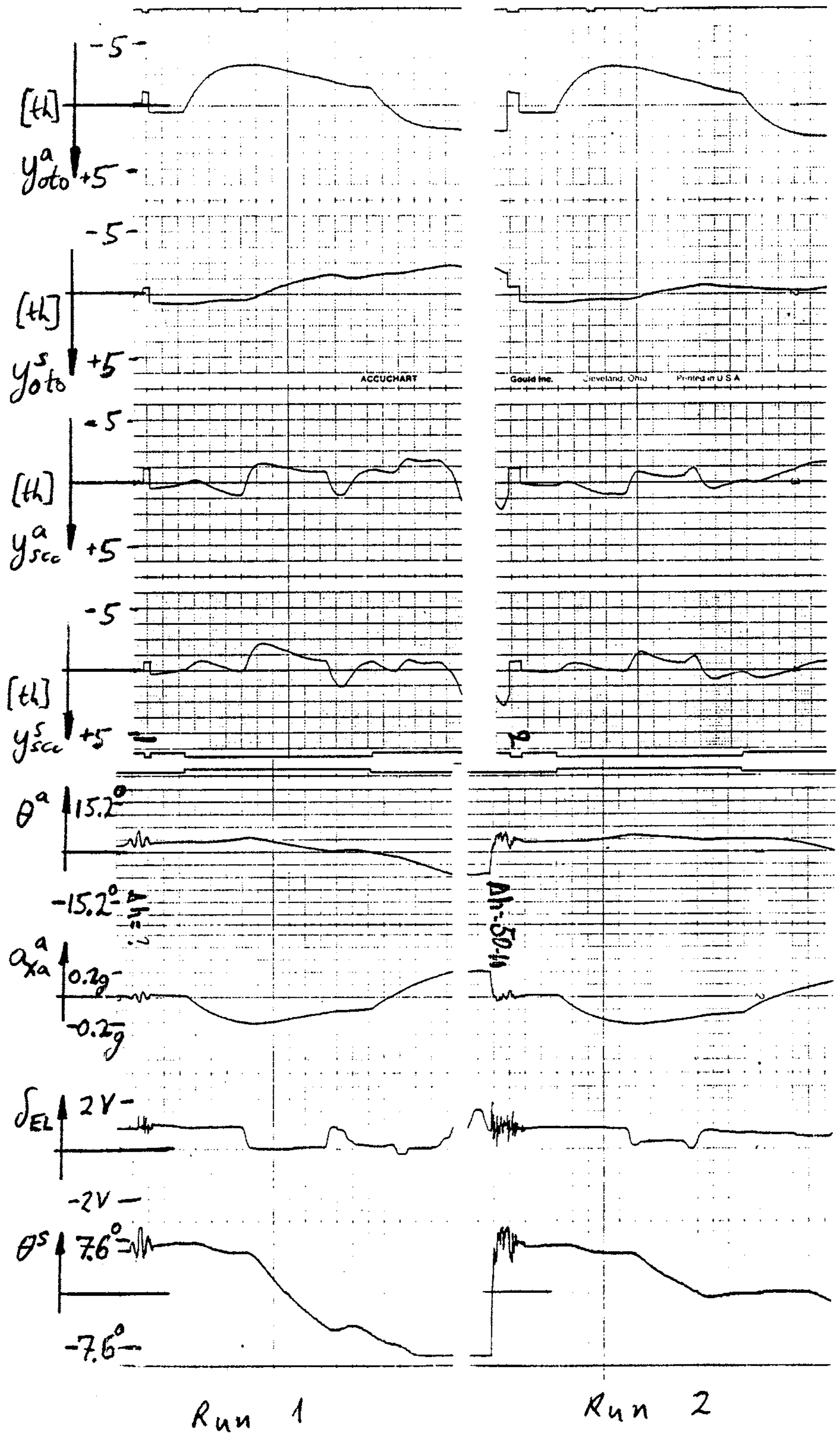


Figure 14. Flaps-down experiment runs 1 and 2 of pilot JH for $Q_o/Q_s = 0.32$. Engine $2,300\,rpm$ i.e. $100\,mph$ before flaps down. Visual pitch and roll strip display was on.

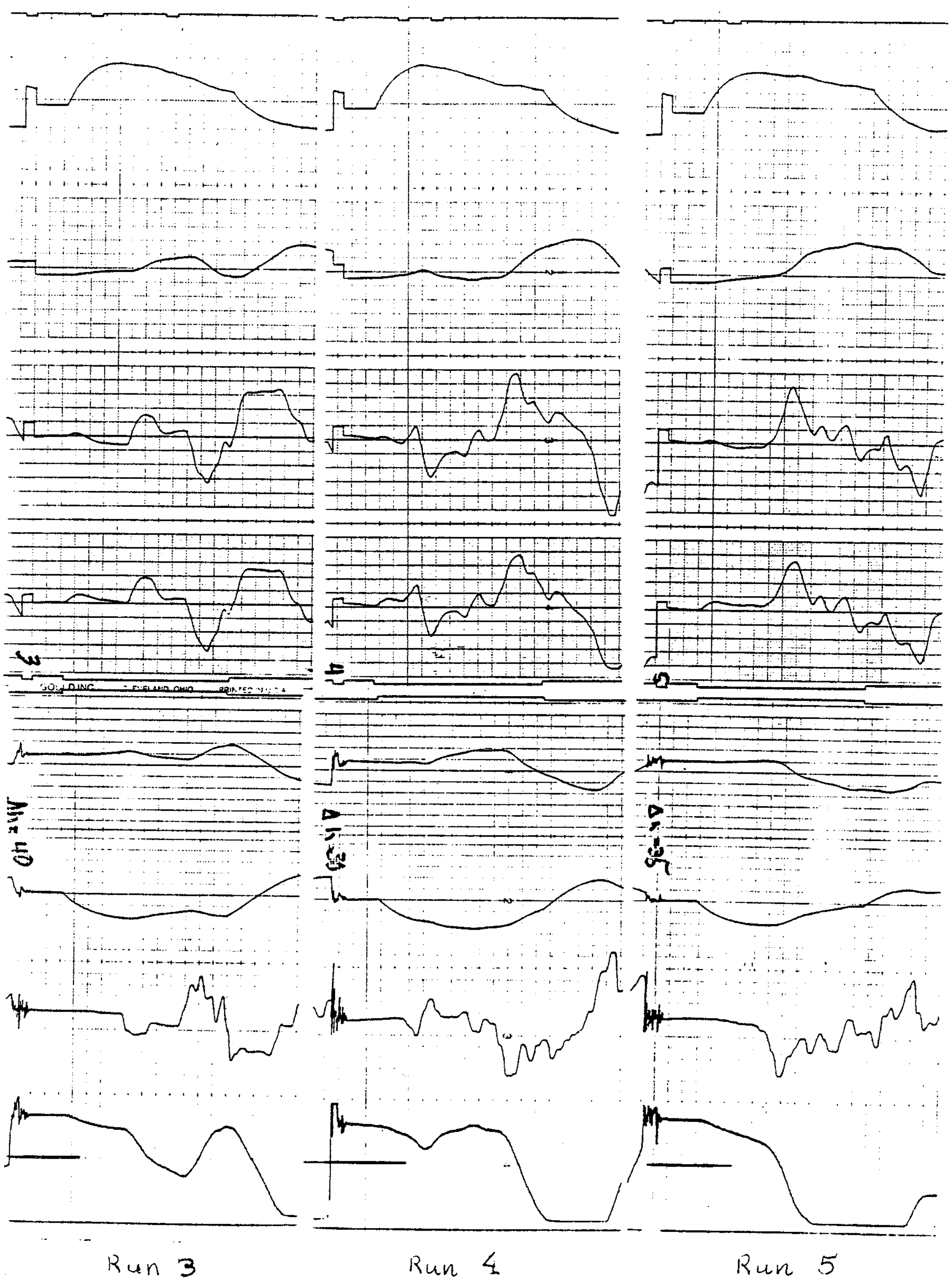


Figure 15. Flaps-down experiment runs 3, 4 and 5 of pilot JH for $Q_o/Q_s = 0.32$. Engine 2,300 *rpm* i.e. 100 *mph* before flaps down. Visual pitch and roll strip display was on.

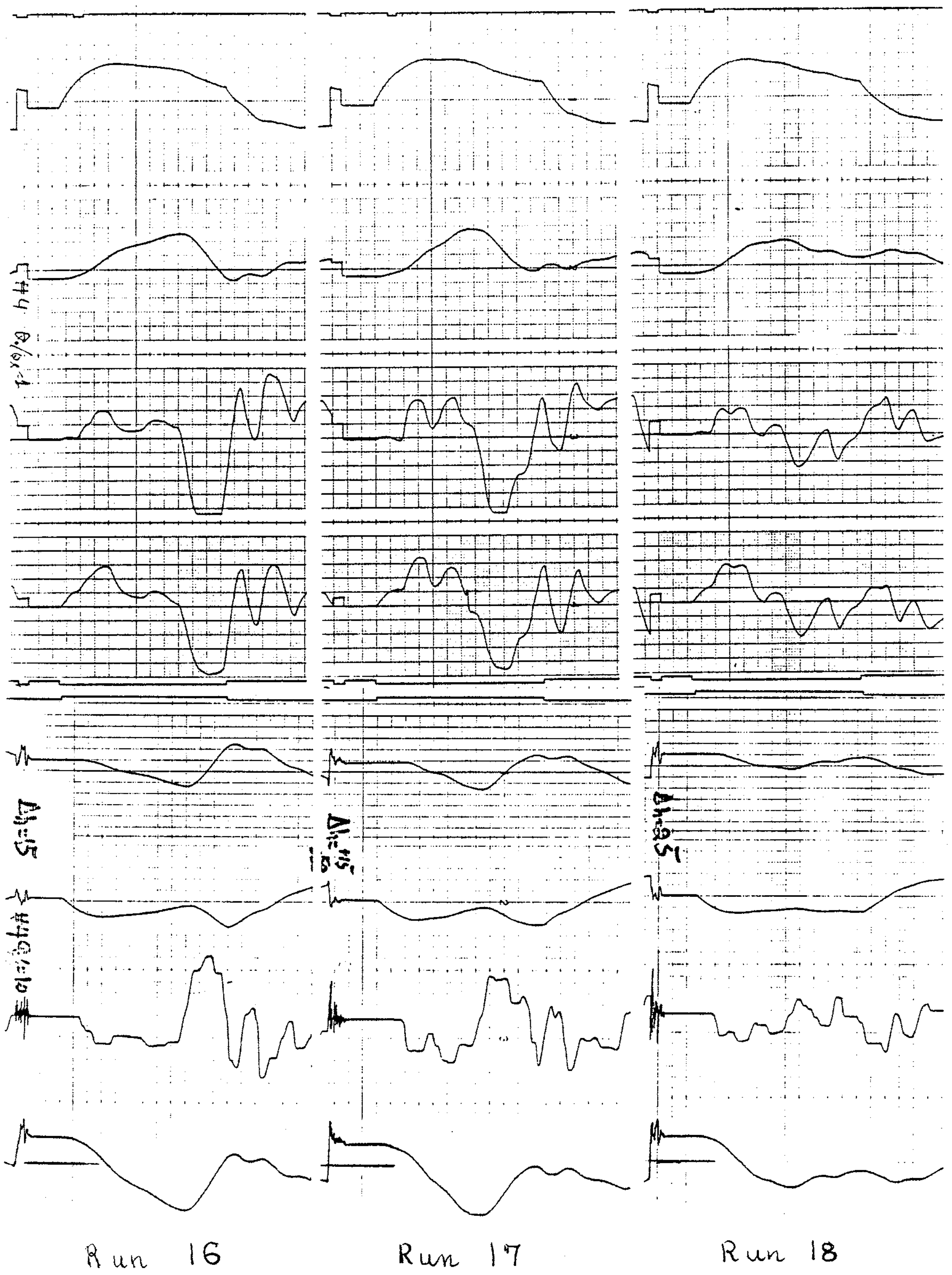


Figure 16. Flaps-down experiment runs 16, 17 and 18 of pilot JH for $Q_o/Q_s = 1.0$. Engine $2,300\,rpm$ i.e. $100\,mph$ before flaps down. Visual pitch and roll strip display was on.

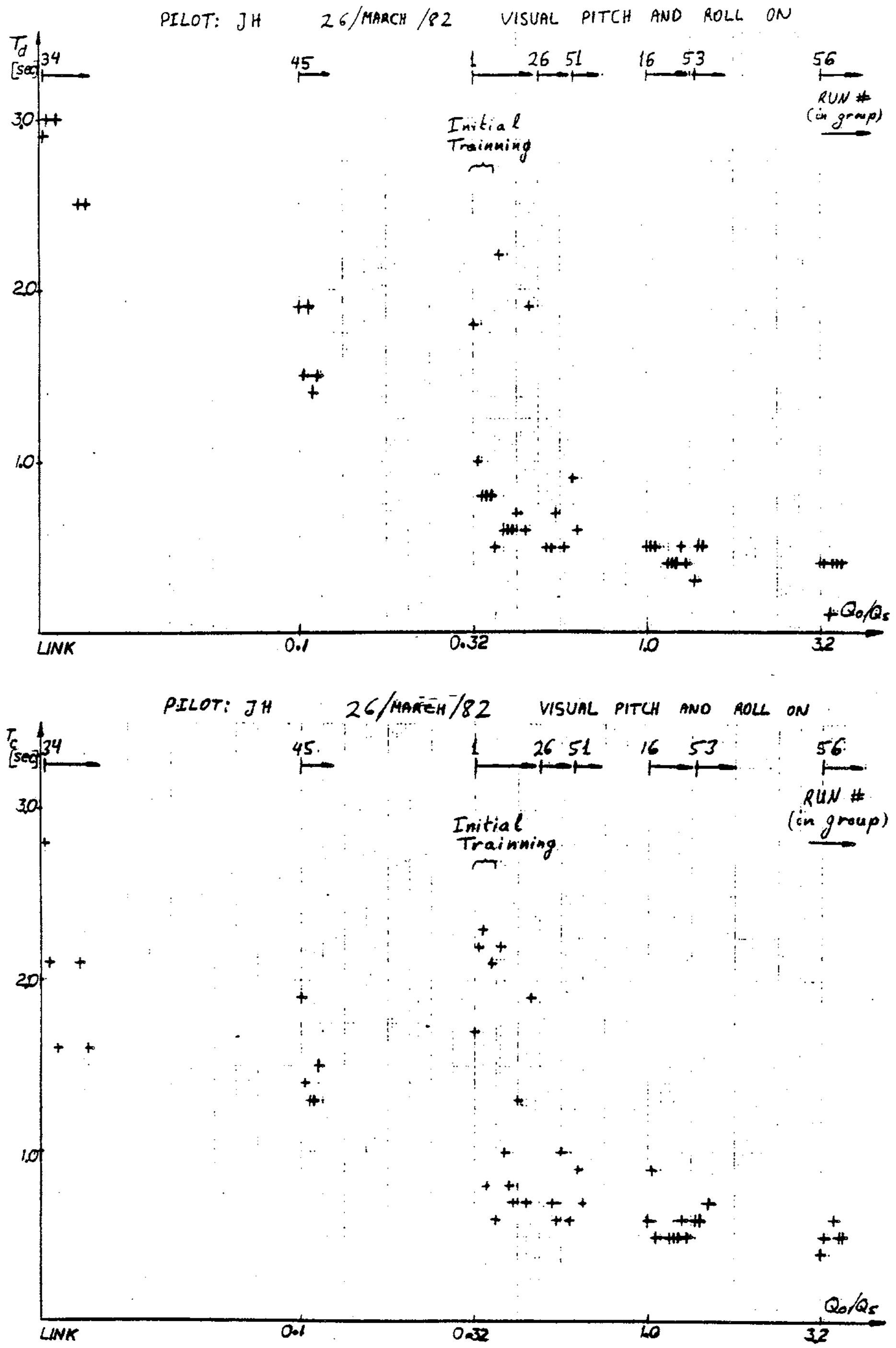


Figure 17. Pilot JH *detection time* T_d and the *first control time* T_c as functions of both the washout and the run number. This three dimension plot is shown in two dimensions by use of an unusual doubly sorted ordinate; first by washout and second by run number.

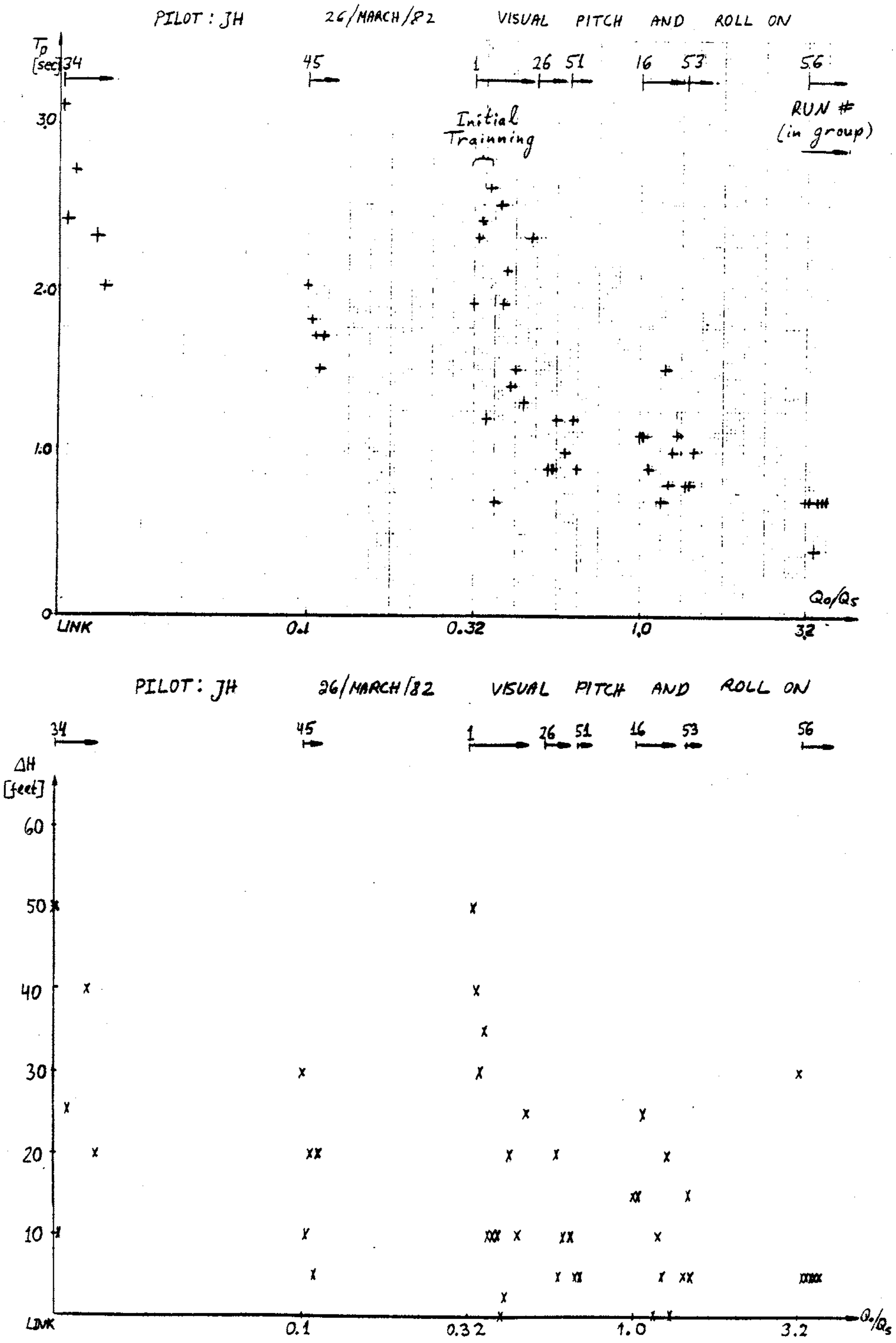


Figure 18. Pilot JH *airplane pitch peak time* T_p and maximum *"ballooning"* (increase in altitude) ΔH as functions of both the washout and the run number. This three dimension plot is shown in two dimensions by use of an unusual doubly sorted ordinate; first by washout and second by run number.

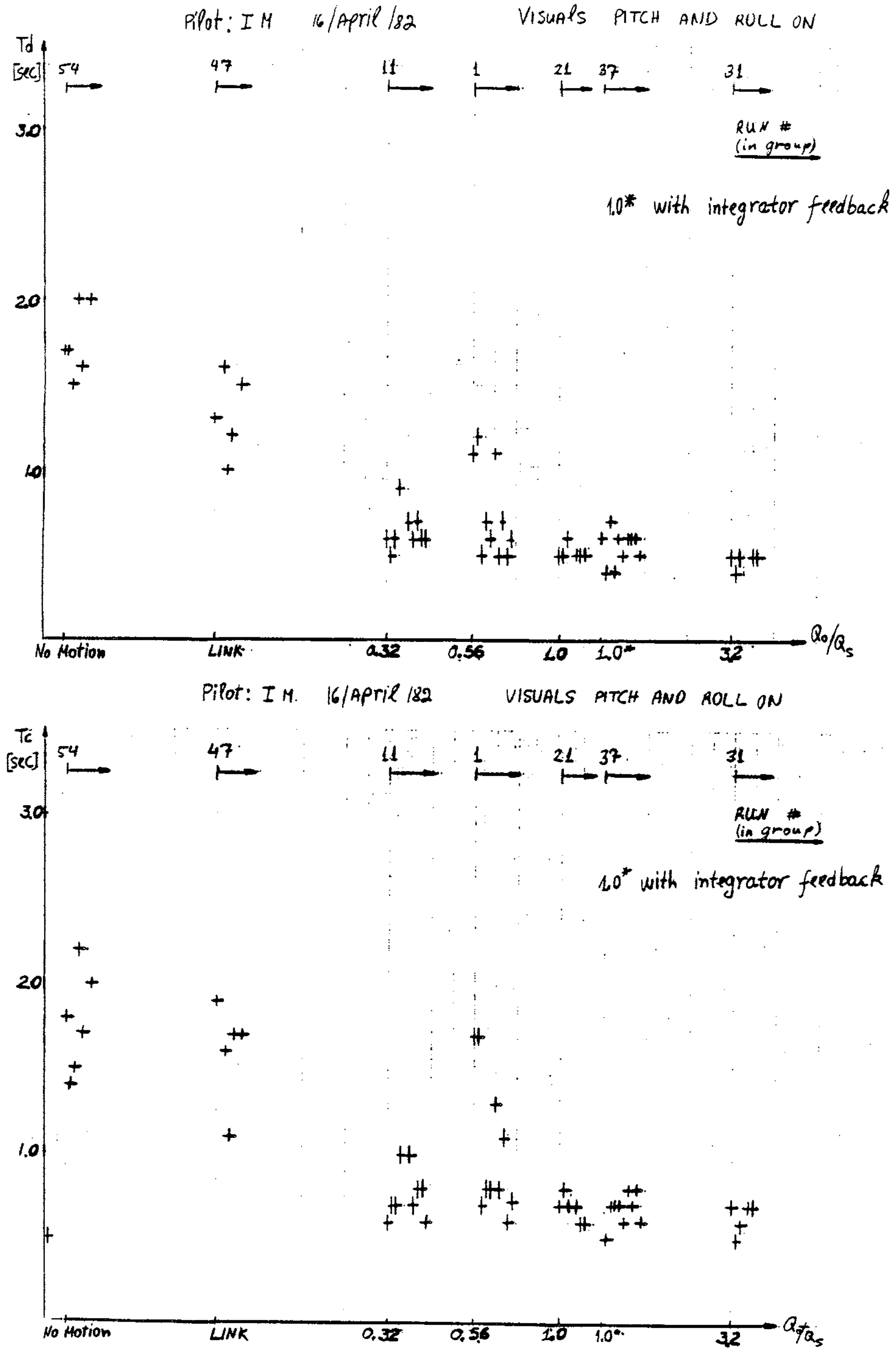


Figure 19. Pilot IM *detection time* T_d and the *first control time* T_c as functions of both the washout and the run number. This three dimension plot is shown in two dimensions by use of an unusual doubly sorted ordinate; first by washout and second by run number.

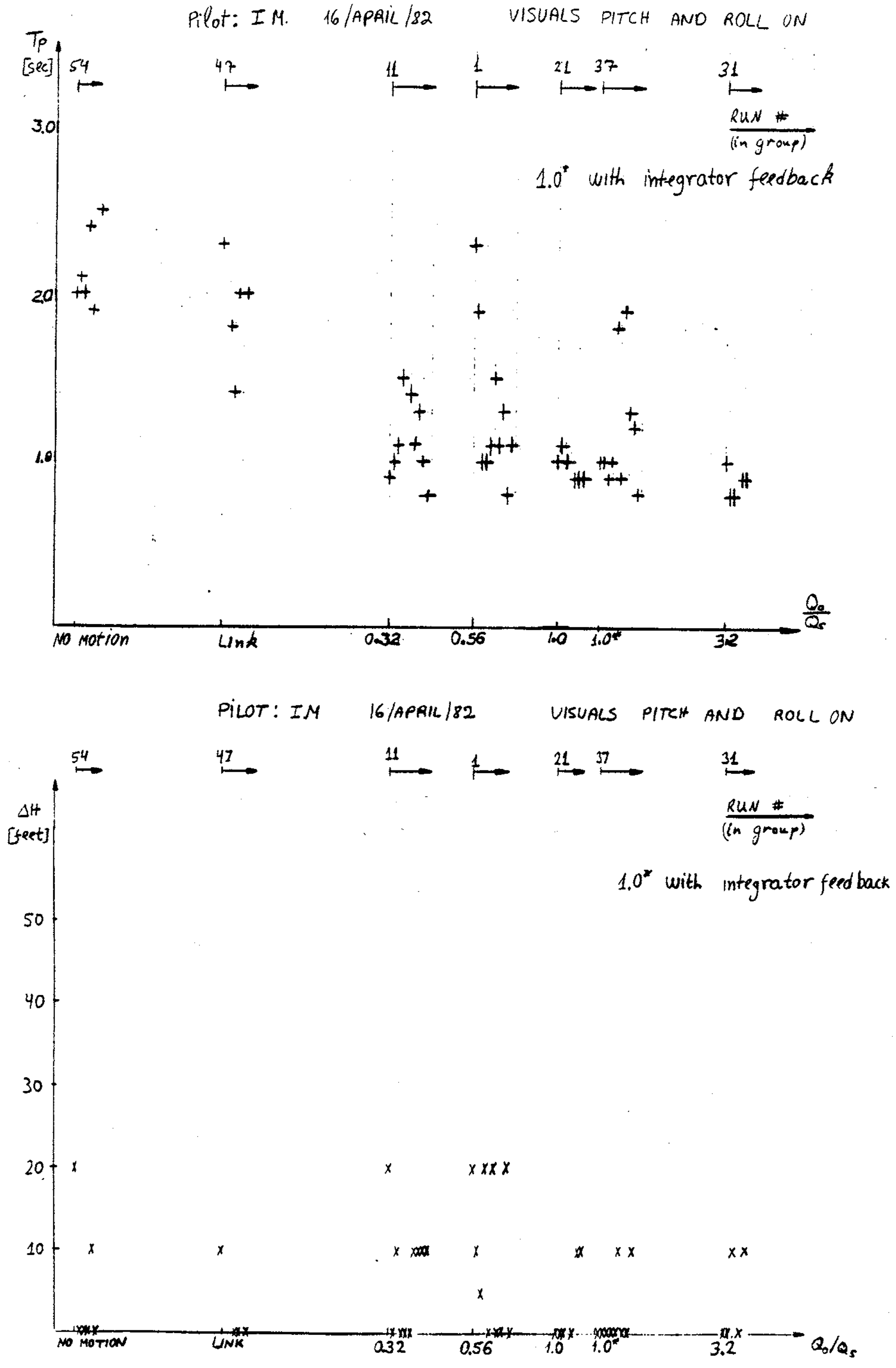


Figure 20. Pilot IM *airplane pitch peak time* T_p and maximum *"ballooning"* (increase in altitude) ΔH as functions of both the washout and the run number. This three dimension plot is shown in two dimensions by use of an unusual doubly sorted ordinate; first by washout and second by run number.

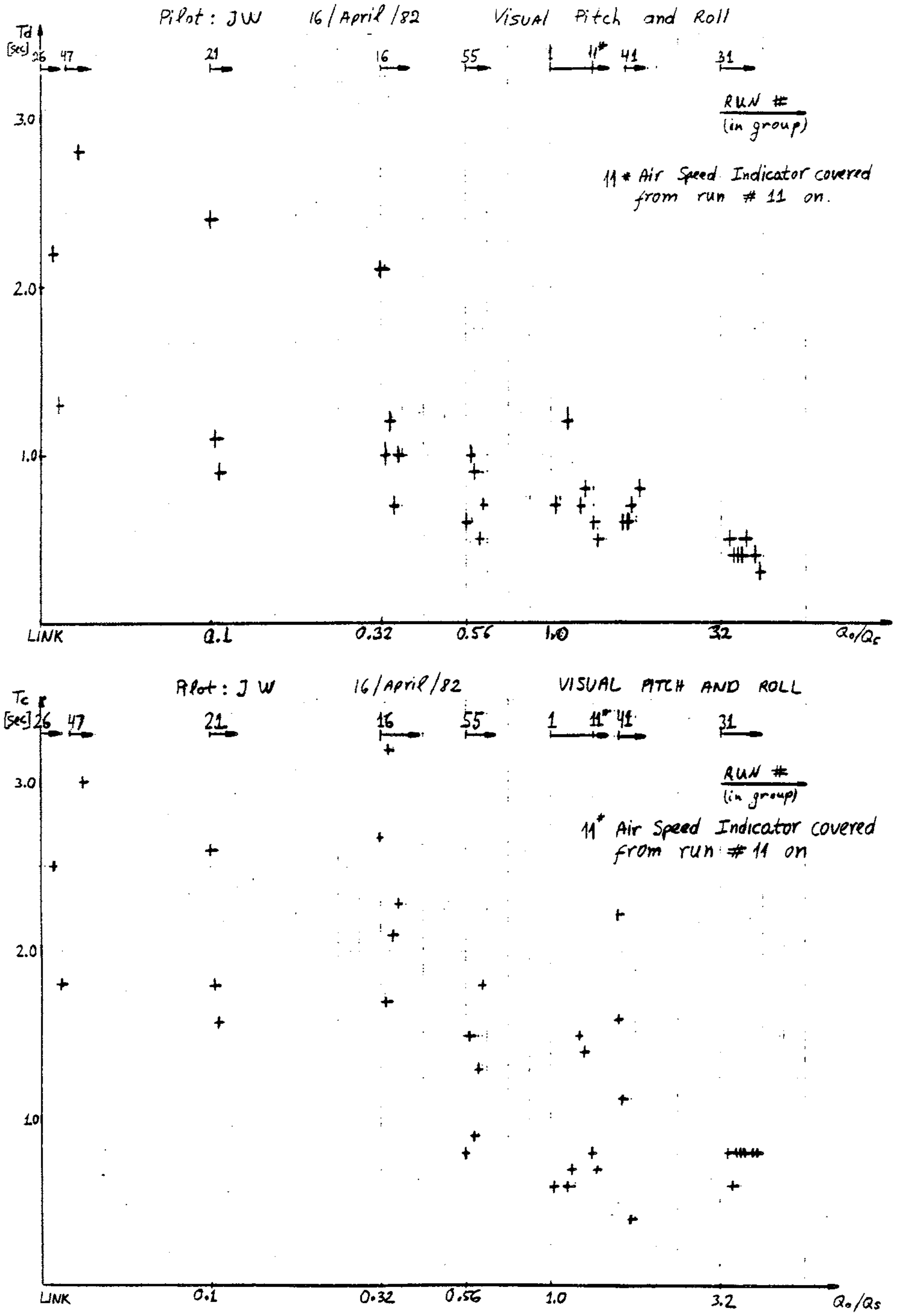


Figure 21. Pilot JW *detection time* T_d and the *first control time* T_c as functions of both the washout and the run number. This three dimension plot is shown in two dimensions by use of an unusual doubly sorted ordinate; first by washout and second by run number.

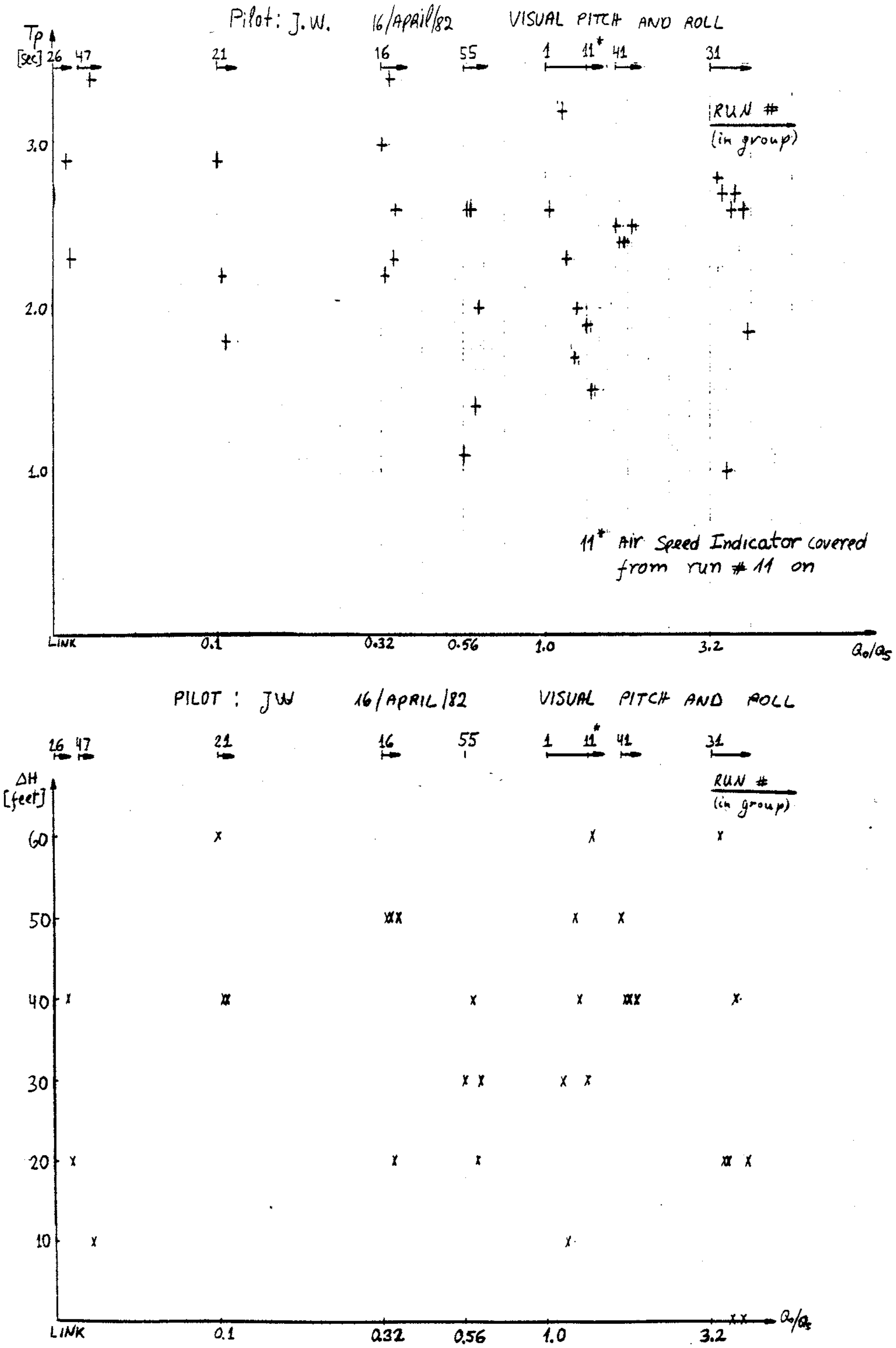


Figure 22. Pilot JW *airplane pitch peak time* T_p and maximum *"ballooning"* (increase in altitude) ΔH as functions of both the washout and the run number. This three dimension plot is shown in two dimensions by use of an unusual doubly sorted ordinate; first by washout and second by run number.

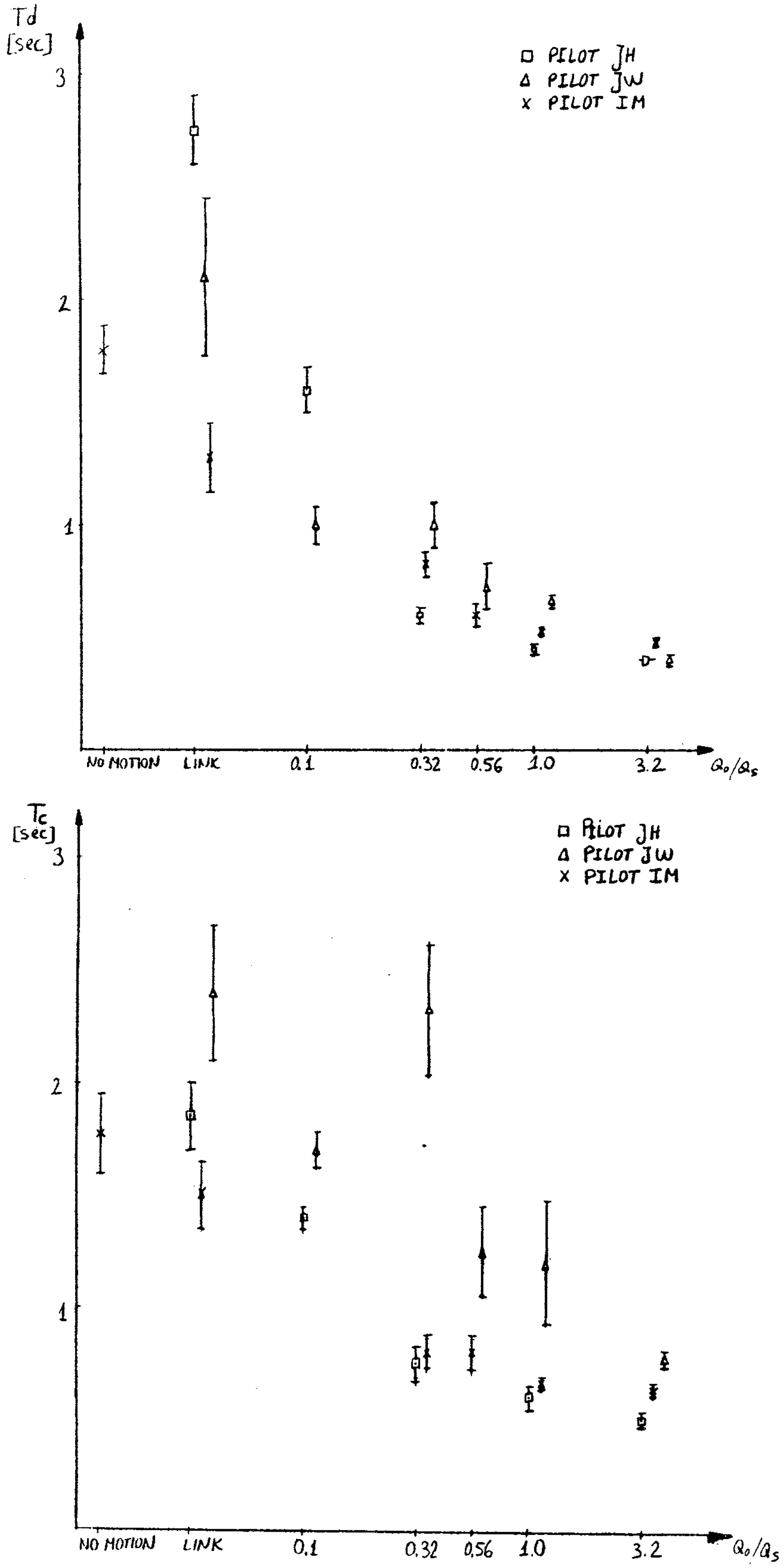


Figure 23. Average results of the three pilots JH, IM and JW for the *detection time* T_d and the *first control time* T_c as functions of washout used.

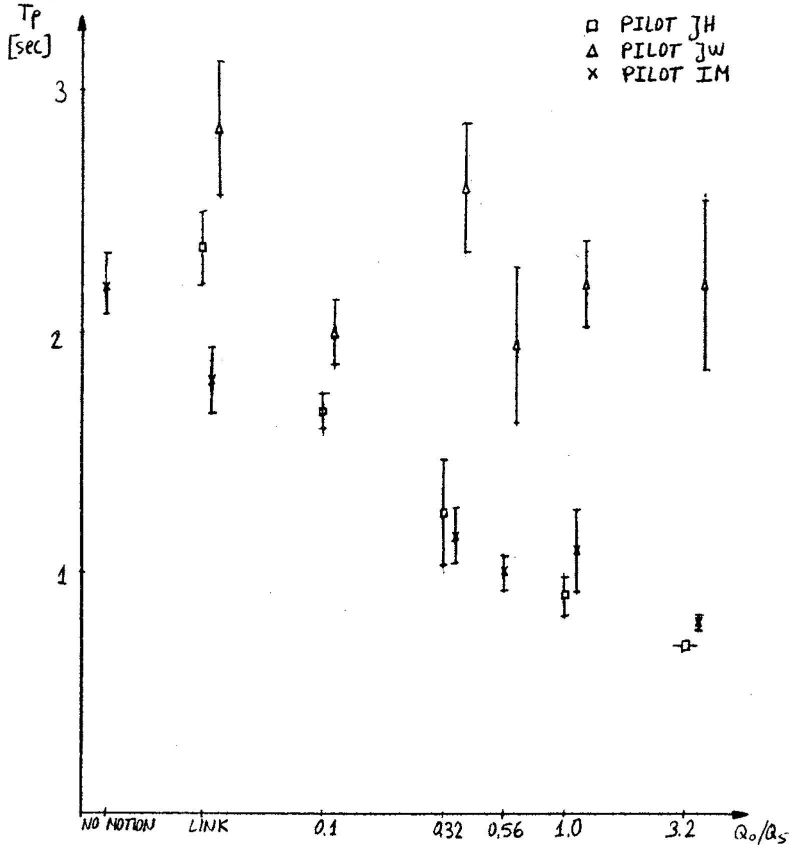


Figure 24. Average results of the three pilots JH, IM and JW for the *airplane pitch peak time* T_p as a function of washout used.

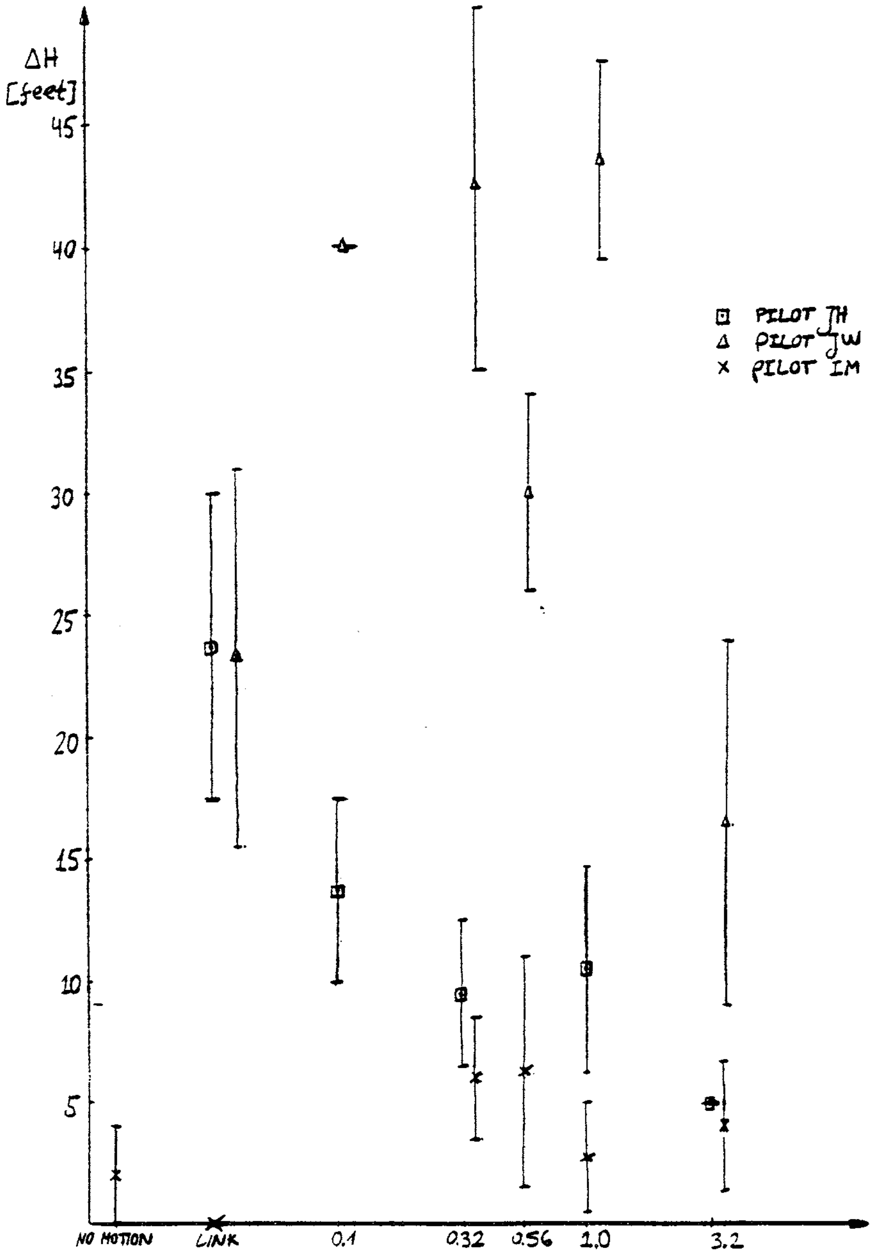


Figure 25. Average results of the three pilots JH, IM and JW for the maximum *"ballooning"* (increase in altitude) ΔH as functions of washout used.

Pilot	Statistic	No motion	Link	0.1	0.32	0.56	1.0	3.2
JH	Td			0.0007	0.003	← 0.0008 →		0.05
	Tc			0.04	0.00001	← 0.02 →		0.15
	Tp			0.01	0.01	← 0.04 →		0.0009
	ΔH			0.23	0.34	← 0.70 →		0.08
IM	Td		0.048	← 0.01 →		0.30	0.16	0.09
	Tc		0.31	← 0.009 →		1	0.13	0.42
	Tp		0.08	← 0.01 →		0.40	0.60	0.02
	ΔH		0.04	← 0.01 →		0.97	0.35	0.64
JW	Td			0.13	0.90	0.1	0.70	0.00003
	Tc			0.20	0.13	0.03	0.80	0.046
	Tp			0.10	0.13	0.13	0.50	1
	ΔH			0.20	0.76	0.20	0.03	0.005

Table 2: Significance level ρ of T-test when comparing neighboring washouts for a given pilot and statistic.

Pilot Pair	Statistic	Link	0.1	0.32	0.56	1.0	3.2
JH-IM	Td	0.0004		*0.18		0.015	0.01
	Tc	0.17		1		0.09	0.06
	Tp	0.03		0.53		0.08	0.007
	ΔH	0.03		0.25		0.02	0.70
JH-JW	Td	*0.30	0.03	0.03		0.0001	1
	Tc	0.20	0.10	0.017		0.01	0.0004
	Tp	0.20	0.40	0.006		0.000001	0.004
	ΔH	0.97	0.006	0.02		0.000001	0.12
IM-JW	Td	*0.20		0.04	0.20	0.0004	0.03
	Tc	0.10		0.017	0.06	0.02	0.04
	Tp	0.05		0.006	0.05	0.000001	0.007
	ΔH	0.10		0.018	0.003	0.000001	0.10

Table 3: Significance level of ρ of T-test when comparing pilots. All columns except the *rd ones are significantly different at a ρ=0.15% or better.

3.4. Experiment 4—Take-off

Seven take-offs by pilot IM are shown as an example of the general results obtained for all pilots tested, Figures 26–29. These traces show the pilot elevator control $\delta_{EL}(t)$, the simulated airplane pitch angle $\theta^a(t)$ and surge linear acceleration $a^a_{ax}(t)$, the motion-base pitch angle $\theta^{s\prime}(t)$ and both the reference and the simulator pilot vestibular model outputs $y^a_{oto}(t)$, $y^s_{oto}(t)$, $y^a_{scc}(t)$, $y^s_{scc}(t)$ in *threshold units*. The runs shown are after several hours of training and are not the first runs in this experimental session but runs 11–19. In these experiments the order of presentation and training effect seem to be a substantial factor in determining the pilot response. That is why the analysis is very qualitative and the results are given through examples. Furthermore, the most interesting results are from the results of training and transitions between washouts. The results shown in Figures 26–29 are for eight different washouts, where the order of presentation was chosen as to minimize the effect of transition between the washouts. The order used is from No-Motion to increasing motion by passing first through the Link-like washout ($\theta^s = \theta^a/2$) and then using an optimal washout with a succession of increasing values of Q_o/Q_s from 0.1 to 10 i.e. a range of $10 = \sqrt{100}$ fold in the ratio of the otolith to semicircular canal threshold. Run 18 in Figure 29 is with $Q_o/Q_s = 32.$ but it is very similar to the result of Run 17 with $Q_o/Q_s = 10.$ and thus not considered to extend the above Q_o/Q_s parameter range. This range seems to capture the whole range of noticeable changes in the optimal washout designed simulator motion. This fact is important since it shows that the choice of ratio of threshold units between the otolith and semicircular canals together with the initial equal weighting of the these two errors is correct within at least a factor of 10, probably within a factor of two as evident from the results and the pilot comments. Furthermore, this suggests that design is sensitive to changes in this ratio parameter and the maximum range one needs to check is 10.

In order to describe the results in Figures 26–29 let us define the following terminology:

(i) t_t—the point of take-off; it is considered when θ^a first increases from zero.

(ii) θ^a_{max}—the take-off maximum pitch angle; it is considered the maximum of $\theta^a(t)$ from take-off to 20 *seconds* after.

(iii) $\dot{\theta}^a(t_t)$ is the maximum airplane pitch rate within 3 *seconds* after take-off.

Looking through sequentially, we see that θ^a_{max} or $\dot{\theta}^a(t_t)$ decrease as more motion is provided to the pilot. Initially both pitch rotation motion and the linear acceleration simulation through "g-tilt" increase. From there on the relative amount of rotation motion is decreased and the "g-tilt" effect is increased. Up to $Q_o/Q_s = 1$ this leads to a decrease of the θ^a_{max} or $\dot{\theta}^a(t_t)$. The reason seems to be that the incorrect increase of surge linear acceleration sensed by the pilot during the initial take-off lead the pilot to believe that his nose (pitch angle) is too high and he increases it cautiously in order to avoid a stall during the take-off, which is fatal in reality. In the last two runs (Figure 28) the dominant effect seems to be the decrease of pitch rotation motion which once again causes the pilot to increase θ^a_{max} or $\dot{\theta}^a(t_t)$.

The interpretation above is based also on the following observations that were found consistent for several pilots. If after a few runs using the optimal washout the pilot is tested with the Link-like washout or even better with No-Motion than the pilot initially loses control of the airplane and pitches up and down, pilot CO even crashed into the ground. This happens even if the washout before was close i.e. $Q_o/Q_s = 0.1$ before the Link-like and the Link-like before the No-Motion. Figure 30 shows this effect for two good pilots IM (150 flight hours) and JH (2,800 flight hours).

Figure 31 shows an example of what happened to pilot WH (several thousands of hours flight experience) when the washout was changed from $Q_o/Q_s = 0.032$ to 10. Basically the pilot "got killed", he did not pull his nose up enough and thus crashed back on the runway. This happens almost with every experienced pilot that tries, for the first time, to fly the simulator with an optimal washout with $Q_o/Q_s > 0.32$ and is usually very disappointing to the pilot. The reason seems to be that the incorrect simulation of the surge linear acceleration at take-off gives the pilot the feeling that he is about to stall the airplane on take-off (which would also lead him to crash). As expected from this explanation the student pilot DM (20 flight hours i.e. not very experienced) was tested and he was the only pilot to stall the airplane on take-off; all the other pilots tested had over 100 hours flight experience.

It is clear that in order for the optimal washout to be satisfactory the motion-base pitch angle at take-off (at t_t) should be small. This can be achieved by adding an integral feedback and the "deterministic washout" shown in Figure 2 and which was implemented on the GAT-1 (see also example in chapter VI.2). The deterministic washout used in this implementation is very simple: $\theta^s(t) = \mu\theta^a(t)$ for $0 \leq \mu \leq 1$, where we chose $\mu = 1$. This washout still provides initially a surge linear acceleration cue, using "g-tilt", but by the time of take-off the "g-tilt" is reduced to a smaller value which makes the take-off simulation after t_t better (Figure 32).

In general it seems that pilots like best the washout with $Q_o/Q_s = 0.32$. It is believed that using the above optimal washout with the deterministic washout and integrator feedback, the preferred value for Q_o/Q_s would be some what higher, say 1. It is interesting to note that due to the simulation of surge linear acceleration by "g-tilt" the maximum required pitch motion of the simulator shrinks and thus the steady-state gains used by the optimal washout can be larger then those used by the Link washout and thus closer to 1—which corresponds to a one to one simulation of the airplane motions.

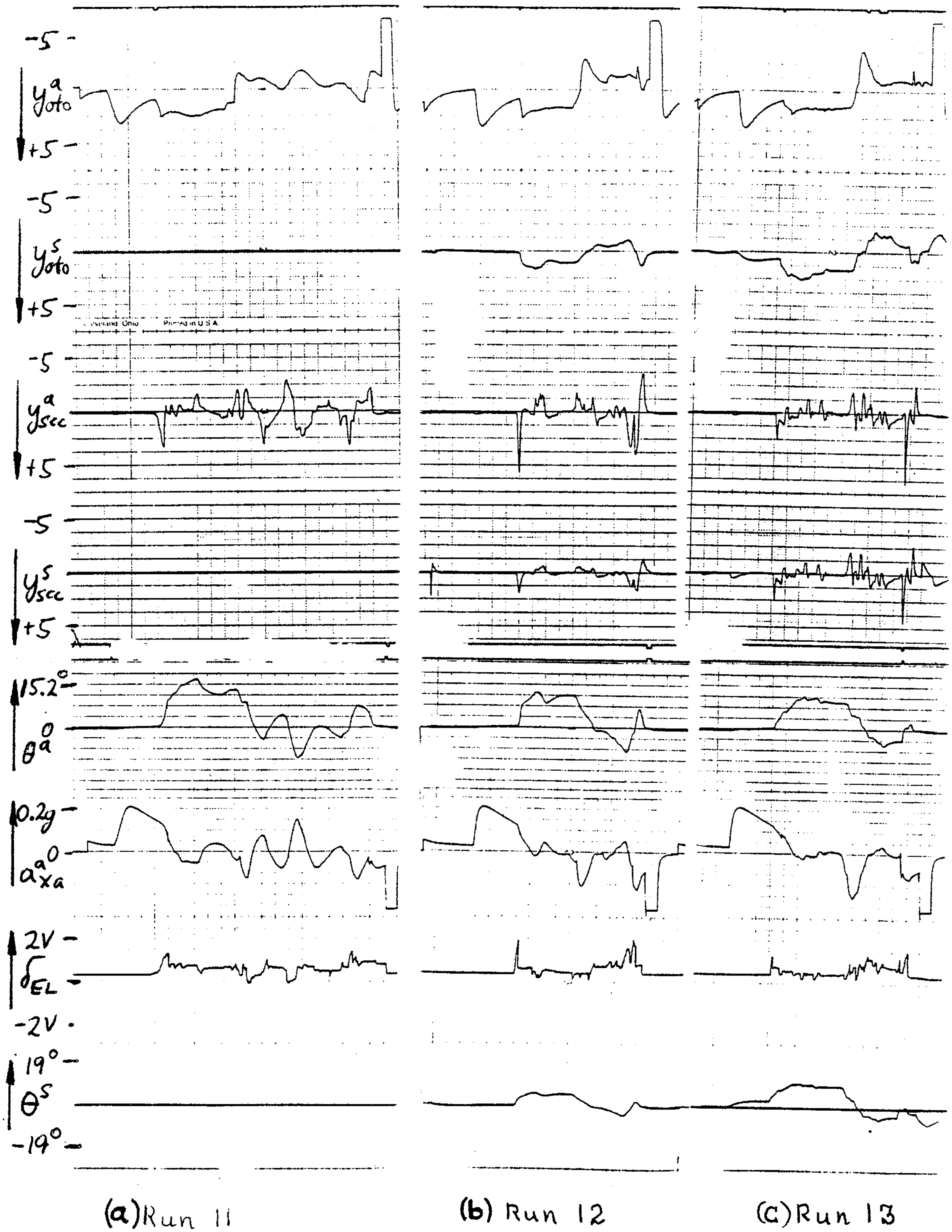
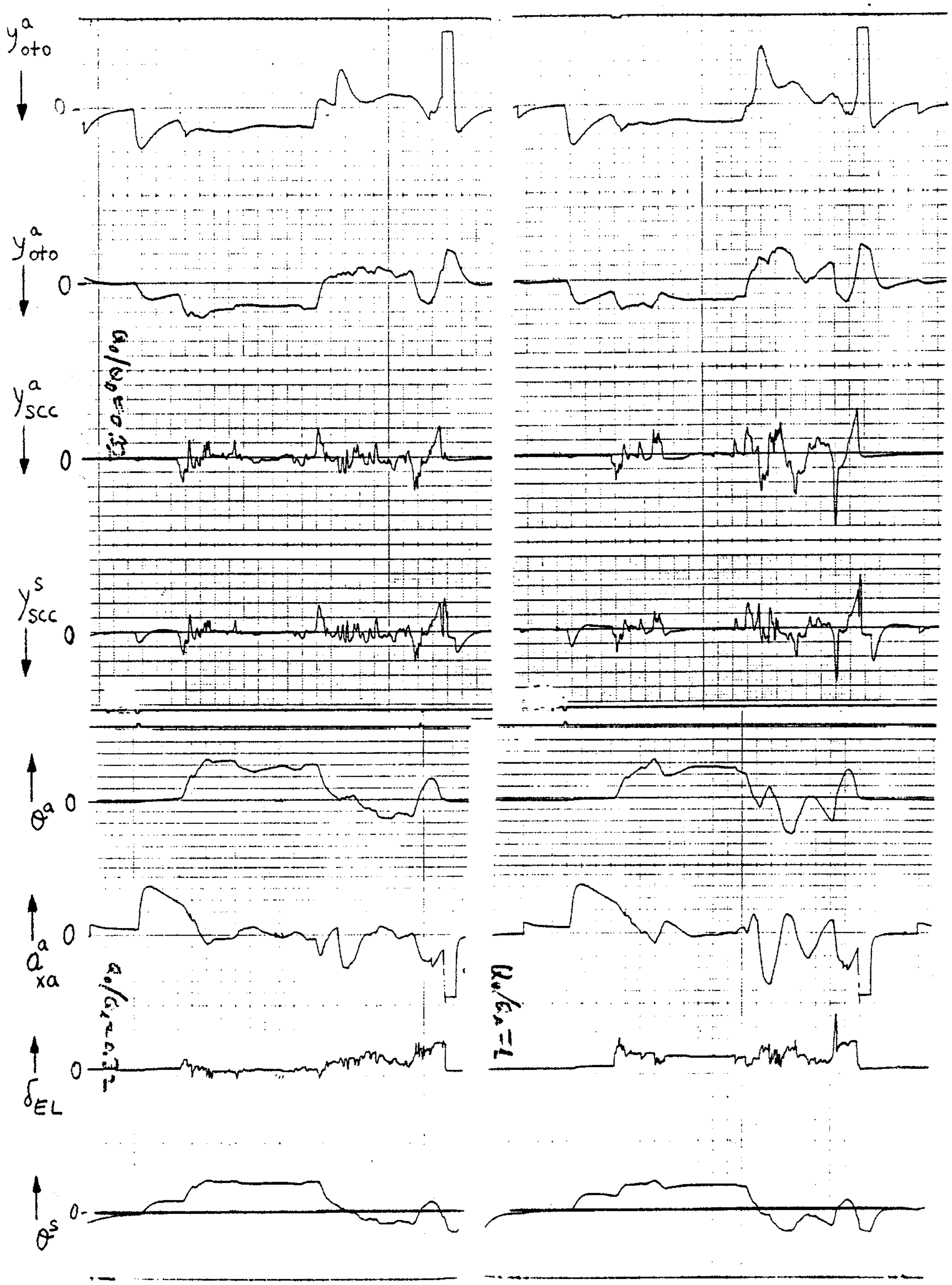


Figure 26. Take-off experimental runs 11–13 of pilot IM with visual pitch and roll display on. The limits of the pitch motion are shown on (a). (a) Run 11, No-Motion. (b) Run 12, Link-like $\theta^s(t) = \theta_a(t)/2$. (c) Run 13, $Q_o/Q_s = 0.1$.



(a) Run 14

(b) Run 15

Figure 27. Take-off experimental runs 14, 15 of pilot IM with visual pitch and roll display on. (a) Run 14, $Q_o/Q_s = 0.32$. (b) Run 15, $Q_o/Q_s = 1.0$.

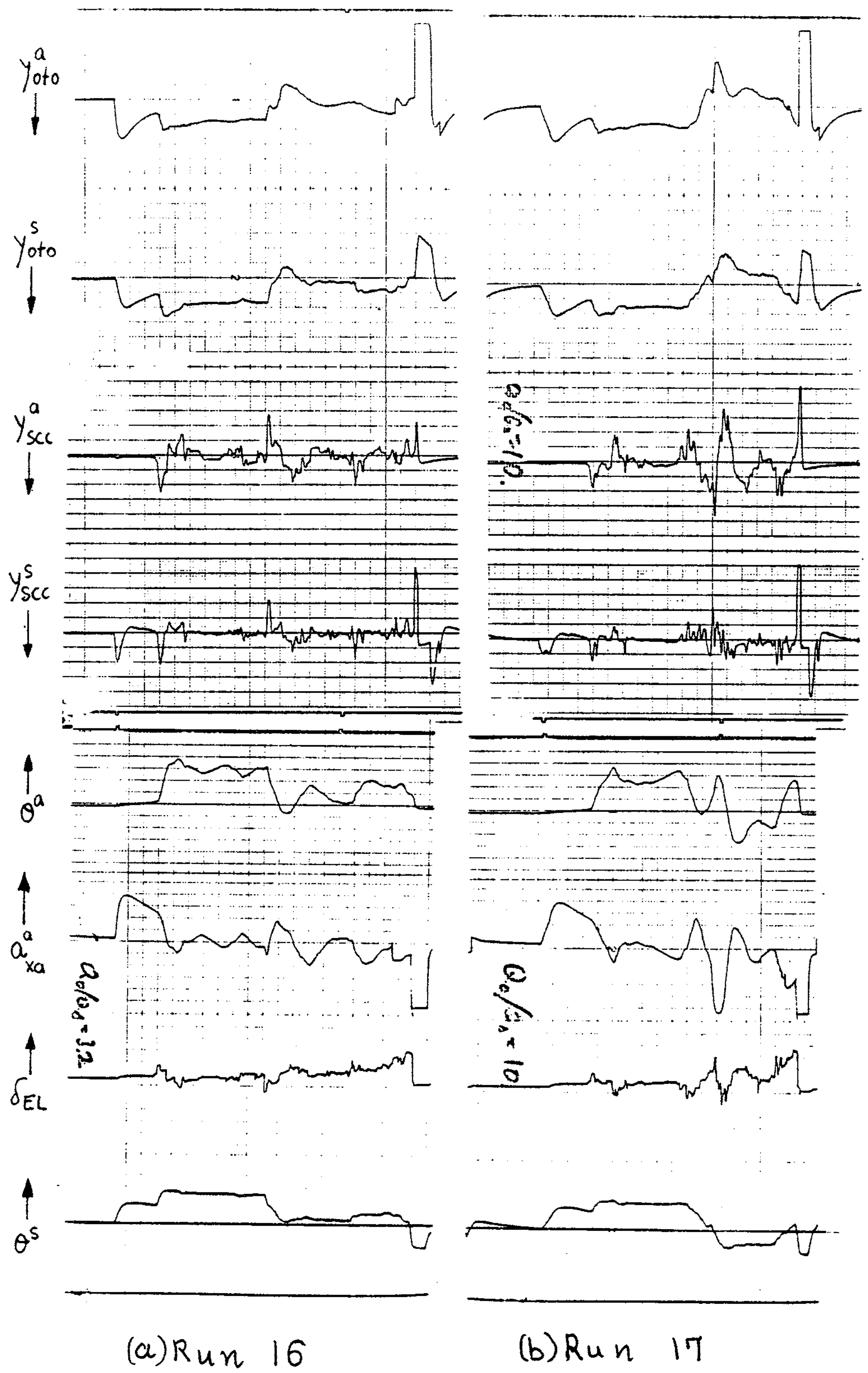


Figure 28. Take-off experimental runs 16, 17 of pilot IM with visual pitch and roll display on. (a) Run 16, $Q_o/Q_s = 3.2$. (b) Run 17, $Q_o/Q_s = 10$.

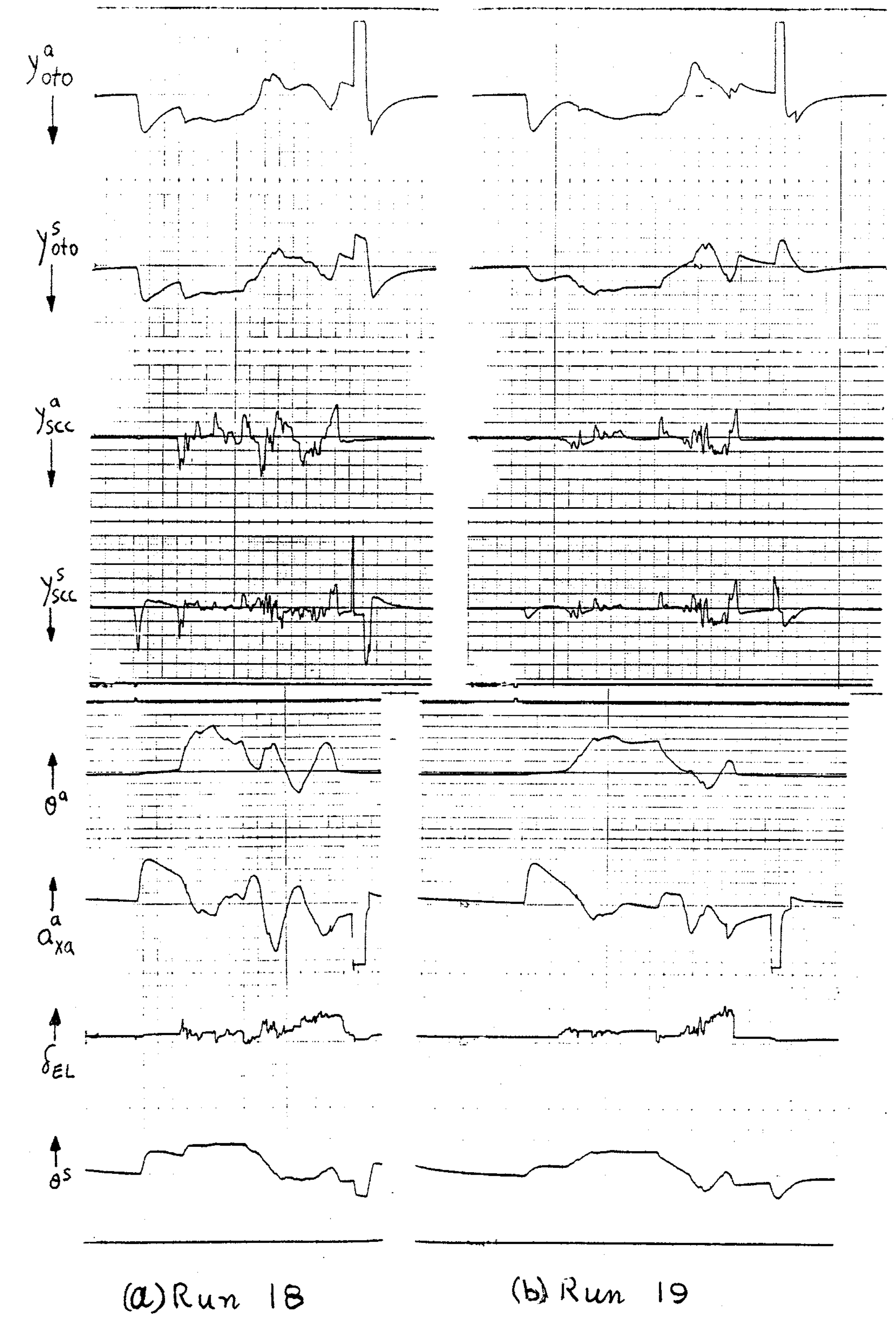


Figure 29. Take-off experimental runs 18, 19 of pilot IM with visual pitch and roll display on. (a) Run 18, $Q_o/Q_s = 32.$. (b) Run 19, is again with $Q_o/Q_s = 0.32$ and can be compared to run 14.

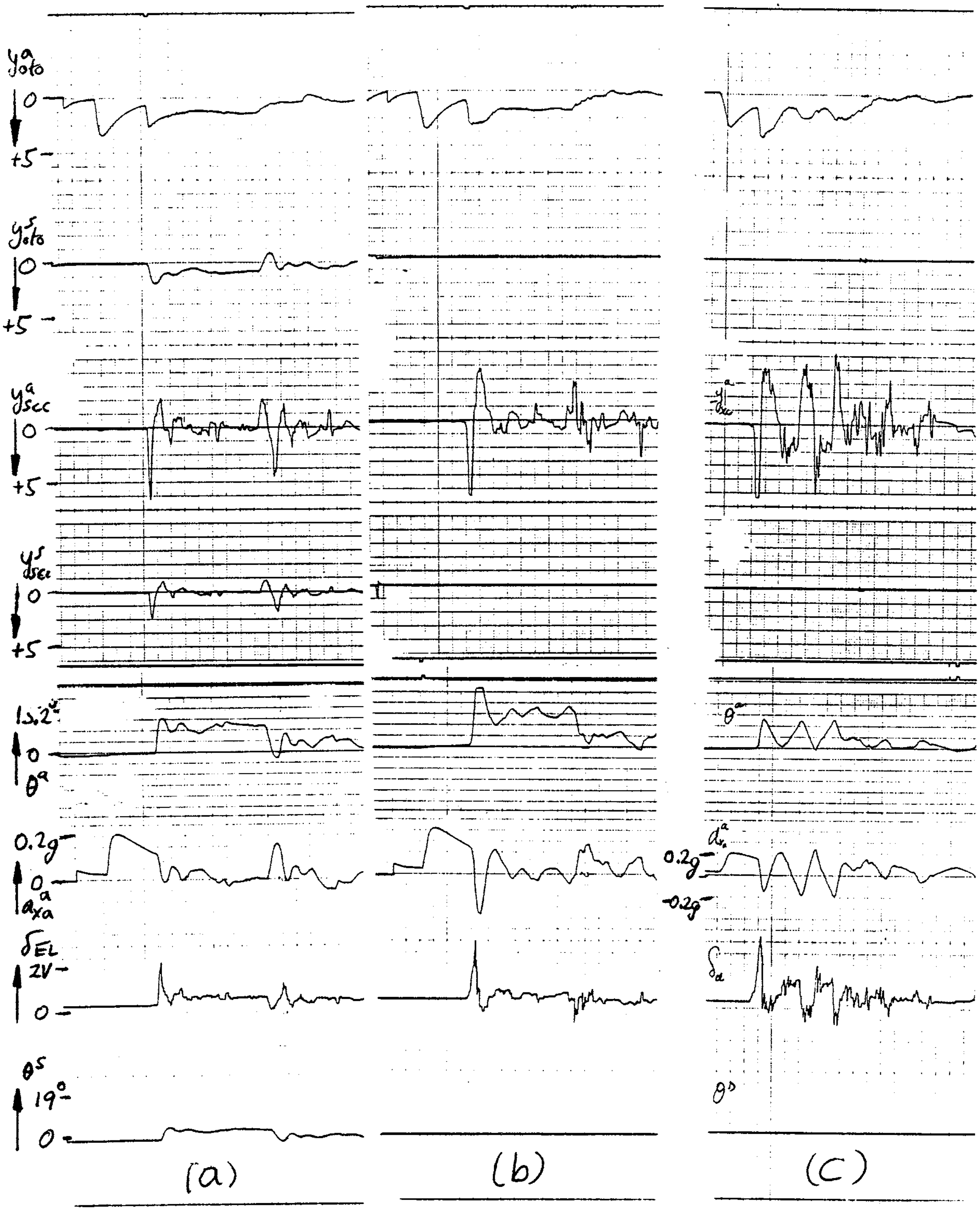


Figure 30. Demonstration of the effect of transition to less motion. (a) IM from $Q_o/Q_s = 0.1$ to Link-like $\theta^s = \theta^a/2$. (b) IM from Link-like to No-Motion. (c) JH first run with No-Motion; had been trained before with no motion; had training two days before on simulator for two hours.

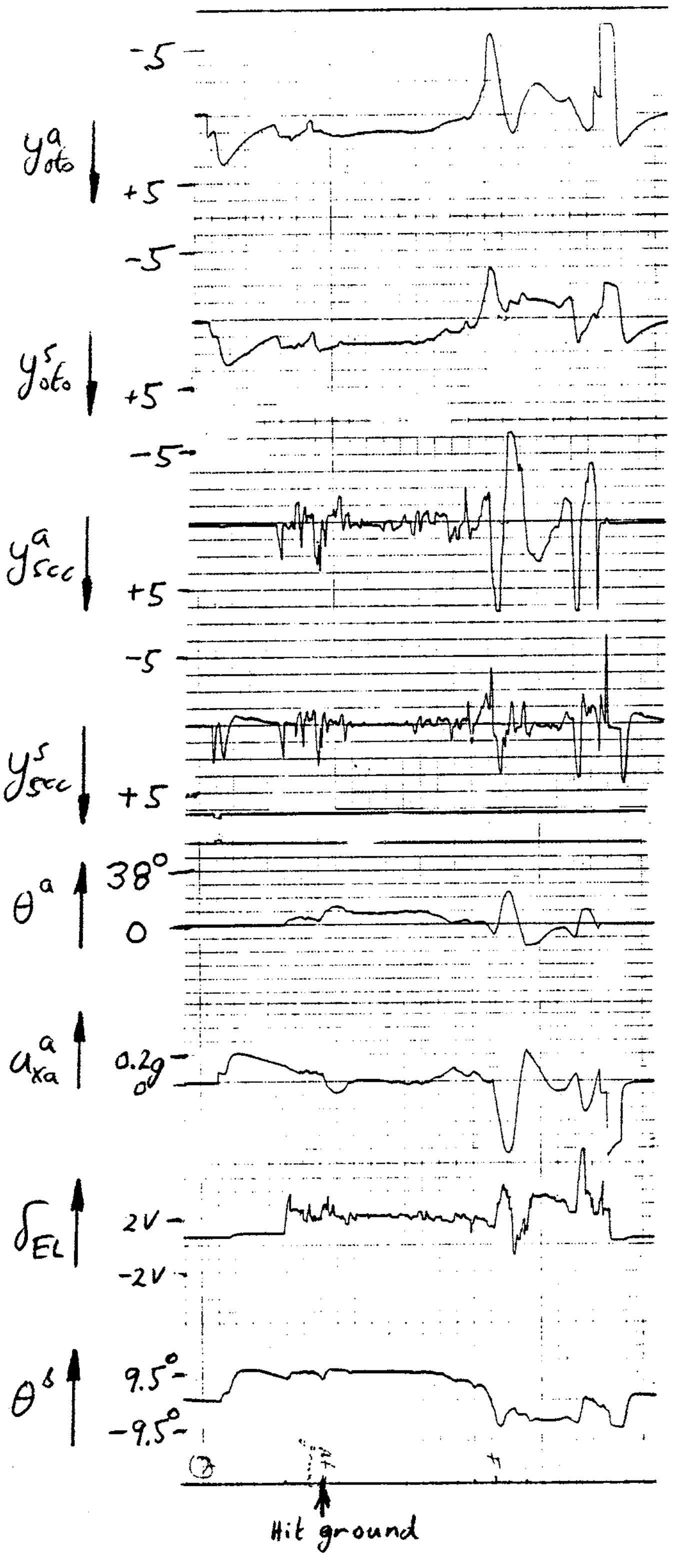


Figure 31. Demonstration of a crash during take-off of pilot WH. The pilot did not pull up the nose due to the incorrect surge linear acceleration at take-off. Note that the washout used here is with $Q_o/Q_s = 10$ and succeeds one with 0.032.

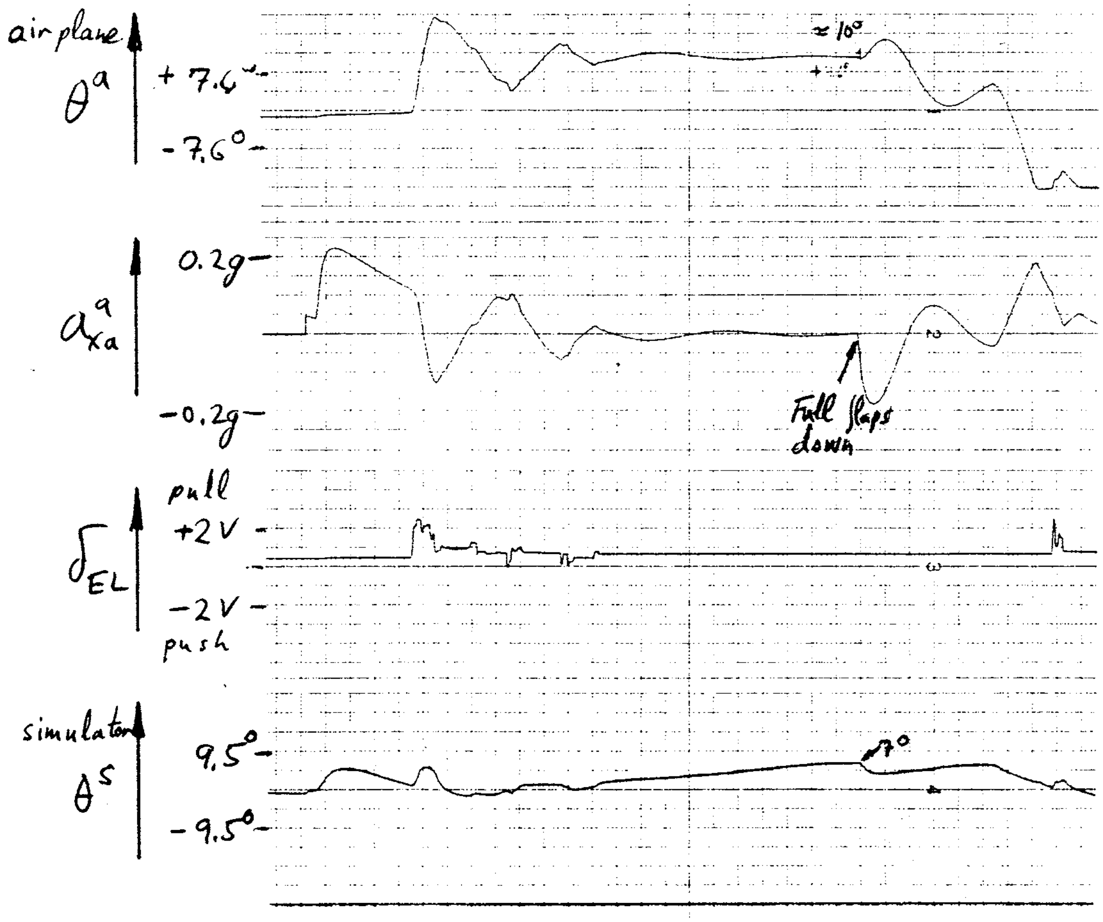


Figure 32. Short Take-off example of "pilot" JI (the experimenter) with a deterministic washout $\theta^s = \theta^a$ and with integrator feedback.

4. Discussion

Twenty pilots were tested using eleven different OWS designs which were implemented for the pitch and surge axes on a Link GAT-1 General Aviation flight simulator Trainer. These tests confirm the suggested design method using a causal, linear, time-invariant, "Gaussian based" OWS (L.Q.G. OWS Chapter III.4), but also point out some of the limitations of such a limited class of designs. The design is confirmed by the small range of design parameter $0.1 \leq Q_o/Q_s \leq 10$ which was experimentally found to cover the whole range that changed noticeably the OWS performance. Furthermore, the best value (based on pilot opinion and a guess based on flaps-down experiment) for this parameter seems to be between 0.32 and 1.0 (change in ratio of otolith to semicircular canal sensitivity of a factor of $1.8 = \sqrt{3.2}$

which confirms the ratio of threshold units chosen and the nominal design of equal weighting for the normalized vestibular linear and rotation components.

The first limitation of the above design is due to the zero mean and ergodic assumptions ("Gaussian based") used to describe the expected airplane motions. During take-off this is clearly not the case and not too surprising the $4^{\underline{th}}$ experiment (take-off) shows the problem. An OWS with $Q_o/Q_s > 0.32$ gives the pilot a false "g-tilt" which causes experienced pilots to crash on take-off due to their reluctance to pull the airplane up, since they feel their nose is too high. This problem can be solved by a time-varying OWS and by use of a deterministic washout i.e. representing the expected airplane motion more accurately using a nonzero mean process that has a time-varying variance. In this case it was shown that it is sufficient to augment the OWS with a very simple deterministic washout ($\theta^s(t) = \theta^a(t)$ in the "steady state") to solve the problem (see example in chapter VI.2 and a demonstration of a short take-off and landing in Figure 32).

A second limitation of the OWS design above is its linearity. The linear OWS designer is required to compromise between hitting the motion limits (very bad [Fuller77]) and giving the pilot more motion. In most cases it is impossible to make a reasonable compromise due to the large dynamic range of the airplane motions. Specifically, in our simulation, pulling the brake after landing gives a very large deceleration which causes the simulator to hit the lower pitch limit. This problem can be solved by a nonlinear washout design using PLQ. Figure 33 shows a comparison of four linear OWS to a nonlinear PLQ OWS design. The input is the simulated airplane acceleration started with full throttle after releasing the parking brake, continued by acceleration to a speed of 70 mph and then decelerating by pulling the parking brake. It is clear that the PLQ OWS design gives the maximum motion without hitting the motion limit. In fact it gives an acceleration motion that is as large as the one obtained for most relaxed limitation linear OWS design with $\kappa = 0$. For more detail of this example see Chapter IV Section 2.3. One should note that using a time-varying OWS would further help to solve this problem but this requires the designer to know in advance at what time, or what "state" of the airplane would lead to a requirement to change the "gain" of the OWS.

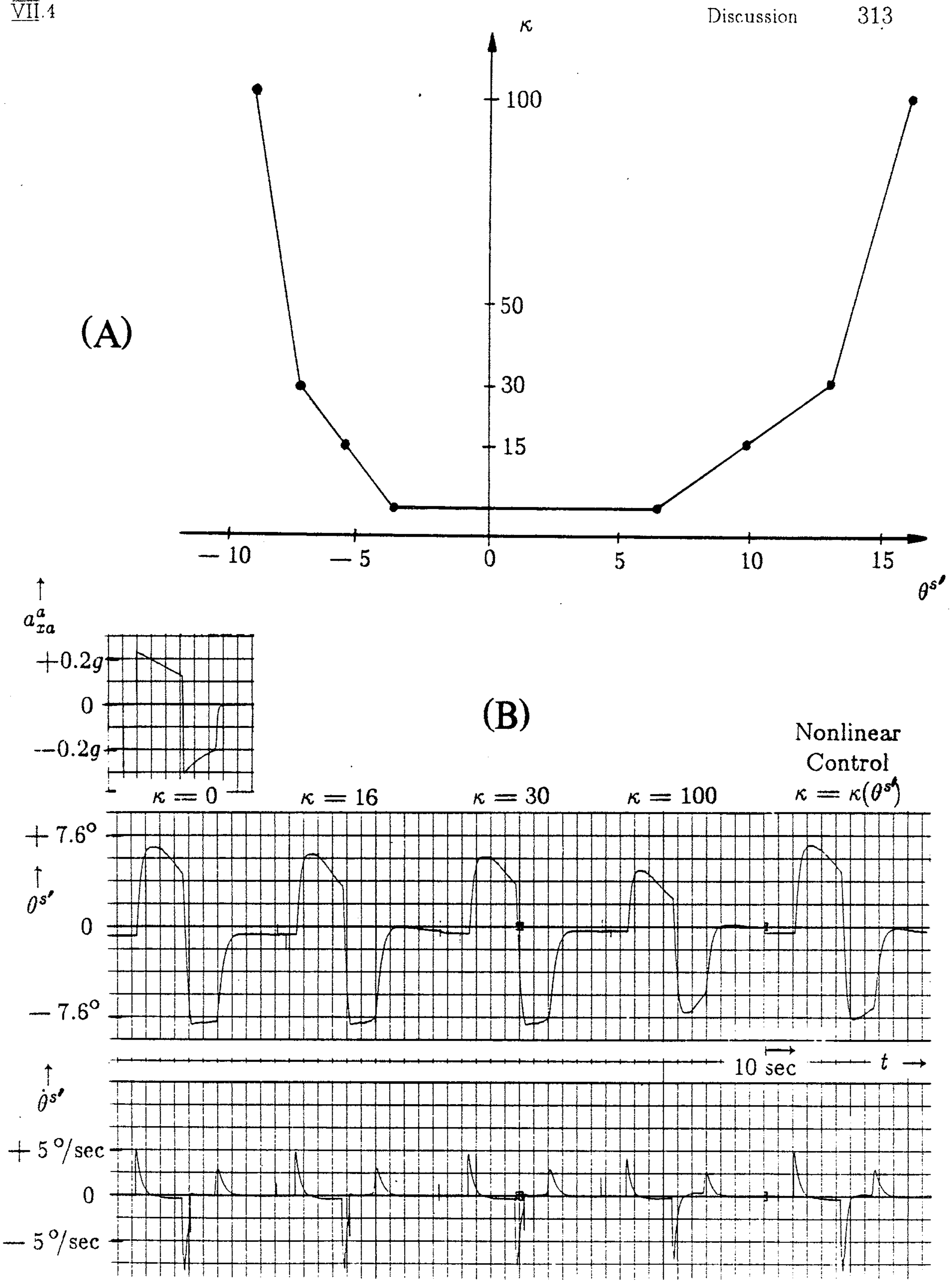


Figure 33. Nonlinear PLQ OWS. (a) Cost function "parameter" $\kappa(\theta^{s'})$ in Chapter $\underline{\text{IV}}$ equation ($\underline{\text{IV}}$.81). (b) Comparison of the nonlinear PLQ OWS for inputs a^a_{xa}, $\theta^a(t) = 0$, $\dot{\theta}^a(t) = 0$, for $\kappa = \kappa(\theta^{s'})$ in (a) to the four linear OWS's used in the PLQ nonlinear OWS implementation.

The blind-test experiment shows, that unless, directed pilots do not notice major motion cues given during the simulation: such as the acceleration cue during take-off (0.1 g given to pilot) and the braking after landing (0.15 g given to pilot). These linear acceleration cues are so large that all non-pilot subjects tested noticed it immediately. Nevertheless there were changes in the pilot controls which they were not aware of as documented in the washout detection and the flaps-down experiments. The flaps-down experiment also suggests a possible reason for many general aviation accidents that occur due to a stall during a landing approach (due to its similarity to this experiment). It was found that even very experienced pilots with more then ten thousand flight hours can easily be confused initially by the motion and make a wrong elevator control. However, they report making the right control. Thus we conclude that there are cases where the simulator motion has a significant influence on the pilot controls and as also seen from the take-off experiment (a crash during take-off). Furthermore pilot comments have to be treated with great caution.

It is not clear if the pilot reaction in the flaps-down experiment is to the incorrect rotation cue (pitch down) or to the correctly simulated deceleration cue. To resolve this question one needs a simulator with linear motion capability and/or a comparison test with a real airplane. A preliminary test done by pilot DM seems to indicate that the deceleration effect in a Cessna 150/152 is smaller (the flaps come down 2 to 3 times slower) but it also causes a pitch down sensation to the pilot. One should note that doing a comparison test in an airplane my differ in the following respects: (i) Heave up acceleration which can be up to 0.1 g. (ii) Flaps-down motor sound. (iii) Vibrations due to flaps down. (iv) Engine sound changes differently. In JI's flight test in a cessna 172 only effect (ii) was noticed.

Pilots like best the Link-like washout due to its lower bandwidth, which reduces the chance of PIO (Pilot Induced Oscillations). The lower bandwidth is suspected to compensate for poor airplane aerodynamic and controls simulation in the GAT-1. It is easier to fly the Link-like washout since the motion given can be used by the pilot as an additional signal that corresponds to pitch attitude as given by the artificial horizon.

Intuitively the times T_d and T_c for each run should be the same since the time of first control T_c, after proper training, should also represent the time of flaps-down detection. Looking at Figures 17, 19 and 21 one can see that the results for T_d are more consistent then the results for T_c. Thus one can assume that the flaps-down detection decision indicated by T_d is "more filtered" then that for T_c. It is surprising to find that in general T_d is shorter then T_c nevertheless. For washouts with $Q_o/Q_s \geq 0.32$ the time $T_c - T_d \approx 0.15\,second$ for both JH and IM. For JW this time difference is much larger $\approx 0.5\,second$ and is probably the reason for his relatively poor performance. This finding relates to the interpretation of the information flow in the pilot conceptual model shown in Chapter II (Section 3.2 and Figure II.4). This finding proves that the outputs of the conceptual "orientation estimator" to the "controller" and to the "pilot orientation feeling" are not the same.

References

[Fuller77] Fuller, R.G., "An Evaluation of the Fidelity of Motion Simulators Using a Model of Human Dynamic Orientation," M.Sc. Thesis, Naval Postgraduate School Monterey, California, September, 1977.

[Harrington79] Harrington, W.W. and Maki, L.R., "LAMARS Motion Fidelity Study in the Heave, Pitch and Sway Degrees-Of-Freedom Using an F-15 Airframe," Air Force Flight Dynamics Lab. Director of Science and Technology Air Force Systems Commend, Wright-Patterson Air Force Base Ohio, AFFDL-TM-79-74-FGD, September, 1979.

[Ish-Shalom82] Ish-Shalom, J., "Recorded data of flaps-down detection and take-off experiments," Man-Vehicle Lab. Report, December, 1982.

[Kohavi70] Kohavi, Z., *Switching and Finite Automata Theory*, McGraw-Hill, p. 408, 1970.

[Parrish76] Parrish, R.V. and Martin, Jr, D.J., "Comparison of a Linear and a Nonlinear WAshout for Motion Simulators Utilizing Objective and Subjective Data from CTOL Transport Landing Approaches," NASA Langley Research Center, NASA TN D-8157, p. 17, June, 1976.

[Puig78] Puig, J.A., Harris, W.T. and Ricard, G.L., "Motion in Flight Simulation: An Annotated Bibliography," Technical Report NAVTRAEQUIPCEN IH-298, Naval Training Equipment Center, Orlando, Florida, July, 1978.

[Sinacori70] Sinacori, J.B., "Validation of Ground Based Simulation," *Journal of American Helicopter Society*, pp. 10–21, July, 1970.

[Sinacori77M] Sinacori, J.B., "A Brief Survey of Motion Simulators' Drive Logic with Emphasis on the Roll Axis," Systems Technology Inc. working paper no. 1094-2, Contract F33615-77-C-0508, May, 1977.

[Sinacori77S] Sinacori, J.B., "The Determination of Some Requirements for a Helicopter Flight Simulation Facility," Technical Report no. 1097-1, Systems Technology Inc., Contract NAS2-9421, pp. 2–12, September, 1977.

Appendix A: Suggested Experiments on the VMS at NASA Ames

1. General Outline of Experiments

This is a description of an initial framework for experiments to test the optimal design methodology for the design of a motion washout system for large flight simulators. It is assumed that the budget for these experiments would be 20 hours of experiment time on the simulator using six pilots as subjects. Not included in these 20 hours are any setup time, equipment testing time, equipment failure time, instruction time for pilots or any other time out not included in the actual experiments. Furthermore, it is assumed that the six subjects are pilots (preferably test pilots) with at least 500 flight hours of experience in a real aircraft and with at least 50 hours of recent flight on the specific airplane to be used in the simulation. This would reduce the training required for the experiments. All six pilots will perform the same experiments. During all experiments, "standard" objective data will be recorded and comparison criteria will be computed. These include platform motion, velocity and acceleration, pilot controls, performance measure, instrument readings, etc. A description of the nonstandard hardware required is given in the next section.

The total of 20 hours experiment time would be broken down into two types of experiments:

1. 11 hours—objective evaluation—using a Two Interval Two Alternative Forced Choice Confidence rating paradigm (2I2AFC).
2. 9 hours—subjective evaluation—comparison of three washout filters.

The goal of the first type of experiment is to find the smallest travel required in order to still give the pilot "acceptable motion" using the optimal washout system in a given maneuver. This result, hopefully, will test the "quality" of the optimal washout system and give the future designer the ability to better predict the smallest travel that gives "acceptable motion" for other maneuvers. Since the first type of experiment does extensive testing only on one maneuver, it is desirable to test other maneuvers as well, and also the overall feeling of the pilots about the motion in the simulator. This is the goal of the second type of experiment. In these tests, the pilot will have a considerable amount of freedom and time to test the airplane simulator performance in several washout settings. These tests will be only generally structured and thus subjective evaluation will be the prime outcome, although objective evaluation will be attempted using the "standard" objective data recorded during the experiment.

The available time in the first experimental stage will be divided into two parts:

1. 8 hours—40 × 6 runs of two minutes each for the six pilots.
2. 3 hours—practice time for the experiments and a few additional runs according to the results of the first part.

Each run of the 40 runs per pilot will be a repetition of a given maneuver that will last for two minutes. A suitable maneuver seems to be a coordinated turn from a fixed initial position and air speed. Another maneuver might be a double coordinated turn. A third possibility is to do a tracking task using a head-up display (HUD).

Each experiment will consist of two runs, referred to as R_1 and R_2, which will be the two intervals of the 2I2AFC paradigm. There are some advantages in having each experiment consist of only one run, where the two washout filters will be interchanged approximately every 30 seconds. The pilot will be informed by an indicator which washout is being used. The response of the pilot immediately after the two runs (referred to as the experiment) will be a four choice rating comparison between the two runs. The possible ratings are:

1. motion in R_1 much better than in R_2
2. motion in R_1 better than in R_2
3. motion in R_1 worse than in R_2
4. motion in R_1 much worse than in R_2

The runs will be chosen from eight different optimal washout filters that will be indexed with the scalar parameter ρ that corresponds to each of them. In general, increasing ρ monotonically decreases the maximum travel required in order to do a given maneuver. It should be understood that changing ρ changes all the parameters for the washout filter. These runs will also include a ninth washout filter that is the best setting of the standard washout filter used for the simulator (this will

```
g\     # ||  0  |  1  |  2  |  3  |  4  |  5  |  6  |  7  |  8  ||  #   |
 r\ ---- ||-----|-----|-----|-----|-----|-----|-----|-----|-----|| runs |
  o\  ρ  |stand-|     |     |     |     |     |     |     |     ||  in  |
   u\    ||dard | 0.1 |0.316|  1  | 2.15| 4.64| 10  | 31.6| 100 ||group |
    p\   ||     |     |     |     |     |     |     |     |     ||      |
-------\ ||-----|-----|-----|-----|-----|-----|-----|-----|-----||------|
         ||     |     |     |     |     |     |     |     |     ||      |
         ||     |     |     |  X_____________________________X  ||      |
   #1    ||     |     |     |     |     |     |     |     |     ||  4   |
         ||     |  X_____________________________X  |     |     ||      |
---------||-----|-----|-----|-----|-----|-----|-----|-----|-----||------|
         ||     |     |     |     |     |     |     |     |     ||      |
         ||     |     |     |     |     |     |  X___________X  ||      |
         ||     |     |     |  X___________X  |     |     |     ||      |
   #2    ||     |     |     |     |  X___________X  |     |     || 12   |
         ||     |  X___________X  |     |     |     |     |     ||      |
         ||     |     |     |     |     |  X___________X  |     ||      |
         ||     |     |  X___________X  |     |     |     |     ||      |
---------||-----|-----|-----|-----|-----|-----|-----|-----|-----||------|
         ||     |     |     |     |     |     |     |  X_____X  ||      |
         ||     |     |     |     |     |  X_____X  |     |     ||      |
         ||     |     |     |  X_____X  |     |     |     |     ||      |
   #3    ||     |  X_____X  |     |     |     |     |     |     || 14   |
         ||     |     |  X_____X  |     |     |     |     |     ||      |
         ||     |     |     |     |  X_____X  |     |     |     ||      |
         ||     |     |     |     |     |     |  X_____X  |     ||      |
---------||-----|-----|-----|-----|-----|-----|-----|-----|-----||------|
         ||     |     |     |     |     |     |     |     |     ||      |
         ||  X_______________________X  |     |     |     |     ||      |
   #4    ||          five times         |     |     |     |     || 10   |
---------||-----|-----|-----|-----|-----|-----|-----|-----|-----||------|
                                                                ||      |
                                      total runs for each pilot :  40   |
                                                                ||------|
```

Figure 1. Objective experiment structure.

be referred to as the standard washout). The experiments fall into four groups, where the fourth group is a "competition" between the optimal washout filter and the standard one. These experiment groups will be performed in the same order for all subjects, namely groups 1, 2, 3, 4. It may be advantageous to spread the experiments in group 4 between the others, but this is still under consideration. The experiments within each group will be randomized. These experiments are described in Figure 1, where a line connects the values of ρ for the two runs that constitute one experiment. Additional pilot ratings for each run can be made individually if they do not confuse the differential ratings described above and will not add too much time. Further analysis of the experiments will be based on comparison of the "standard" objective data recordings.

Although the objective experiment is written as a very structured experiment with all its parameters preset, it would be a good idea to have more flexibility in its design. This is necessary in order to eliminate runs that turn out to be useless and replace them by better choices. This flexibility will be necessary in at least the following sense:

1. Have more than nine washout filter settings available to choose from.
2. Have more than one task (maneuver) available that can be selected easily.
3. Be able to change the experimental protocol easily, quickly and even between runs.

The ability to obtain this flexibility and more depends mainly on the software programming resource available to this experiment.

I hope that this design best utilizes the resources to give the most significant results and to achieve the following goals:

1. Finding the smallest travel required in order to give the pilot "acceptable motion" for a given maneuver.
2. Finding some quantitative relationship between the maximum travel allowed and the quality of the motion produced in the simulator, using the optimal washout filter for a given maneuver.
3. Comparing the optimal washout filter's performance with that of a standard washout, in a given maneuver.
4. Generally testing the "performance" of the optimal washout filter in more general maneuvering of an airplane and comparison with a standard washout.

2. Non-Standard Hardware Requirements

The non-standard hardware requirements in the simulator cab for the VMS experiments are outlined in the following list.

2.1. Response Box

This will consist of four response pushbuttons that are connected to the computer. A schematic diagram of this box is given in Figure 2. These buttons will be active at the end of each run for t_r seconds (to be determined, but approximately 30 seconds). The last button pressed within t_r will light up and stay lit until another button is pressed. The last button pressed in the interval will be recorded by the computer and the light extinguished. At most, one button will be lit at any time during the interval. The·labeling shown in Figure 2 will be on the buttons themselves. The box will be oriented vertically in the simulator cab and will be within reach and sight of the pilot. Its exact location is not important.

2.2. Motion Quality Change Reporter

This enables the pilot to report subjective motion quality changes during the experimental run itself, as an attempt to correlate low motion quality at a given instant with the computed "vestibular error".

The reporter will have two pushbuttons (Figure 3). Each will light up only during the time it is being pressed. The computer will record the time and duration of each press. The pilot will press each button (BETTER or WORSE) for a time he subjectively feels is proportional to the change in the motion quality. Thus, the running sum of time that the "BETTER" button was pressed minus the running sum for the "WORSE" button will be a subjective estimate of the current (at that time) motion quality and will be labeled Q(t). It would be nice to have a plot of Q(t) for each run. Since the magnitude estimate required is not very well defined and consequently difficult for the subject to judge, one should probably rely heavily on the timing information of when each button was pressed.

Another possibility for this device is a *level motion quality reporter* shown in Figure 4. The reporter would have five buttons. The computer will record the time at which each button was pressed. At any time during the simulation, only the last button pushed will be lit. This light will be extinguished at the end of the simulation. At the beginning of each run, the center light (ACCEPTABLE) will be turned on and flashed by the computer. The flashing of the center light will stop as soon as the pilot makes his first choice. The flashing cycle will have a period of about 0.1 *second*. Thus, the reporter will also serve to indicate to the pilot the start of a run (center light goes on) and the end of the run (all lights go out). It is important, however, to keep the option of starting a run with the center light steadily lit or turned off.

2.3. Cab Motion Sensor

Linear and angular position, velocity and acceleration detectors as required by the washout system. These are not fully characterized, but should accommodate

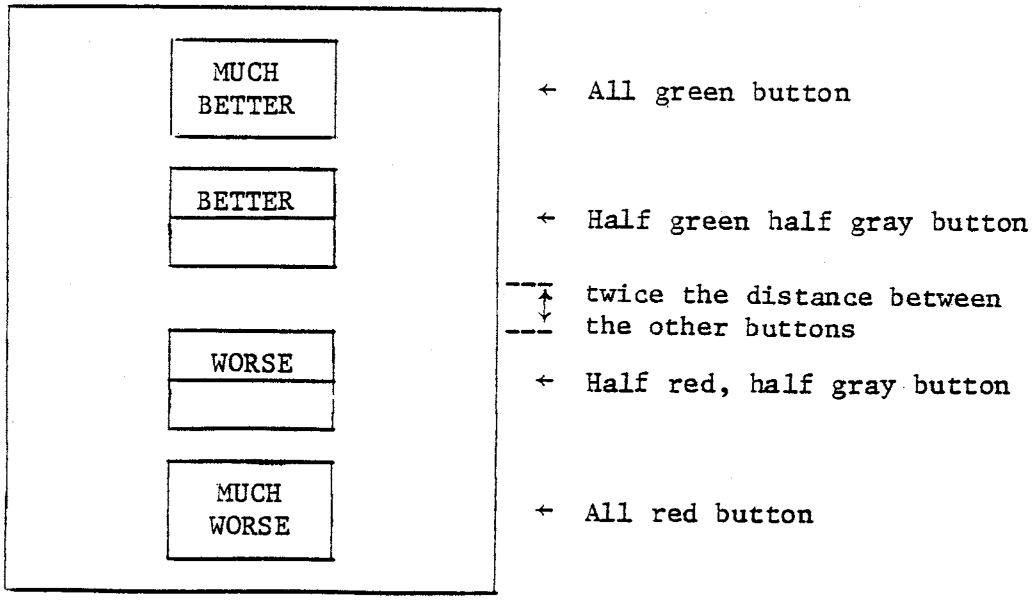

MUCH BETTER
BETTER
WORSE
MUCH WORSE
← All green button
← Half green half gray button
twice the distance between the other buttons
← Half red, half gray button
← All red button

Figure 2. A schematic diagram of the response box

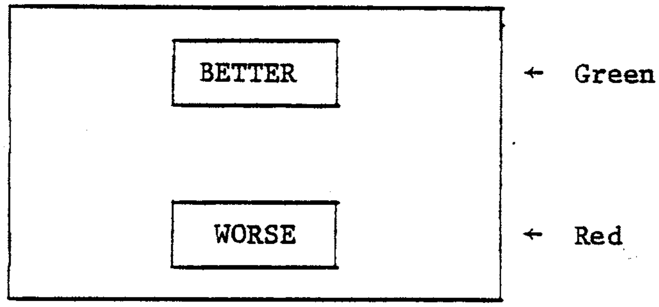


Figure 3. A schematic diagram of the reporter

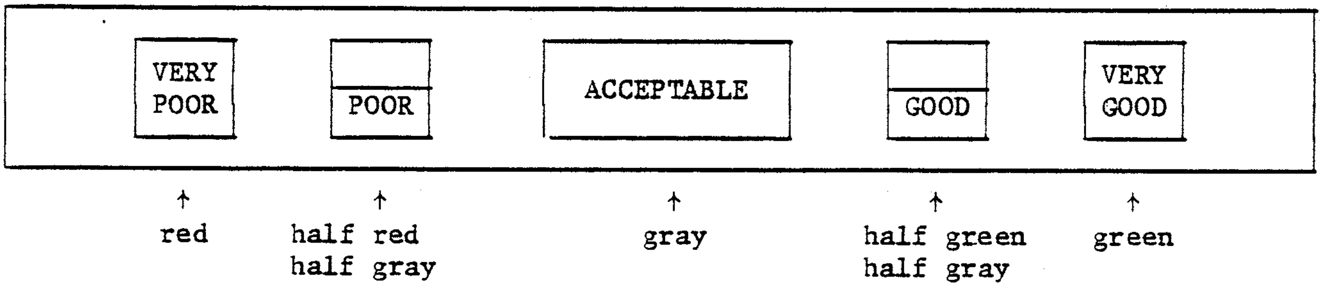


Figure 4. A schematic diagram of the level motion quality reporter

the use of the current available sensors in the VMS, while trying also to come up with a design that will have a low sampling rate.

2.4. Voice Channel Recordings

We would like to have full recordings of both voice channels (subject and operator). On both recordings, it would be necessary to have two cue signals (say two short (0.2 *second*) tone bursts), the first to indicate the beginning of a run (higher tone) and the second to indicate the end of the run (lower frequency tone). If possible, it would be useful to have an automatic recording of the time in voice form before the beginning of each run (or in some unique code, which could be audibly decoded, while listening to the tape recording).

Beyond this, it would be valuable to have a separate recording that will include the times when the pilot initiates a comment on the motion quality (during the run or after), or when the experimenter decides to add a comment on the run. This could be initiated manually by the experimenter. It is as yet undecided whether the pilot should have a special comment button or if his recording would be initiated by voice activation. At the end of each recording segment, there should be an indication of the time, either voice or code. If during the time recording, another segment of voice recoding is required, it will have priority on the time recording.

3. Information Required for the Use of the VMS Simulator

The following is the information to be furnished by the researcher according to Appendix B in the Operations Manual: Vertical Motion Simulator (VMS) S.08, May 1980, NASA Ames Technical Memorandum TM-88180.

II Information for Simulation Management Furnished by Researcher

A. Research Goals

1. This simulation is requested in order to do initial testing of the newly developed optimal washout system. Furthermore, we will also try to test objective and subjective measures of this system. This may enable us to begin linking the vestibular error (deviation from the expected vestibular output) to the subject's opinion of the simulator motion. The VMS is required because of its large motion capability.

2. It is hoped that this experiment will introduce this new washout to the flight simulators users' community where it will be further tested and developed.

3. At this initial stage, this washout system is best suited for very large motion capability flight simulators such as those available at NASA Ames, the LAMARS (at WPAFB) and similar large motion base simulators. Further development will be required in order to use this system on smaller simulators, such as hexapod (synergistic platform) and others.

4. It is also hoped that Ames will support the execution of this experiment using its large motion base system and the available computer software, modified by use of the new washout system. It is hoped that a working familiarity with this new washout system will be acquired by the SSD during the experiment set-up and execution, so that it can be further used in other simulations.

5. There are two critical elements in this simulation program:

   (i) Since the washout system works in a closed loop, it is necessary that the suggested control loop around the motion base (through the PDP 11/55) has a short sampling time. The maximum sampling time which could be tolerated has not yet been determined. Note that this loop sampling time does not necessarily require a higher iteration rate in the math model and that the usual 40–50 msec computation times for each iteration are acceptable.

   (ii) Since most of the experiment is composed of a very large number (approximately 300) of short runs (about two minutes each), we need to go from run to run with as short a pause as possible (hopefully much less than 30 seconds, possibly as short as 5 to 10 seconds).

   Note: If the required sampling times turn out to be less than 5—10 msec, which is probably shorter than can be achieved using the PDP 11/55, there are at least two possible solutions:

   1. Use an analog computer for this.

   2. Use an open loop washout system (filter) as originally proposed in the paper "On the Optimal Design of Optimal Simulators with Application to Flight Simulators", presented at the 1980 IEEE Conference on Man, Society and Cybernetic, September, 1980.

B. Test Plan

1. The plan is to use six experienced test pilots who will all participate in the entire set of experimental runs. Initially, each pilot would have one half hour to familiarize himself with the simulator. Then we will

have three one-half hour training sessions using three washout settings. During these sessions, some subjective data will also be obtained. The three settings to be used are the best of Ames' currently used washout settings and two of the new washout system settings. During these runs, which will be loosely structured, the pilot will be encouraged to comment on his subjective opinion and comparisons of the three washouts. A list of test conditions and the number of runs for the objective experiments are given in Table 1. The values of ρ (a washout design parameter which is roughly inversely proportional to the maximum required travel, smallest $\rho =$ largest required travel) are tentative and actually more than that one parameter may vary between runs. However, the general structure of the experiments will not change.

2. The current estimate of required runs is 264.
3. The current estimate of net simulator time (not including pauses between runs, set-up time, instruction time, equipment testing, equipment failure, instruction time for pilots or any other time not included in the actual experiments) is 23 hours. This is probably equivalent to one half to one third of the actual simulator time (46 to 69 hours). If we are in danger of running out of time, the experiment would be shortened by using fewer pilots (but not less than four). If more time is available, we would repeat runs and/or try different test conditions (i.e. new ρ values). If less than six pilots are available,then the experiment would be expanded as above to use the entire length of time assigned to this experiment.

III Hardware Information Furnished By Searcher

A. Cockpit Requirements

1. Instrument panel—as on the actual airplane with the addition of a response box and a motion quality change reporter described above. The exact location of the response box is not critical as long as the pilot can easily see it and reach to press the buttons at the end of each run. The location of the Motion Quality Change Reporter is more important and should be such that the pilot can press its buttons many times during the run without significantly distracting from his main task of flying the simulator. It also should be located so that the pilot can notice the color of the lights on the buttons which will confirm that he has pressed the button he intended to press. A good place might be on the stick itself, if possible, or the use of some unused buttons on the stick.
2. Controls—as in the actual airplane and such that they behave as closely as possible to those in the airplane.
3. Head movement monitor (desired but not required)—which will measure all three head angular rotations ϕ^h, θ^h, ψ^h (roll, pitch and yaw Euler angles relative to the pilot's seat). A prototype exists for use in the event no suitable equipment is available at Ames. Its required accuracy is 5 percent

of full range. In roll, $\left|\phi^h\right| \leq 60\,deg$ around the vertical with the head up. In pitch, $\left|\theta^h\right| \leq 80\,deg$ around the horizontal (pilot looking straight ahead). Yaw, similar to pitch, $\left|\psi^h\right| \leq 80\,deg$. The bandwidth required is $20\,Hz$ with a noise level less than 0.1 percent of the full range.

B. Display Requirements

The use of the model board is preferred, if this provides a 25+ *mile* range of travel at an altitude of 1,000 to $2,000\,feet$. The flight speed will be 0.7 to $0.85\,Mach$ for two minutes. We would like to have the best possible display system in daylight under these conditions. We would also like the displays system delay to be as small as possible and preferably below $50\,msec$ to reach 90 percent of the display position change in each iteration.

IV Math Model Information Furnished by the Researcher

A–G

1. Significant aircraft roll rates: It would be valuable to be able to roll into a 30–45 *degree* bank angle in 0.5 to 1.0 *seconds*.
2. High cruise speed, so as to obtain these lateral forces with low yaw rate. This is needed because the simulation will have only three degrees of freedom of motion active (vertical, lateral and roll z, y, ϕ). The yaw axis will not be activated.
3. Test pilots available with sufficient flight experience on the chosen flight vehicle and hopefully at least some experience as a simulator pilot. This is required to shorten the training time necessary on the simulator. Furthermore, it is important that the pilots have a good notion of the motion of the real flight vehicle so that they can make the judgments required from them during the experiment.
4. Try to choose a flight vehicle which will be already set up before the experiments start, so as to reduce the amount of time (to zero if possible) needed to verify or debug the math model of the flight vehicle. If this is possible and we were able to use the same pilots used in the previous experiment, then there would be advantages to both the MIT test program and possibly also to the previous users.
5. Have a low cycle time vehicle math model to keep the entire simulator cycle time small.
6. Use a flight vehicle that is of the current (or recent) interest so that we might get help in setting up the experiment and so that this washout may have continued use on the VMS.

II Atmospheric Disturbance Model

It may be possible that an atmospheric disturbance would be desired, but the experiment requirements will probably be satisfied with the current available repertoire. We would like to obtain some information on the available models, so that we will be prepared to choose a model if it seems useful.

I Math Model Validation

We will use the previously defined procedures for the model.

Appendix B: Washout Design Programs

This appendix includes two sets of programs: WP and WPI. WPI includes a cost on the integral of the pitch angle $\theta^{s'}$ which leads to an integrator in the feedback loop. Note that the program WP is a special case of program WPI and thus the results of WP can be obtained from WPI using parameter 18 equal 0 (RPINT=0). This method of use of WPI is not recommended due to numerical calculation problems in WPI.

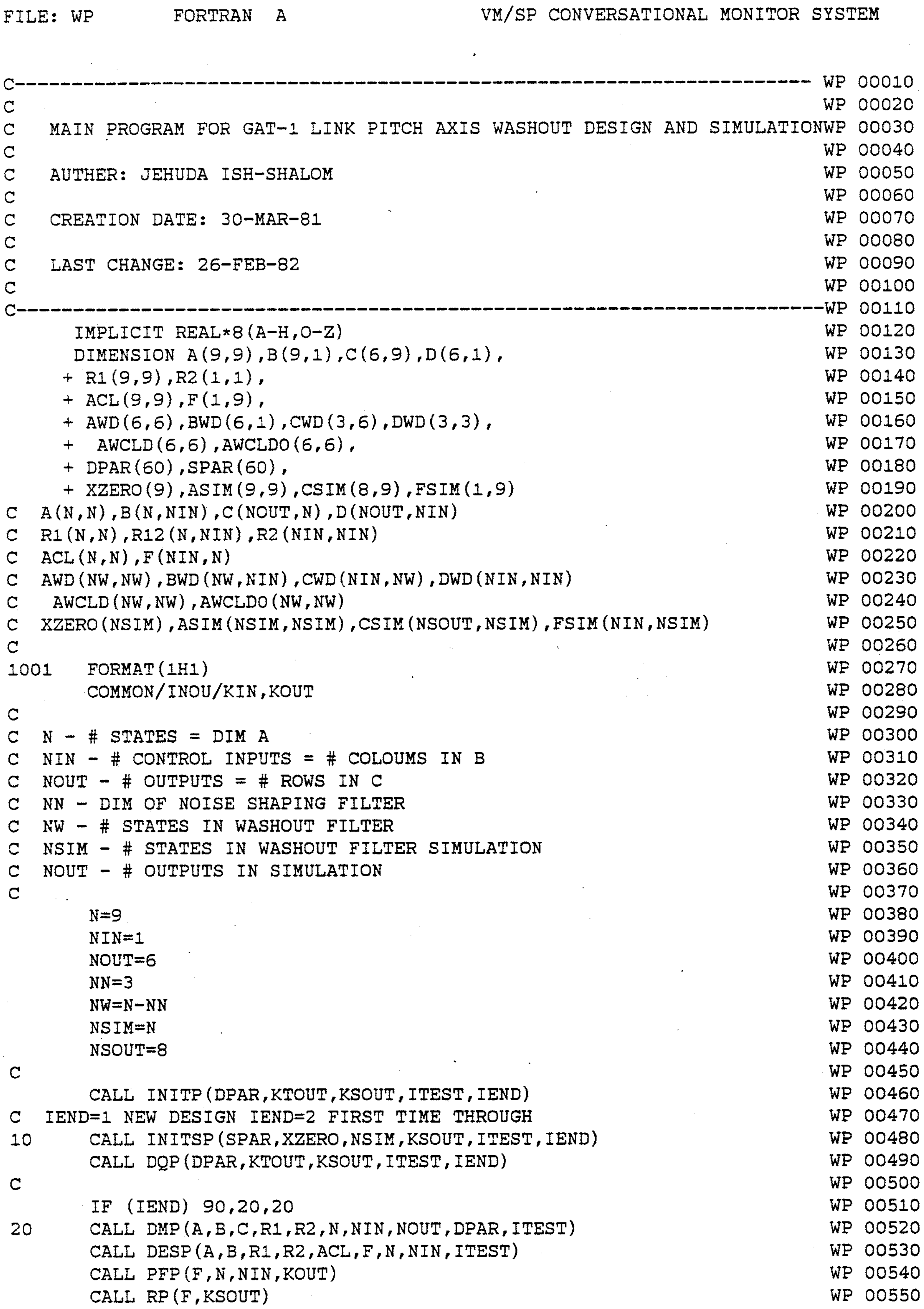
Program WP

This program includes an exec, main program and twelve subroutines: INITP, INITSP, DQP, DMP, DESP, PFP, RP, STP, SQP, SMP, SIMP and TPRINT (given in the set for WPI programs).

Program WPI

This program includes an exec, main program and twelve subroutines: INITPI, INISPI, DQPI, DMPI, DESPI, PFPI, RPI, STPI, SQP, SMPI, SIMPI and TPRINT.

```
FILE: WP       EXEC     A                VM/SP CONVERSATIONAL MONITOR SYSTEM


FILEDEF 7 TERM
FILEDEF 5 TERM
FILEDEF 6 DISK O DATA E (RECFM F LRECL 132 BLKSIZE 132
FILEDEF 8 DISK S DATA E (RECFM F LRECL 132 BLKSIZE 132
LOAD WP (NOMAP START
PRINT S DATA E (CC)
PRINT O DATA E (CC)
CP SPOOL PRINT CLOSE
QPRINT S DATA E (SIDES 2 CC
COST
```



```
FILE: WP       FORTRAN  A                   VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------
C
C   MAIN PROGRAM FOR GAT-1 LINK PITCH AXIS WASHOUT DESIGN AND SIMULATION
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 30-MAR-81
C
C   LAST CHANGE: 26-FEB-82
C
C------------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(9,9),B(9,1),C(6,9),D(6,1),
     + R1(9,9),R2(1,1),
     + ACL(9,9),F(1,9),
     + AWD(6,6),BWD(6,1),CWD(3,6),DWD(3,3),
     +  AWCLD(6,6),AWCLDO(6,6),
     + DPAR(60),SPAR(60),
     + XZERO(9),ASIM(9,9),CSIM(8,9),FSIM(1,9)
C  A(N,N),B(N,NIN),C(NOUT,N),D(NOUT,NIN)
C  R1(N,N),R12(N,NIN),R2(NIN,NIN)
C  ACL(N,N),F(NIN,N)
C  AWD(NW,NW),BWD(NW,NIN),CWD(NIN,NW),DWD(NIN,NIN)
C   AWCLD(NW,NW),AWCLDO(NW,NW)
C  XZERO(NSIM),ASIM(NSIM,NSIM),CSIM(NSOUT,NSIM),FSIM(NIN,NSIM)
C
1001   FORMAT(1H1)
       COMMON/INOU/KIN,KOUT
C
C  N - # STATES = DIM A
C  NIN - # CONTROL INPUTS = # COLOUMS IN B
C  NOUT - # OUTPUTS = # ROWS IN C
C  NN - DIM OF NOISE SHAPING FILTER
C  NW - # STATES IN WASHOUT FILTER
C  NSIM - # STATES IN WASHOUT FILTER SIMULATION
C  NOUT - # OUTPUTS IN SIMULATION
C
       N=9
       NIN=1
       NOUT=6
       NN=3
       NW=N-NN
       NSIM=N
       NSOUT=8
C
       CALL INITP(DPAR,KTOUT,KSOUT,ITEST,IEND)
C  IEND=1 NEW DESIGN IEND=2 FIRST TIME THROUGH
10     CALL INITSP(SPAR,XZERO,NSIM,KSOUT,ITEST,IEND)
       CALL DQP(DPAR,KTOUT,KSOUT,ITEST,IEND)
C
       IF (IEND) 90,20,20
20     CALL DMP(A,B,C,R1,R2,N,NIN,NOUT,DPAR,ITEST)
       CALL DESP(A,B,R1,R2,ACL,F,N,NIN,ITEST)
       CALL PFP(F,N,NIN,KOUT)
       CALL RP(F,KSOUT)
      WRITE(KSOUT,1001)
      CALL PFP(F,N,NIN,KTOUT)
      CALL RP(F,KOUT)
C  CHECK FOR ERROR (ITEST<0)
      IF (ITEST.LT.0) GOTO 90
      CALL STP(ACL,B,F,DTW,AWD,BWD,CWD,DWD,AWCLD,AWCLDO,DPAR,
     +  N,NIN,NW,NN,KTOUT,ITEST)
30    CALL SQP(SPAR,XZERO,NSIM,KTOUT,KSOUT,ITEST,IEND)
C  IEND=0 CONTINUE, IEND>0 NEW DESIGN, IEND<0 END
      IF (IEND) 90,40,10
40    CALL SMP(ACL,C,D,F,ASIM,CSIM,FSIM,XZERO,DPAR,SPAR,
     +  N,NIN,NOUT,NSIM,NSOUT,NW,ITEST)
      CALL SIMP(ASIM,CSIM,FSIM,XZERO,SPAR,NSIM,NIN,NSOUT,ITEST)
C  IF ITEST<0 ERROR IN SIMULATION => END PROGRAM
      IF (ITEST.GE.0) GOTO 30
90    CONTINUE
      END
```

```
FILE: INITP    FORTRAN  A                VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------
C
C   SUBROUTINE TO INITIALIZE QUERY VARIABELS AND SIMULATION STATE
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   INPUT: ITEST
C     ITEST - IF ITEST>0 PRINT DEBUGING PRINTOUTS
C
C   OUTPUT: DPAR(60),KTOUT,KSOUT,IEND
C     DPAR - DESIGN PARAMETERS
C     KTOUT - FILE NUMBER OF TERMINAL
C     KSOUT - FILE NUMBER OF SUMMERY FILE
C     IEND  - =2 TO SIGNAL NEW DESIGN THE FIRST TIME THROUGH TO SQP SUB.
C
C   AUTHOR: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 30-MAR-81
C   LAST CHANGED: 16-NOV-81
C
C-----------------------------------------------------------------------
C
      SUBROUTINE INITP(DPAR,KTOUT,KSOUT,ITEST,IEND)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION DPAR(60),SPAR(60),PAR(25),ZIN(9)
      EQUIVALENCE (GO,PAR(1)),(BO,PAR(2)),(AO,PAR(3)),(GS,PAR(4)),
     + (AS,PAR(5)),(BNL,PAR(6)),(BNR,PAR(7)),(BNI,PAR(8)),(BR,PAR(9)),
     + (BBR,PAR(10)),(PIP,PAR(11)),(BI,PAR(12)),
     + (GEE,PAR(13)),(PKO,PAR(14)),(PKS,PAR(15)),(PQ,PAR(16)),
     + (RHO,PAR(20))
C
      COMMON/INOU/KIN,KOUT
      KIN=5
      KOUT=6
      KTOUT=7
      KSOUT=8
C
C     FILE ASSIGNMENT
C
C     FILE 5: INPUT FROM THE TERMINAL (KIN)
C     FILE 6: OUTPUT TO THE LINE PRINTER (KOUT)
C     FILE 7: OUTPUT TO THE TERMINAL  (KTOUT)
C     FILE 8: SUMMERY OUTPUT TO THE LINE PRINTER (KSOUT)
C
C    UNITS USED     G'S,RADENS,SECANDS;
C                    OTHER UNIT ARE NORMALIZED
C
1001  FORMAT(' SCALERS EXITING SUB. INIT',/,' IEND= ',I3,' ITEST= ',I3,
     + ' KTOUT= ',I3,' KIN= ',I3,' KOUT= ',I3,' KSOUT= ',I3)
1002  FORMAT(1H1,' GAT-1 LINK PITCH AXIS WASHOUT DESIGN PROGRAM',/,/,
     + 4X,' WRITTEN BY JEHUDA ISH-SHALOM (OCTOBER 1981)',/)
1003  FORMAT(/,' MOTION BASE COMMAND INPUT: ANGULER VELOCITY',/,
     + ' MEASURED STATES: ANGULER POSTION AND VELOCITY, AND LINEAR',
     + ' ACCELERATION IN MOTION BASE INIRIAL AXES',/,
     + ' VESTIBULAR MODEL INPUTS:',/,
     + 10X,' LINEAR  = OTOLITH : LINEAR ACCELERATION',/,
     + 10X,' ANGULER = SEMICIRCULAR CANAL: ANGULER VELOCITY',/)
C
      WRITE(KTOUT,1002)
      WRITE(KOUT,1002)
      WRITE(KOUT,1003)
C
      ITEST=0
      IEND=2
C   OTOLITH MODEL  OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD).
      GO=21.168D0
      BO=.076D0
      AO=.19D0
C   SEMICIRCULAR CANAL MODEL OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD).
      AS=1/5.9D0
      GS=40.D0
C
      GEE=-1.D0
C
C  NOISE FILTER
C
      BNL=0.1D0
      BNR=0.1D0
      BNI=1.D-5
C  MOTION BASE
      BR=3.13D0
      BBR=0.167D0
      PIP=0.74D0
      BI=0.D0
C  COST VALUES
      PKO=1.D0
      PKS=1.D0
      PQ=0.D0
      PAR(17)=1.D0
C
      PAR(18)=0.D0
      PAR(19)=0.D0
C
      RHO=0.04D0
C  MOTION BASE PARAMETERS
      PAR(21)=0.3D0
      PAR(22)=3000.D0
      PAR(23)=50.D0
      PAR(24)=0.D0
      PAR(25)=2000.D0
C
      DO 25 I=1,60
25    DPAR(I)=0.
      DO 30 I=1,25
30    DPAR(I)=PAR(I)
C
      CALL TPRINT(1,1,60,DPAR,'DPAR AFTER SUB. INIT $',ITEST)
      IF (ITEST.GT.0) WRITE(KOUT,1001) IEND,ITEST,KTOUT,KIN,KOUT,KSOUT
      RETURN
      END
```

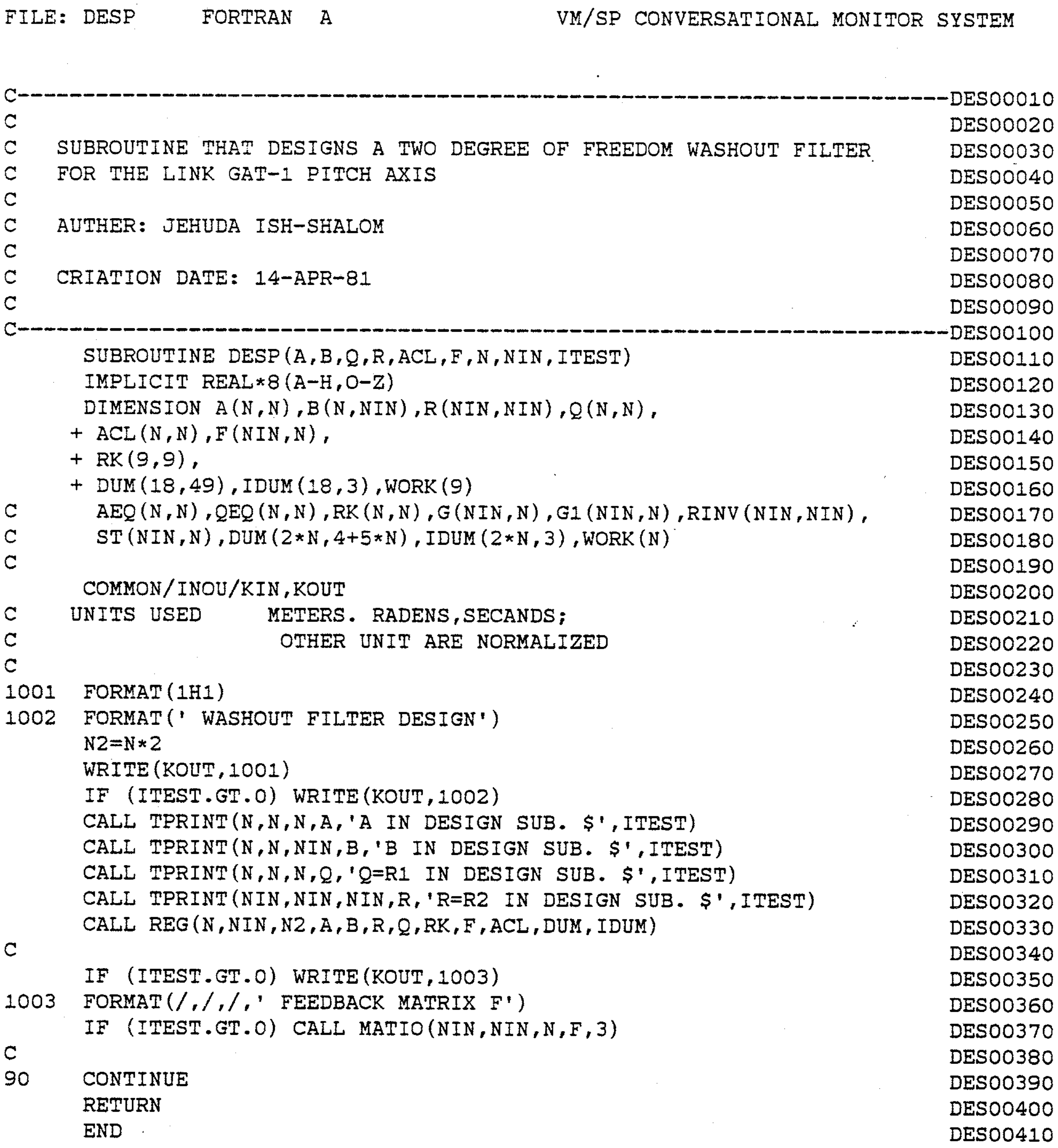
```
FILE: INITSP   FORTRAN  A                  VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------INI00010
C                                                                       INI00020
C   SUBROUTINE TO INITIALIZE QUERY SIMULATION VARIABELS AND STATES      INI00030
C   FOR THE LINK GAT-1 PITCH AXIS                                       INI00040
C                                                                       INI00050
C   INPUT: NSIM,ITEST,IEND                                              INI00060
C     NSIM - DIMESION OF SIMULATION MATRIX ASIM                         INI00070
C     ITEST - IF ITEST>0 PRINT DEBUGING PRINTOUTS                       INI00080
C     IEND - IF IEND = 1 PRINT NEW SIMULATION IN FILE KSOUT             INI00090
C                                                                       INI00100
C   OUTPUT: SPAR(60),XZERO(NSIM)                                        INI00110
C     SPAR - SIMULATION PARAMETERS                                      INI00120
C     XZERO - SIMULATION INITIAL STATE                                  INI00130
C                                                                       INI00140
C   AUTHOR: JEHUDA ISH-SHALOM                                           INI00150
C                                                                       INI00160
C   CREATION DATE: 12-OCT-81                                            INI00170
C   LAST CHANGED: 21-OCT-81                                             INI00180
C                                                                       INI00190
C-----------------------------------------------------------------------INI00200
C                                                                       INI00210
      SUBROUTINE INITSP(SPAR,XZERO,NSIM,KSOUT,ITEST,IEND)               INI00220
      IMPLICIT REAL*8(A-H,O-Z)                                          INI00230
      DIMENSION SPAR(60),PAR(25),ZIN(9),XZERO(NSIM)                     INI00240
      EQUIVALENCE                                                       INI00250
     + (XZL,ZIN(1)),(SLL,ZIN(2)),(PLMAX,ZIN(3)),                        INI00260
     + (XZR,ZIN(4)),(SLR,ZIN(5)),(RMAX,ZIN(6)),                         INI00270
     + (DT,ZIN(7)),(T,ZIN(8)),(XZRV,ZIN(9))                             INI00280
      COMMON /INOU/KIN,KOUT                                             INI00290
C                                                                       INI00300
C    UNITS USED     G'S,RADENS,SECANDS;                                 INI00310
C                    OTHER UNIT ARE NORMALIZED                          INI00320
C                                                                       INI00330
1001  FORMAT(' NEW SIMULATION',//)                                      INI00340
      IF (IEND.EQ.1) WRITE(KSOUT,1001)                                  INI00350
1002  FORMAT(' SCALARS EXITING SUB. INITSP',/,                          INI00360
     + ' IEND= ',I3,' ITEST= ',I3,' NSIM= ',I3,' NPTS= ',I5,' NPRPL= ', INI00370
     + I2,' DT= ',D20.15,' T= ',D20.15)                                 INI00380
C                                                                       INI00390
      XZL=0.2D0                                                         INI00400
      SLL=0.D0                                                          INI00410
      PLMAX=0.D0                                                        INI00420
      XZR=0.D0                                                          INI00430
      SLR=0.D0                                                          INI00440
      RMAX=0.D0                                                         INI00450
      XZRV=0.D0                                                         INI00460
      DO 20 I=1,NSIM                                                    INI00470
20    XZERO(I)=0.D0                                                     INI00480
      XZERO(7)=XZL                                                      INI00490
      XZERO(8)=XZR                                                      INI00500
      XZERO(9)=XZRV                                                     INI00510
C                                                                       INI00520
      DT=.2D0                                                           INI00530
      NPTS=51                                                           INI00540
C  NPRPL > 0 PRINT AND PLOT                                             INI00550
```

```
C  NPRPL = 0 PRINT ONLY
C  NPRPL < 9 PLOT ONLY
      NPRPL=-10
      T=DT*(NPTS-1)
C
      DO 25 I=1,60
25    SPAR(I)=0.
      SPAR(1)=DFLOAT(NPTS)
      SPAR(2)=DFLOAT(NPRPL)
      SPAR(3)=DT
      DO 32 I=1,9
32    SPAR(I+3)=ZIN(I)
C
      CALL TPRINT(1,1,60,SPAR,'SPAR AFTER SUB. INIT $',ITEST)
      CALL TPRINT(NSIM,NSIM,1,XZERO,'XZERO AFTER SUB. INIT $',ITEST)
      IF (ITEST.GT.0) WRITE(KOUT,1002) IEND,ITEST,NSIM,NPTS,NPRPL,DT,T
      RETURN
      END
```

```
FILE: DQP      FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------DQP00010
C   SUBROUTINE THAT QUERYS THE PARAMETERS FOR THE WASHOUT DESIGN        DQP00020
C              FOR THE PITCH AXIS OF THE LINK GAT-1 FLIGHT SIMULATOR    DQP00030
C                                                                       DQP00040
C   AUTHER: JEHUDA ISH-SHALOM                                           DQP00050
C                                                                       DQP00060
C   CREATION DATE: 27-MAR-81                                            DQP00070
C                                                                       DQP00080
C  CHANGE 10-OCT-81: COST OF MOTION BASE ANGLE, ANGULER RATE AND COMMANDDQP00090
C                    NOT NORMILIZED TO 1. EACH IS INDEPENDEDNT.         DQP00100
C   LAST CHANGE: 12-OCT-81                                              DQP00110
C                                                                       DQP00120
C-----------------------------------------------------------------------DQP00130
      SUBROUTINE DQP(DPAR,KTOUT,KSOUT,ITEST,IEND)                       DQP00140
      IMPLICIT REAL*8(A-H,O-Z)                                          DQP00150
      DIMENSION PAR(25),DPAR(60)                                        DQP00160
      INTEGER*2 STAR,BLANK,IS(25)                                       DQP00170
      REAL*4 DAYTIM(5)                                                  DQP00180
C   DAYTIM HOLDS THE DAY AND TIME AFTER THE CALL WHEN(DAYTIM)           DQP00190
C      MO/DY/YR HR*MN*SC.SC(*10**-2)                                    DQP00200
C                                                                       DQP00210
      EQUIVALENCE (GO,PAR(1)),(BO,PAR(2)),(AO,PAR(3)),(GS,PAR(4)),      DQP00220
     + (AS,PAR(5)),(BNL,PAR(6)),(BNR,PAR(7)),(BNI,PAR(8)),              DQP00230
     + (BR,PAR(9)),(BBR,PAR(10)),(PIP,PAR(11)),(BI,PAR(12)),            DQP00240
     + (GEE,PAR(13)),(PKO,PAR(14)),(PKS,PAR(15)),(PQ,PAR(16)),          DQP00250
     + (RHO,PAR(20)),                                                   DQP00260
     + (PLCG,PAR(21)),(W,PAR(22)),(PIYY,PAR(23)),(PB,PAR(24)),          DQP00270
     + (PKM,PAR(25))                                                    DQP00280
C                                                                       DQP00290
      COMMON/INOU/KIN,KOUT                                              DQP00300
C                                                                       DQP00310
      DATA STAR/2H */,BLANK/2H  /                                       DQP00320
C                                                                       DQP00330
C    UNITS USED     METERS. RADENS,SECANDS;                             DQP00340
C                    OTHER UNIT ARE NORMALIZED                          DQP00350
C                                                                       DQP00360
C  FORMAT STATMETS                                                      DQP00370
1001  FORMAT(' OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK',    DQP00380
     + ' LOAD)',/,' (THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))',/,           DQP00390
     + A2,'  1  GO=',F9.3,' (1/G)',/,                                   DQP00400
     + A2,'  2  BO=',F9.3,' (RAD/SEC)',/,                               DQP00410
     + A2,'  3  AO=',F9.3,' (RAD/SEC)',/,                               DQP00420
     + ' SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS',          DQP00430
     + ' (WITH WORK LOAD)',/,                                           DQP00440
     + ' (THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))',/,               DQP00450
     + A2,'  4  GS=',F9.3,' (1/RAD)',/,                                 DQP00460
     + A2,'  5  AS=',F9.3,' (RAD/SEC)',/,                               DQP00470
     + ' NOISE FILTER',/,                                               DQP00480
     + A2,'  6  BNL=',F8.3,' (RAD/SEC)',/,                              DQP00490
     + A2,'  7  BNR=',F8.3,' (RAD/SEC)',/,                              DQP00500
     + A2,'  8  BNI=',D8.2,' (1/SEC)',/,                                DQP00510
     + ' MOTION BASE MODEL',/,                                          DQP00520
     + A2,'  9  BR=',F9.3,' (1/SEC)',/,                                 DQP00530
     + A2,'  10 BBR=',F8.3,' (RAD/(SEC**2*VOLT))',/,                    DQP00540
     + A2,'  11 PIP=',D8.2,' (1/SEC**2)',/,                             DQP00550
     + A2,'  12 BI=',D9.2,' (1/SEC)',/,' GRAVITY CONSTANT',/,
     + A2,'  13 GEE=',F8.3,' (G)')
10011 FORMAT( ' COST FUNCTION PRAMETERS',/,
     + '  ERROR',/,
     + A2,'  14 PKO=',F12.10,' OTO SCALING OF REQ. RESPONSE PKO>0',/,
     + A2,'  15 PKS=',F12.10,' SCC SCALING OF REQ. RESPONSE PKS>0',/,
     + A2,'  16 PQ=',F13.10,' PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL',/,
     + A2,'  17 QO/QS=',D10.2,' OTOLITH/SEMICIRCULAR CANAL',/,
     + '  ROTATION MOTIONS',/,
     + A2,'  18 RRO/R=',D10.2,' ANGLE/VELOCITY COMMAND',/,
     + A2,'  19 RR1/R=',D10.2,' VELOCITY/VELOCITY COMMAND',/,
     + '  GLOBAL SCALING',/,
     + A2,'  20 RHO=',D12.2,' WEIGHT OF MOTION OVER ERROR',/)
C    + ' PITCH MOTION BASE PARAMETERS',/,
C    + A2,'  21 PLCG=',D11.2,' LENGTH TO C.G., POS=INV PENDULAM (M)',/,
C    + A2,'  22 W=',D14.2,' SIMULATOR WIGHT (NET=KG*M/SEC**2)',/,
C    + A2,'  23 PIYY=',D11.2,' MOMENT OF INERTIA (KG*M**2)',/,
C    + A2,'  24 PB=',D13.2,' VISCOUS FRICTION (NET*M*SEC)',/,
C    + A2,'  25 PKM=',D12.2,' MOTOR CONSTANT (NET*M/SEC)')
1002  FORMAT(' CHANGE PAR # (-1 EXIT; 0 CONTINUE; 80,81,82 PRINT;',
     + '90,91,92 TEST) : ')
1003  FORMAT(' PAR(',I2,')=',D20.9,' NEW VALUE :')
1004  FORMAT(D20.9)
1005  FORMAT(/,' WAHOUT FILTER DESIGN PARAMETERS (',5A4,')',/)
1006  FORMAT(' DPAR(I)',/,
     + ' DPAR(',I2,')=GO=',D15.9,' DPAR(',I2,')=BO=',D15.9,
     + ' DPAR(',I2,')=AO=',D15.9,/,
     + ' DPAR(',I2,')=GS=',D15.9,' DPAR(',I2,')=AS=',D15.9,/,
     + ' DPAR(',I2,')=BNL=',D15.9,' DPAR(',I2,')=BNR=',D15.9,
     + ' DPAR(',I2,')=BNI=',D15.9,/,
     + ' DPAR(',I2,')=BR=',D15.9,' DPAR(',I2,')=BBR=',D15.9,
     + ' DPAR(',I2,')=PIP=',D15.9,' DPAR(',I2,')=BI=',D15.9,/,
     + ' DPAR(',I2,')=GEE=',D15.9,/,
     + ' DPAR(',I2,')=PKO=',D15.9,' DPAR(',I2,')=PKS=',D15.9,
     + ' DPAR(',I2,')=PQ=',D15.9,/,
     + ' DPAR(',I2,')=QO/QS=',D15.9,' DPAR(',I2,')=RRO/R=',D15.9,
     + ' DPAR(',I2,')=RR1/R=',D15.9,/,
     + ' DPAR(',I2,')=RHO=',D15.9,/,
     + ' DPAR(',I2,')=PLCG=',D15.9,' DPAR(',I2,')=W=',D15.9,/,
     + ' DPAR(',I2,')=PIYY=',D15.9,' DPAR(',I2,')=PB=',D15.9,
     + ' DPAR(',I2,')=PKM=',D15.9,/,
     + 34(' DPAR(',I2,')=',D20.9,/))
1007  FORMAT(1H1)
C
      CALL WHEN(DAYTIM)
C
C  IDP OFFSET OF BEGINING OF COMPUTED COST PARAMETERS
      IDP=30
      DO 17 I=1,25
17    IS(I)=BLANK
      DO 18 I=1,25
18    PAR(I)=DPAR(I)
C
20    WRITE(KTOUT,1002)
      IP=0
```

```
FILE: DQP      FORTRAN  A                  VM/SP CONVERSATIONAL MONITOR SYSTEM

      READ(KIN,*,ERR=20) IP
      IF (IP.EQ.-1) GOTO 23
      IF (IP.EQ.0) GOTO 30
      IF (IP.EQ.80) GOTO 22
      IF (IP.EQ.81) GOTO 221
      IF (IP.EQ.82) GOTO 222
      IF (IP.EQ.90) ITEST=0
      IF (IP.EQ.91) ITEST=1
      IF (IP.EQ.92) ITEST=2
      IF (IP.LT.0) GOTO 20
      IF (IP-20) 21,21,20
21    WRITE(KTOUT,1003) IP,PAR(IP)
      READ(KIN,1004,ERR=21) PAR(IP)
      IS(IP)=STAR
      IF (IP.EQ.21) GOTO 24
      IF (IP.EQ.22) GOTO 24
      IF (IP.EQ.23) GOTO 25
      IF (IP.EQ.24) GOTO 26
      IF (IP.EQ.25) GOTO 27
      GOTO 20
22    WRITE(KTOUT,1001) ((IS(I),PAR(I)),I=1,13)
      WRITE(KTOUT,10011)((IS(I),PAR(I)),I=14,20)
      GOTO 20
221   WRITE(KTOUT,1001) ((IS(I),PAR(I)),I=1,13)
      GOTO 20
222   WRITE(KTOUT,10011) ((IS(I),PAR(I)),I=14,20)
      GOTO 20
23    IEND=-1
      GOTO 90
C
24    PIP=PLCG*W/PIYY
      IS(11)=STAR
      GOTO 20
25    BR=(PB+PKM)/PIYY
      PIP=PLCG*W/PIYY
      BBR=PKM/PIYY
      DO 251 I=9,11
251   IS(I)=STAR
      GOTO 20
26    BR=(PB+PKM)/PIYY
      IS(9)=STAR
      GOTO 20
27    BR=(PB+PKM)/PIYY
      BBR=PKM/PIYY
      DO 271 I=9,10
271   IS(I)=STAR
      GOTO 20
C
30    CONTINUE
      WRITE(KTOUT,1001) ((IS(I),PAR(I)),I=1,13)
      WRITE(KTOUT,10011) ((IS(I),PAR(I)),I=14,20)
      WRITE(KOUT,1005) DAYTIM
      WRITE(KOUT,1001) ((IS(I),PAR(I)),I=1,13)
      WRITE(KOUT,10011) ((IS(I),PAR(I)),I=14,20)
      WRITE(KSOUT,1007)
      WRITE(KSOUT,1005) DAYTIM                                          DQP01660
      WRITE(KSOUT,1001) ((IS(I),PAR(I)),I=1,13)                         DQP01670
      WRITE(KSOUT,10011) ((IS(I),PAR(I)),I=14,20)                       DQP01680
      Q2=1/DSQRT(PAR(17)**2+1.D0)                                       DQP01690
      Q1=PAR(17)*Q2                                                     DQP01700
      R=RHO                                                             DQP01710
      RR0=PAR(18)                                                       DQP01720
      RR1=PAR(19)                                                       DQP01730
C                                                                       DQP01740
      DO 31 I=1,26                                                      DQP01750
31    DPAR(I)=PAR(I)                                                    DQP01760
      DPAR(IDP+1)=Q1                                                    DQP01770
      DPAR(IDP+2)=Q2                                                    DQP01780
      DPAR(IDP+3)=R                                                     DQP01790
      DPAR(IDP+4)=RR0                                                   DQP01800
      DPAR(IDP+5)=RR1                                                   DQP01810
      IF (ITEST.LE.0) GOTO 90                                           DQP01820
      WRITE(KOUT,1006) ((I,DPAR(I)),I=1,60)                             DQP01830
C                                                                       DQP01840
90    CONTINUE                                                          DQP01850
      RETURN                                                            DQP01860
      END                                                               DQP01870
```

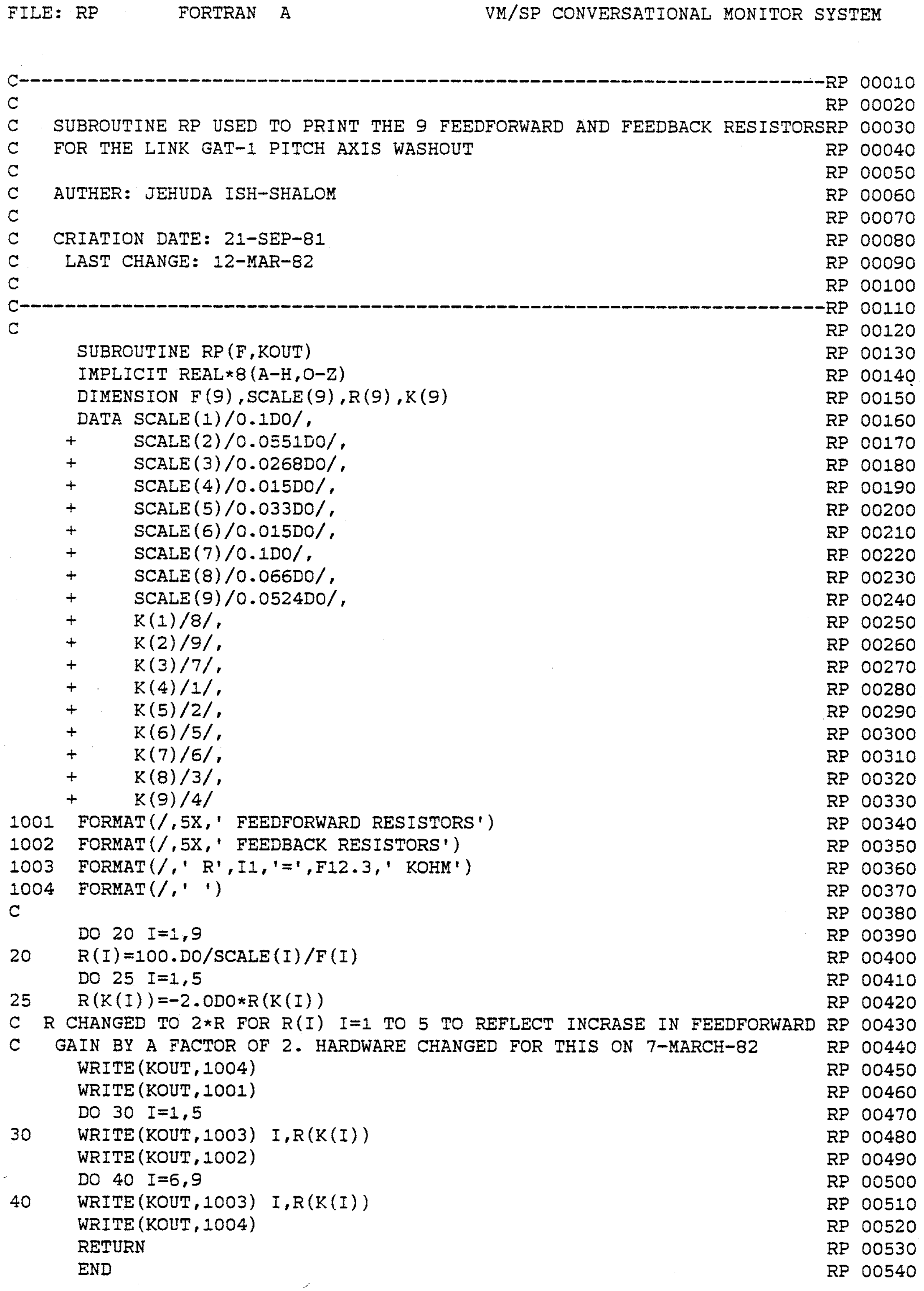
```
FILE: DMP      FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------DMP00010
C                                                                       DMP00020
C   SUBROUTINE THAT FILLS THE MATRICES FOR THE DESIGN OF THE            DMP00030
C              PITCH AXIS OF THE LINK GAT-1 FLIGHT SIMULATOR            DMP00040
C              (USING A VELOCITY CONTROL)                               DMP00050
C                                                                       DMP00060
C   AUTHER: JEHUDA ISH-SHALOM                                           DMP00070
C                                                                       DMP00080
C   CRIATION DATE: 27-MAR-81                                            DMP00090
C    LAST CHANGE: 26-FEB-82                                             DMP00100
C                                                                       DMP00110
C-----------------------------------------------------------------------DMP00120
      SUBROUTINE DMP(A,B,C,Q,R,N,NIN,NOUT,DPAR,ITEST)                   DMP00130
      IMPLICIT REAL*8(A-H,O-Z)                                          DMP00140
      DIMENSION A(N,N),B(N,NIN),C(NOUT,N),                              DMP00150
     + Q(N,N),R(NIN,NIN),DPAR(60),PAR(25),                              DMP00160
     + QQ(6,6),WORK(14)                                                 DMP00170
C      QQ(NOUT,NOUT),WORK(N)                                            DMP00180
C                                                                       DMP00190
      EQUIVALENCE (GO,PAR(1)),(BO,PAR(2)),(AO,PAR(3)),(GS,PAR(4)),      DMP00200
     + (AS,PAR(5)),(BNL,PAR(6)),(BNR,PAR(7)),(BNI,PAR(8)),              DMP00210
     + (BR,PAR(9)),(BBR,PAR(10)),(PIP,PAR(11)),(BI,PAR(12)),            DMP00220
     + (GEE,PAR(13)),                                                   DMP00230
     + (PKO,PAR(14)),(PKS,PAR(15)),(PQ,PAR(16))                         DMP00240
C                                                                       DMP00250
      COMMON/INOU/KIN,KOUT                                              DMP00260
C                                                                       DMP00270
C                                                                       DMP00280
      DO 20 I=1,26                                                      DMP00290
20    PAR(I)=DPAR(I)                                                    DMP00300
      IDP=30                                                            DMP00310
      Q1=DPAR(IDP+1)                                                    DMP00320
      Q2=DPAR(IDP+2)                                                    DMP00330
      R1=DPAR(IDP+3)                                                    DMP00340
      RR0=DPAR(IDP+4)                                                   DMP00350
      RR1=DPAR(IDP+5)                                                   DMP00360
C                                                                       DMP00370
      DO 21 I=1,N                                                       DMP00380
      DO 21 J=1,N                                                       DMP00390
21    A(I,J)=0.0D0                                                      DMP00400
      A(1,1)=-AO                                                        DMP00410
      A(1,7)=-(AO-BO)                                                   DMP00420
      A(1,8)=-GEE*A(1,7)                                                DMP00430
      A(2,2)=-AS                                                        DMP00440
      A(2,9)=-AS                                                        DMP00450
      A(3,3)=A(1,1)                                                     DMP00460
      A(3,5)=A(1,8)                                                     DMP00470
      A(4,4)=A(2,2)                                                     DMP00480
      A(4,6)=A(2,2)                                                     DMP00490
      A(5,5)=-BI                                                        DMP00500
      A(5,6)=1.D0                                                       DMP00510
      A(6,5)=PIP                                                        DMP00520
      A(6,6)=-BR                                                        DMP00530
      A(7,7)=-BNL                                                       DMP00540
      A(8,8)=-BNI                                                       DMP00550
      A(8,9)=1.D0                                                       DMP00560
      A(9,9)=-BNR                                                       DMP00570
      CALL TPRINT(N,N,N,A,'A AUGMENTED $',ITEST)                        DMP00580
C                                                                       DMP00590
      DO 22 I=1,N                                                       DMP00600
      DO 22 J=1,NIN                                                     DMP00610
22    B(I,J)=0.0D0                                                      DMP00620
      B(6,1)=BBR                                                        DMP00630
      CALL TPRINT(N,N,NIN,B,'B AUGMENTED $',ITEST)                      DMP00640
C                                                                       DMP00650
      DO 23 I=1,NOUT                                                    DMP00660
      DO 23 J=1,N                                                       DMP00670
23    C(I,J)=0.0D0                                                      DMP00680
      C(1,1)=PKO*GO                                                     DMP00690
      C(1,7)=C(1,1)                                                     DMP00700
      C(1,8)=-GEE*C(1,1)                                                DMP00710
      C(2,2)=PKS*GS                                                     DMP00720
      C(2,9)=C(2,2)                                                     DMP00730
      C(3,3)=GO                                                         DMP00740
      C(3,5)=-GEE*C(3,3)                                                DMP00750
      C(4,4)=GS                                                         DMP00760
      C(4,6)=C(4,4)                                                     DMP00770
      C(5,5)=1.D0                                                       DMP00780
      C(6,6)=1.D0                                                       DMP00790
C                                                                       DMP00800
      CALL TPRINT(NOUT,NOUT,N,C,'C AUGMENTED $',ITEST)                  DMP00810
C                                                                       DMP00820
      DO 25 I=1,NOUT                                                    DMP00830
      DO 25 J=1,NOUT                                                    DMP00840
25    QQ(I,J)=0.0D0                                                     DMP00850
      QQ(1,1)=Q1*(1.D0-PQ)                                              DMP00860
      QQ(2,2)=Q2*(1.D0-PQ)                                              DMP00870
      QQ(1,3)=-(1.D0+PQ)*Q1                                             DMP00880
      QQ(2,4)=-(1.D0+PQ)*Q2                                             DMP00890
      QQ(3,3)=QQ(1,1)                                                   DMP00900
      QQ(4,4)=QQ(2,2)                                                   DMP00910
      QQ(3,1)=QQ(1,3)                                                   DMP00920
      QQ(4,2)=QQ(2,4)                                                   DMP00930
      QQ(5,5)=RR0                                                       DMP00940
      QQ(6,6)=RR1                                                       DMP00950
      CALL TPRINT(NOUT,NOUT,NOUT,QQ,'QQ EXTENDED COST MATRIX $',ITEST)  DMP00960
C                                                                       DMP00970
      CALL MQF(NOUT,NOUT,N,NOUT,N,QQ,C,Q,WORK)                          DMP00980
      CALL TPRINT(N,N,N,Q,'R1 = Q IN PROGRAM $',ITEST)                  DMP00990
C                                                                       DMP01000
C     R12=0                                                             DMP01010
C                                                                       DMP01020
      R(1,1)=R1                                                         DMP01030
C                                                                       DMP01040
90    CONTINUE                                                          DMP01050
      RETURN                                                            DMP01060
      END                                                               DMP01070
```



```
FILE: DESP     FORTRAN  A                     VM/SP CONVERSATIONAL MONITOR SYSTEM


C---------------------------------------------------------------------------
C
C   SUBROUTINE THAT DESIGNS A TWO DEGREE OF FREEDOM WASHOUT FILTER
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CRIATION DATE: 14-APR-81
C
C---------------------------------------------------------------------------
      SUBROUTINE DESP(A,B,Q,R,ACL,F,N,NIN,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(N,N),B(N,NIN),R(NIN,NIN),Q(N,N),
     + ACL(N,N),F(NIN,N),
     + RK(9,9),
     + DUM(18,49),IDUM(18,3),WORK(9)
C      AEQ(N,N),QEQ(N,N),RK(N,N),G(NIN,N),G1(NIN,N),RINV(NIN,NIN),
C      ST(NIN,N),DUM(2*N,4+5*N),IDUM(2*N,3),WORK(N)
C
      COMMON/INOU/KIN,KOUT
C    UNITS USED     METERS. RADENS,SECANDS;
C                    OTHER UNIT ARE NORMALIZED
C
1001  FORMAT(1H1)
1002  FORMAT(' WASHOUT FILTER DESIGN')
      N2=N*2
      WRITE(KOUT,1001)
      IF (ITEST.GT.0) WRITE(KOUT,1002)
      CALL TPRINT(N,N,N,A,'A IN DESIGN SUB. $',ITEST)
      CALL TPRINT(N,N,NIN,B,'B IN DESIGN SUB. $',ITEST)
      CALL TPRINT(N,N,N,Q,'Q=R1 IN DESIGN SUB. $',ITEST)
      CALL TPRINT(NIN,NIN,NIN,R,'R=R2 IN DESIGN SUB. $',ITEST)
      CALL REG(N,NIN,N2,A,B,R,Q,RK,F,ACL,DUM,IDUM)
C
      IF (ITEST.GT.0) WRITE(KOUT,1003)
1003  FORMAT(/,/,/,' FEEDBACK MATRIX F')
      IF (ITEST.GT.0) CALL MATIO(NIN,NIN,N,F,3)
C
90    CONTINUE
      RETURN
      END
```

FILE: PFP   FORTRAN A   VM/SP CONVERSATIONAL MONITOR SYSTEM

```
C-----------------------------------------------------------------------
C
C   SUBROUTINE PFP USED TO PRINT THE FEEDBACK GAINS FA, FS, FM, FN
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CRIATION DATE: 14-APR-81
C   LAST CHANGED: 26-FEB-82
C
C-----------------------------------------------------------------------
C
      SUBROUTINE PFP(F,N,NIN,KOUT)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION F(NIN,N),LABLE(5),NBA(6)
      DATA LABLE(1)/'FA  '/,LABLE(2)/'FS  '/,LABLE(3)/'FM  '/,
     +                     LABLE(4)/'FN L'/,LABLE(5)/'FN R'/
1001  FORMAT(1H1,10X,' FEEDBACK GAINS')
1002  FORMAT(/,5X,A4)
1003  FORMAT(/,' ',3(D20.13,3X))
C
      NBLOCK=5
      NV=2
      NMR=2
      NNL=1
      NNR=2
      NBA(1)=0
      NBA(2)=NV
      NBA(3)=NBA(2)+NV
      NBA(4)=NBA(3)+NMR
      NBA(5)=NBA(4)+NNL
      NBA(6)=NBA(5)+NNR
      WRITE(KOUT,1001)
      DO 20 K=1,NBLOCK
      WRITE(KOUT,1002) LABLE(K)
      DO 20 I=1,NIN
      JO=NBA(K)+1
      JE=NBA(K+1)
 20   WRITE(KOUT ,1003) (F(I,J),J=JO,JE)
      RETURN
      END
```



```
FILE: RP       FORTRAN  A                 VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------
C
C   SUBROUTINE RP USED TO PRINT THE 9 FEEDFORWARD AND FEEDBACK RESISTORS
C   FOR THE LINK GAT-1 PITCH AXIS WASHOUT
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CRIATION DATE: 21-SEP-81
C    LAST CHANGE: 12-MAR-82
C
C-----------------------------------------------------------------------
C
      SUBROUTINE RP(F,KOUT)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION F(9),SCALE(9),R(9),K(9)
      DATA SCALE(1)/0.1D0/,
     +     SCALE(2)/0.0551D0/,
     +     SCALE(3)/0.0268D0/,
     +     SCALE(4)/0.015D0/,
     +     SCALE(5)/0.033D0/,
     +     SCALE(6)/0.015D0/,
     +     SCALE(7)/0.1D0/,
     +     SCALE(8)/0.066D0/,
     +     SCALE(9)/0.0524D0/,
     +     K(1)/8/,
     +     K(2)/9/,
     +     K(3)/7/,
     +     K(4)/1/,
     +     K(5)/2/,
     +     K(6)/5/,
     +     K(7)/6/,
     +     K(8)/3/,
     +     K(9)/4/
1001  FORMAT(/,5X,' FEEDFORWARD RESISTORS')
1002  FORMAT(/,5X,' FEEDBACK RESISTORS')
1003  FORMAT(/,' R',I1,'=',F12.3,' KOHM')
1004  FORMAT(/,' ')
C
      DO 20 I=1,9
20    R(I)=100.D0/SCALE(I)/F(I)
      DO 25 I=1,5
25    R(K(I))=-2.0D0*R(K(I))
C  R CHANGED TO 2*R FOR R(I) I=1 TO 5 TO REFLECT INCRASE IN FEEDFORWARD
C   GAIN BY A FACTOR OF 2. HARDWARE CHANGED FOR THIS ON 7-MARCH-82
      WRITE(KOUT,1004)
      WRITE(KOUT,1001)
      DO 30 I=1,5
30    WRITE(KOUT,1003) I,R(K(I))
      WRITE(KOUT,1002)
      DO 40 I=6,9
40    WRITE(KOUT,1003) I,R(K(I))
      WRITE(KOUT,1004)
      RETURN
      END
```

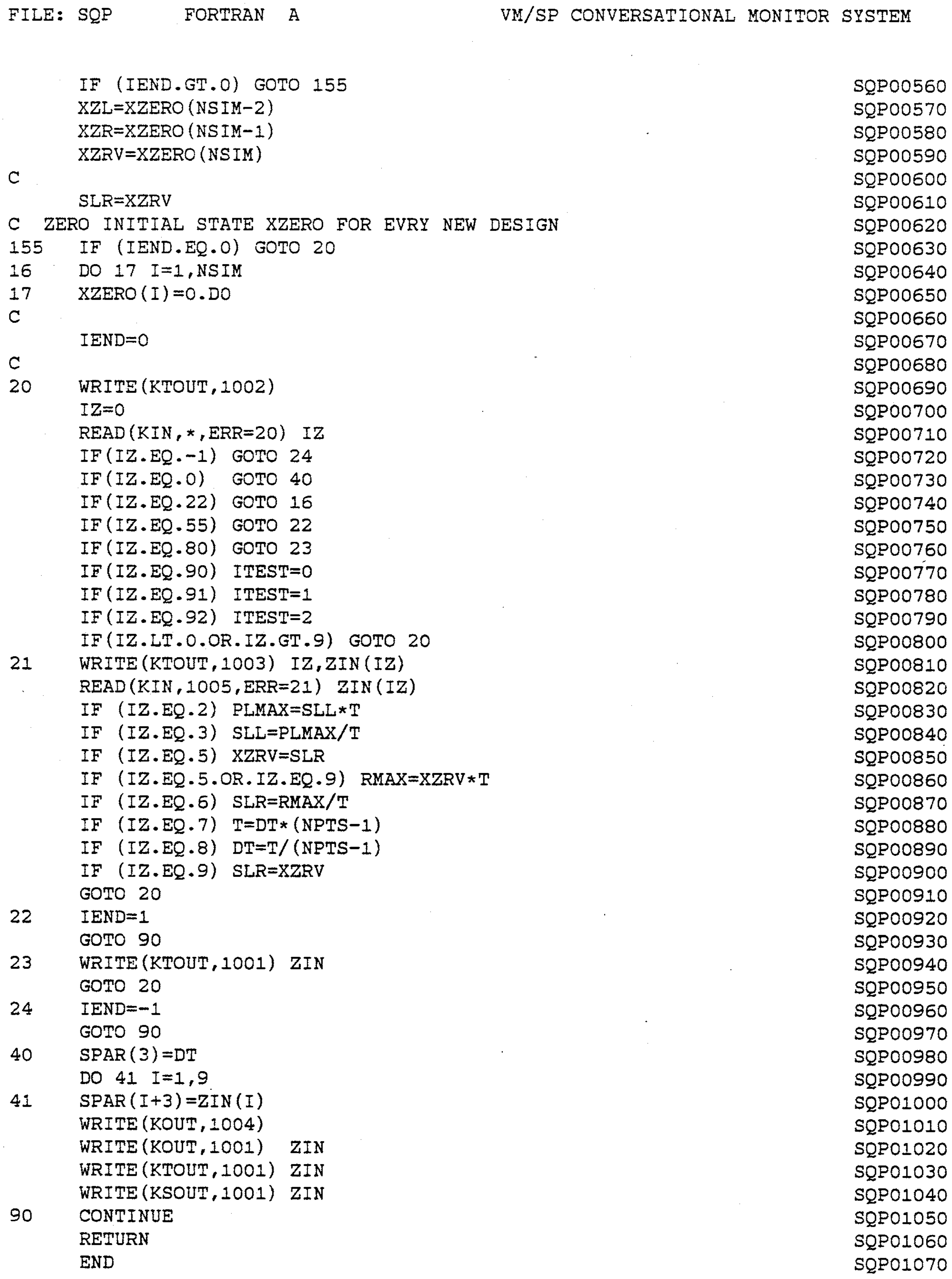
```
C---------------------------------------------------------------------
C  FOR PITCH AXIS OF LINK GAT-1 FLIGHT SIMULATOR
C
C  CHECK IF THE DESCRITIZED SYSTEM IN CLOSED LOOP IS STABLE
C  (AT SAMPLING INTERVAL DTW). ALSO CHECKS IF SYSTEM (AWCLD0)
C  IS STILL STABLE WHEN FORCING ZERO VESTIBULAR ERROR:
C    F0(1,1)=0, F0(2,1)=0, F0(1,2)=0, F0(2,2)=0
C    F0(1,3)=0, F0(2,3)=0, F0(1,4)=0, F0(2,5)=0,
C              F0(1,7)=F(1,7)-G0, F0(1,8)=F(1,8)+GEE*G0
C  MAKES BODE PLOTS OF WASHOUT FILTER  AND COMPUTS THE DC GAINS
C
C  INPUT: MATRICES A,B,F; SCALARS DTW,NIN,NW,N
C         DTW - SAMPLING TIME
C         NIN - # WASHOUT OUTPUTS = # OF CONTROL INPUTS TO THE SIMULATOR
C         NW - DIMANTION OF WASHOUT FILTER
C         N - DIMANTION OF AUGMENTED A SYSTEM MATRIX
C      NN=N-NW = # WASHOUT INPUTS= # INPUT STATES FORM AIRPLANE SIMULATION
C  OUTPUT: MATRICES AWD,BWD,CWD,DWD,AWCLD,AWCLD0
C  CONTROL INPUTS: KTOUT,ITEST
C
C  AUTHOR: JEHUDA ISH-SHALOM
C
C  CREATION DATE: 24-APR-81
C
C  LAST CHANGED: 30-OCT-81
C---------------------------------------------------------------------
C
      SUBROUTINE STP(ACL,B,F,DTW,AWD,BWD,CWD,DWD,AWCLD,AWCLD0,DPAR,
     + N,NIN,NW,NN,KTOUT,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ACL(N,N),B(N,NIN),F(NIN,N),
     + AWD(NW,NW),BWD(NW,NIN),CWD(NN,NW),DWD(NN,NN),
     + AWCLD(NW,NW),AWCLD0(NW,NW),DPAR(60),
     + F0(1,6),AW(6,6),BW(6,3),BW1(6,1),BW2(6,1),BW3(6,1),TWDC(3,3),
     + WR(6),WI(6),FV1(6),IV1(6),DUM1(3,3),DUM(6,20),IDUM(6),
     + DUM3(6,19),DUM4(51,6)
C   AW(NW,NW),BW(NW,NN),BW1(NW,1),BW2(NW,1),BW3(NW,1),TWDC(NN,NN),
C   WR(NW),WI(NW),FV1(NW),IV1(NW),DUM1(NN,NN),DUM(NW,2*(1+NW+NN)),
C   IDUM(NW),DUM3(NW,1+3*NW),DUM4(NPD*NDEC+1,6)
C
      COMMON/INOU/KIN,KOUT
C
C   NN= NUMBER OF INPUTS TO THE WASHOUT FILTER
      NN=N-NW
C
10011 FORMAT(/,' ERROR CODE FROM EIGVAL SUB.(AW COMPUTATION)=',I16)
10012 FORMAT(/,' ERROR CODE FROM EIGVAL SUB.(AWCLD COMPUTATION)=',I16)
10013 FORMAT(/,' ERROR CODE FROM EIGVAL SUB.(AWCLD0 COMPUTATION)=',I16)
10021 FORMAT(/,/,' EIGEN VALUES OF AW')
10022 FORMAT(/,/,' EIGEN VALUES OF DESCRITIZED SYSTEM - CLOSED LOOP')
10023 FORMAT(/,/,' EIGEN VALUES OF D. SYS - CLOSED LOOP, VEST ERR=0')
10032 FORMAT(' ',6X,'REAL PART',13X,'IMAG PART',13X,
     + 'NAT FREQ(RAD/SEC)',3X,'AWCLD',/,I2,3(2X,D20.14))
10033 FORMAT(' ',6X,'REAL PART',13X,'IMAG PART',13X,
     + 'NAT FREQ(RAD/SEC)',3X,'AWCLD0',/,I2,3(2X,D20.14))
```

```
FILE: STP      FORTRAN  A                  VM/SP CONVERSATIONAL MONITOR SYSTEM


1005    FORMAT(' ENTER SAMPLING TIME(SEC): ')                                  STP00560
1006    FORMAT(D20.9)                                                           STP00570
1007    FORMAT(/,' SAMPLING TIME(SEC)= ',F16.14)                                STP00580
1008    FORMAT(/,/,' DC GAIN FOR ACCELERATION, ANGLE, ANGULER VEL',/,/,        STP00590
     +   3(D20.14,2X))                                                          STP00600
1009    FORMAT(1H1,' WASHOUT BODE PLOT FOR ACC INPUT (G) OUT (RAD/SEC)')        STP00610
1010    FORMAT(1H1,' WASHOUT BODE PLOT FOR D_TETA INPUT,UNITS (RAD/SEC)')STP00620
C                                                                               STP00630
        GO=DPAR(1)                                                              STP00640
        GS=DPAR(4)                                                              STP00650
        GEE=DPAR(13)                                                            STP00660
C                                                                               STP00670
        DO 20 I=1,NW                                                            STP00680
        DO 20 J=1,NW                                                            STP00690
20      AW(I,J)=ACL(I,J)                                                        STP00700
        AW(5,5)=0.D0                                                            STP00710
        DO 21 I=1,NW                                                            STP00720
        DO 21 J=1,NN                                                            STP00730
21      BW(I,J)=ACL(I,J+NW)                                                     STP00740
        DO 22 J=1,NW                                                            STP00750
        CWD(1,J)=-F(1,J)                                                        STP00760
        DO 22 I=2,NN                                                            STP00770
22      CWD(I,J)=0.D0                                                           STP00780
        CWD(2,5)=1.D0                                                           STP00790
        CWD(3,6)=1.D0                                                           STP00800
        DO 23 I=1,NIN                                                           STP00810
        DO 23 J=1,NN                                                            STP00820
23      DWD(I,J)=-F(I,J+NW)                                                     STP00830
        DO 24 I=2,3                                                             STP00840
        DO 24 J=1,NN                                                            STP00850
24      DWD(I,J)=0.D0                                                           STP00860
C                                                                               STP00870
        CALL TPRINT(NW,NW,NW,AW,'AW TOP LEFT OF NWXNW OF ACL $',ITEST)         STP00880
        CALL TPRINT(NW,NW,NN,BW,'BW TOP NW ROWS NW+1 TO N COL OF ACL $',       STP00890
     +  ITEST)                                                                  STP00900
        CALL TPRINT(NN,NN,NW,CWD,'CW=CWD OF WASHOUT FILTER $',ITEST)            STP00910
        CALL TPRINT(NN,NN,NN,DWD,'DW=DWD OF WASHOUT FILTER $',ITEST)            STP00920
C                                                                               STP00930
C   CALL TO MSCALE ADDED TO FIXE UP ERROR IN DCGAIN SUB.                        STP00940
C    COMPUTES D+C.A**-1 .B INSTEAD OF D-C.A**-1 .B                              STP00950
C                                                                               STP00960
        CALL MSCALE(NW,NW,NN,-1.0D0,BW)                                         STP00970
        CALL DCGAIN(NW,NN,AW,BW,CWD,DWD,TWDC,DUM1,IDUM)                         STP00980
        CALL MSCALE(NW,NW,NN,-1.0D0,BW)                                         STP00990
C   RESATING THE VALUE OF MATRIX BW                                             STP01000
C                                                                               STP01010
        CALL TPRINT(NN,NN,NN,TWDC,'DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:COSTP01020
     +M,ANGLE,D_ANGLE $',1)                                                     STP01030
        CALL TPRINT(NN,NN,NN,TWDC,'DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:COSTP01040
     +M,ANGLE,D_ANGLE $',2)                                                     STP01050
C                                                                               STP01060
        DO 25 I=1,NW                                                            STP01070
        BW1(I,1)=BW(I,1)                                                        STP01080
        BW2(I,1)=BW(I,2)                                                        STP01090
25      BW3(I,1)=BW(I,3)                                                        STP01100
```

```
C
C      CALL DCGAIN(NW,NIN,AW,BW1,CWD,DWD(1,1),TWDC(1,1),DUM1,IDUM)
C      CALL DCGAIN(NW,NIN,AW,BW2,CWD,DWD(1,2),TWDC(1,2),DUM1,IDUM)
C      CALL DCGAIN(NW,NIN,AW,BW3,CWD,DWD(1,3),TWDC(1,3),DUM1,IDUM)
C      WRITE(KOUT,1008)  TWDC
C      WRITE(KTOUT,1008) TWDC
C
C      IP=1000
C      NPD=10
C      NDEC=5
C      WZERO=0.01
C      NPTS=51
C      WRITE(KOUT,1009)
C      CALL BNIN(NW,NPTS,AW,BW1,CWD,DWD(1,1),WZERO,NDEC,NPD,IP,DUM3,
C    +   IDUM,DUM4)
C      WRITE(KOUT,1010)
C      CALL BNIN(NW,NPTS,AW,BW3,CWD,DWD(1,3),WZERO,NDEC,NPD,IP,DUM3,
C    +   IDUM,DUM4)
C30     WRITE(KTOUT,1005)
C      READ(KIN,1006,ERR=30) DTW
C      IF (DTW.EQ.0.D0) GOTO 90
C      WRITE(KOUT,1007) DTW
C      CALL DLIN(NW,NIN,AW,BW,AWD,BWD,DTW,DUM,IDUM)
C      CALL TPRINT(NW,NW,NW,AWD,'AWD $',ITEST)
C      CALL TPRINT(NW,NW,NIN,BWD,'BWD $',ITEST)
C      CALL MMUL(NW,NIN,NW,NW,NW,NIN,BWD,CWD,AWCLD)
C      CALL TPRINT(NW,NW,NW,AWCLD,'-BWD.F $',ITEST)
C      CALL MMUL(NW,NIN,NW,NW,NW,NIN,BWD,F0,AWCLD0)
C      CALL TPRINT(NW,NW,NW,AWCLD0,'-BWD.F0 FOR 0 VESTIBULAR FEEDBACK $'
C    +    ,ITEST)
C      DO 31 I=1,NW
C      DO 31 J=1,NW
C      AWCLD(I,J)=AWD(I,J)+AWCLD(I,J)
C31     AWCLD0(I,J)=AWD(I,J)+AWCLD0(I,J)
C      CALL TPRINT(NW,NW,NW,AWCLD,'AWCLD $',ITEST)
C      CALL TPRINT(NW,NW,NW,AWCLD0,'AWCLD0 $',ITEST)
C
C      IF (ITEST.EQ.0) GOTO 50
C      WRITE(KOUT,10021)
C  DUM DISTROYED BY EIGVAL
C      CALL SAVE(NW,NW,NW,NW,AW,DUM)
C      CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)
C      IF (IERR.NE.0) WRITE(KOUT,10011) IERR
C      CALL SAVE(NW,NW,NW,NW,AWD,DUM)
C      CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)
C
C50     WRITE(KOUT,10022)
C      CALL SAVE(NW,NW,NW,NW,AWCLD,DUM)
C      CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)
C      IF (IERR.EQ.0) GOTO 51
C      WRITE(KTOUT,10012) IERR
C      WRITE(KOUT,10012) IERR
C51     DO 52 I=1,NW
C      FREQN=DSQRT(WR(I)**2+WI(I)**2)
C52     IF (FREQN.GT.1.D0) WRITE(KTOUT,10032) I,WR(I),WI(I),FREQN
```

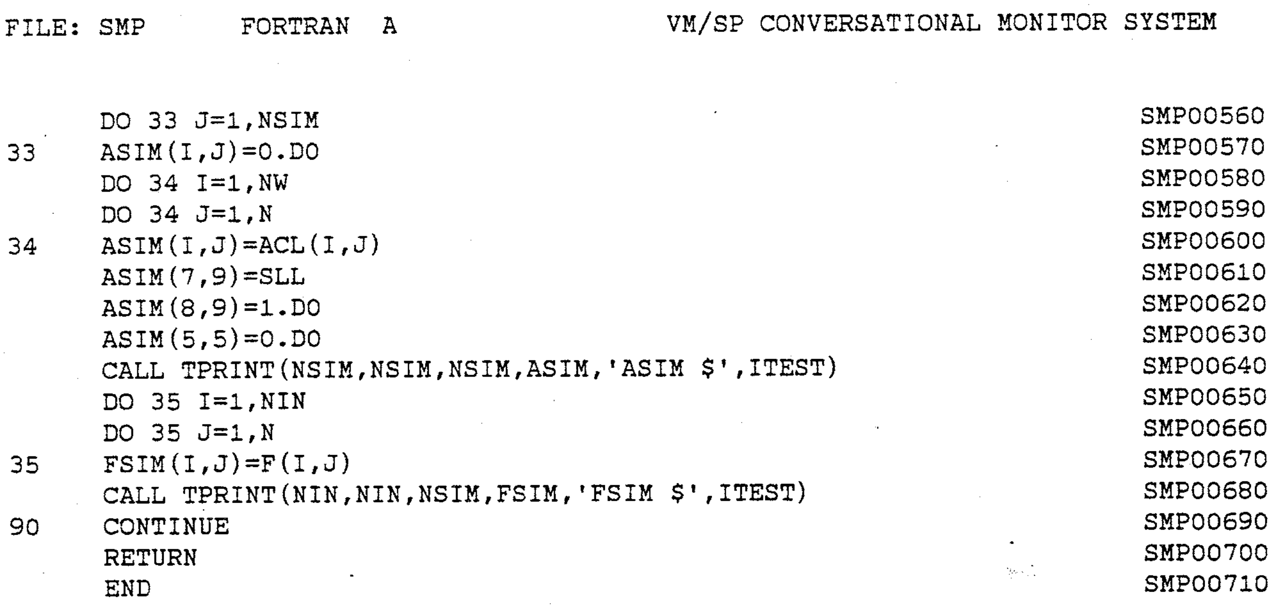
```
FILE: STP      FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


C                                                                       STP01660
C       WRITE(KOUT,10023)                                               STP01670
C       CALL SAVE(NW,NW,NW,NW,AWCLD0,DUM)                               STP01680
C       CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)                       STP01690
C       IF (IERR.EQ.0) GOTO 61                                          STP01700
C       WRITE(KTOUT,10013) IERR                                         STP01710
C       WRITE(KOUT,10013) IERR                                          STP01720
C61     DO 62 I=1,NW                                                    STP01730
C       FREQN=DSQRT(WR(I)**2+WI(I)**2)                                  STP01740
C62     IF (FREQN.GT.1.D0) WRITE(KTOUT,10033) I,WR(I),WI(I),FREQN       STP01750
C                                                                       STP01760
C       GOTO 30                                                         STP01770
90      CONTINUE                                                        STP01780
        RETURN                                                          STP01790
        END                                                             STP01800
```

```
C-----------------------------------------------------------------------
C
C   SUBROUTINE THAT ENTERS THE SIMULATION PARAMETERS
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   INPUT: KTOUT,ITEST,IEND
C     KTOUT=TERMINAL UNIT #
C     ITEST- IF ITEST>0 PRINT DEBUGING STATMENTS
C     IEND-  IF IEND>0 => NEW DESIGN
C            IF IEND=0 => CONTINUE TO SIMULATION
C            IF IEND<0 => EXIT (END THE PROGRAM)
C
C   OUTPUT: SPAR(60) A PARAMETER ARRAY
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 14-APR-81
C
C   LAST CHANGE: 01-MAR-82
C-----------------------------------------------------------------------
C
      SUBROUTINE SQP(SPAR,XZERO,NSIM,KTOUT,KSOUT,ITEST,IEND)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION SPAR(60),ZIN(9),XZERO(NSIM)
      EQUIVALENCE (XZL,ZIN(1)),(SLL,ZIN(2)),(PLMAX,ZIN(3)),
     + (XZR,ZIN(4)),(SLR,ZIN(5)),(RMAX,ZIN(6)),
     + (DT,ZIN(7)),(T,ZIN(8)),(XZRV,ZIN(9))
C
      COMMON/INOU/KIN,KOUT
C
C    UNITS USED     METERS. RADENS,SECANDS;
C                    OTHER UNIT ARE NORMALIZED
C
1001  FORMAT(//,' SIMULATION CONDITIONS',/,
     + '  1  XZL=',F8.3,' |G S|',/,
     + '  2  SLL=',F8.3,' |G S/SEC|',/,
     + '  3  PLMAX=',F6.3,'|G S|',/,
     + '  4  XZR=',F8.3,' |RAD|',/,
     + '  5  SLR=',F8.3,' |RAD/SEC|',/,
     + '  6  RMAX=',F7.3,' |RAD|',/,
     + '  7  DT=',F9.3,' [SEC]',/,
     + '  8  T=',F10.3,' [SEC]',/,
     + '  9  XZRV=',F8.3,' |RAD/SEC|',/)
1002  FORMAT(' (-1 EXIT;0 CONTINUE;',
     +  '22 NEW SIM;55 NEW DESIGN;80 PRINT;90,91,92 TEST)',/,
     +  ' CHANGE PARAMETER # : ')
1003  FORMAT(' ZIN(',I2,')=',D20.15,' NEW VALUE:')
1004  FORMAT(1H1)
1005  FORMAT(D20.9)
C
      NPTS=IDINT(SPAR(1)+0.5)
      NPRPL=IDINT(SPAR(2)+0.5)
      DO 15 I=1,9
15    ZIN(I)=SPAR(I+3)
C  SKIP PRISEVING STATES IF NEW DESIGN
```



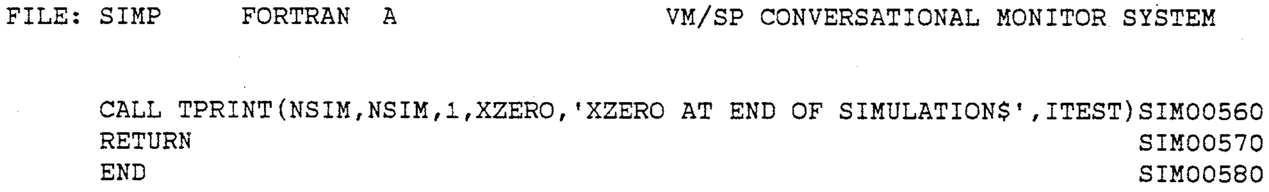
```
FILE: SQP      FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


      IF (IEND.GT.0) GOTO 155                                           SQP00560
      XZL=XZERO(NSIM-2)                                                 SQP00570
      XZR=XZERO(NSIM-1)                                                 SQP00580
      XZRV=XZERO(NSIM)                                                  SQP00590
C                                                                       SQP00600
      SLR=XZRV                                                          SQP00610
C  ZERO INITIAL STATE XZERO FOR EVRY NEW DESIGN                         SQP00620
155   IF (IEND.EQ.0) GOTO 20                                            SQP00630
16    DO 17 I=1,NSIM                                                    SQP00640
17    XZERO(I)=0.D0                                                     SQP00650
C                                                                       SQP00660
      IEND=0                                                            SQP00670
C                                                                       SQP00680
20    WRITE(KTOUT,1002)                                                 SQP00690
      IZ=0                                                              SQP00700
      READ(KIN,*,ERR=20) IZ                                             SQP00710
      IF(IZ.EQ.-1) GOTO 24                                              SQP00720
      IF(IZ.EQ.0)  GOTO 40                                              SQP00730
      IF(IZ.EQ.22) GOTO 16                                              SQP00740
      IF(IZ.EQ.55) GOTO 22                                              SQP00750
      IF(IZ.EQ.80) GOTO 23                                              SQP00760
      IF(IZ.EQ.90) ITEST=0                                              SQP00770
      IF(IZ.EQ.91) ITEST=1                                              SQP00780
      IF(IZ.EQ.92) ITEST=2                                              SQP00790
      IF(IZ.LT.0.OR.IZ.GT.9) GOTO 20                                    SQP00800
21    WRITE(KTOUT,1003) IZ,ZIN(IZ)                                      SQP00810
      READ(KIN,1005,ERR=21) ZIN(IZ)                                     SQP00820
      IF (IZ.EQ.2) PLMAX=SLL*T                                          SQP00830
      IF (IZ.EQ.3) SLL=PLMAX/T                                          SQP00840
      IF (IZ.EQ.5) XZRV=SLR                                             SQP00850
      IF (IZ.EQ.5.OR.IZ.EQ.9) RMAX=XZRV*T                               SQP00860
      IF (IZ.EQ.6) SLR=RMAX/T                                           SQP00870
      IF (IZ.EQ.7) T=DT*(NPTS-1)                                        SQP00880
      IF (IZ.EQ.8) DT=T/(NPTS-1)                                        SQP00890
      IF (IZ.EQ.9) SLR=XZRV                                             SQP00900
      GOTO 20                                                           SQP00910
22    IEND=1                                                            SQP00920
      GOTO 90                                                           SQP00930
23    WRITE(KTOUT,1001) ZIN                                             SQP00940
      GOTO 20                                                           SQP00950
24    IEND=-1                                                           SQP00960
      GOTO 90                                                           SQP00970
40    SPAR(3)=DT                                                        SQP00980
      DO 41 I=1,9                                                       SQP00990
41    SPAR(I+3)=ZIN(I)                                                  SQP01000
      WRITE(KOUT,1004)                                                  SQP01010
      WRITE(KOUT,1001)  ZIN                                             SQP01020
      WRITE(KTOUT,1001) ZIN                                             SQP01030
      WRITE(KSOUT,1001) ZIN                                             SQP01040
90    CONTINUE                                                          SQP01050
      RETURN                                                            SQP01060
      END                                                               SQP01070
```

```
C-----------------------------------------------------------------------
C
C   SUBROUTINE THAT PUTS THE SIMULATION PARAMETERS INTO THIER RIGHT
C     PLACE IN THE SIMULATION MATRICES
C     FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 15-APR-81
C
C-----------------------------------------------------------------------
      SUBROUTINE SMP(ACL,C,D,F,ASIM,CSIM,FSIM,XZERO,DPAR,SPAR,
     + N,NIN,NOUT,NSIM,NSOUT,NW,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ACL(N,N),C(NOUT,N),D(NOUT,NIN),F(NIN,N),
     + DPAR(60),SPAR(60),
     + ASIM(NSIM,NSIM),CSIM(NSOUT,NSIM),FSIM(NIN,NSIM),XZERO(NSIM)
C
      COMMON/INOU/KIN,KOUT
C
C    UNITS USED     METERS. RADENS,SECANDS;
C                    OTHER UNIT ARE NORMALIZED
C
      GO=DPAR(1)
      BO=DPAR(2)
      AO=DPAR(3)
      GEE=DPAR(13)
      PKO=DPAR(14)
      PKS=DPAR(15)
      BNL=DPAR(6)
      BNR=DPAR(7)
      SLL=SPAR(5)
      SLR=SPAR(8)
      XZL=SPAR(4)
      XZR=SPAR(7)
      XZRV=SPAR(12)
      XZERO(7)=XZL
      XZERO(8)=XZR
      XZERO(9)=XZRV
C
      DO 31 I=1,N
      DO 31 J=1,NSIM
31    CSIM(I,J)=0.0D0
      DO 32 J=1,N
      CSIM(1,J)=C(1,J)/PKO-C(3,J)
      CSIM(2,J)=C(2,J)/PKS-C(4,J)
      CSIM(3,J)=C(5,J)
      CSIM(4,J)=C(6,J)
      CSIM(5,J)=C(1,J)/PKO
      CSIM(6,J)=C(3,J)
      CSIM(7,J)=C(2,J)/PKS
32    CSIM(8,J)=C(4,J)
C
      CALL TPRINT(NSOUT,NSOUT,NSIM,CSIM,'CSIM $',ITEST)
      DO 33 I=1,NSIM
```



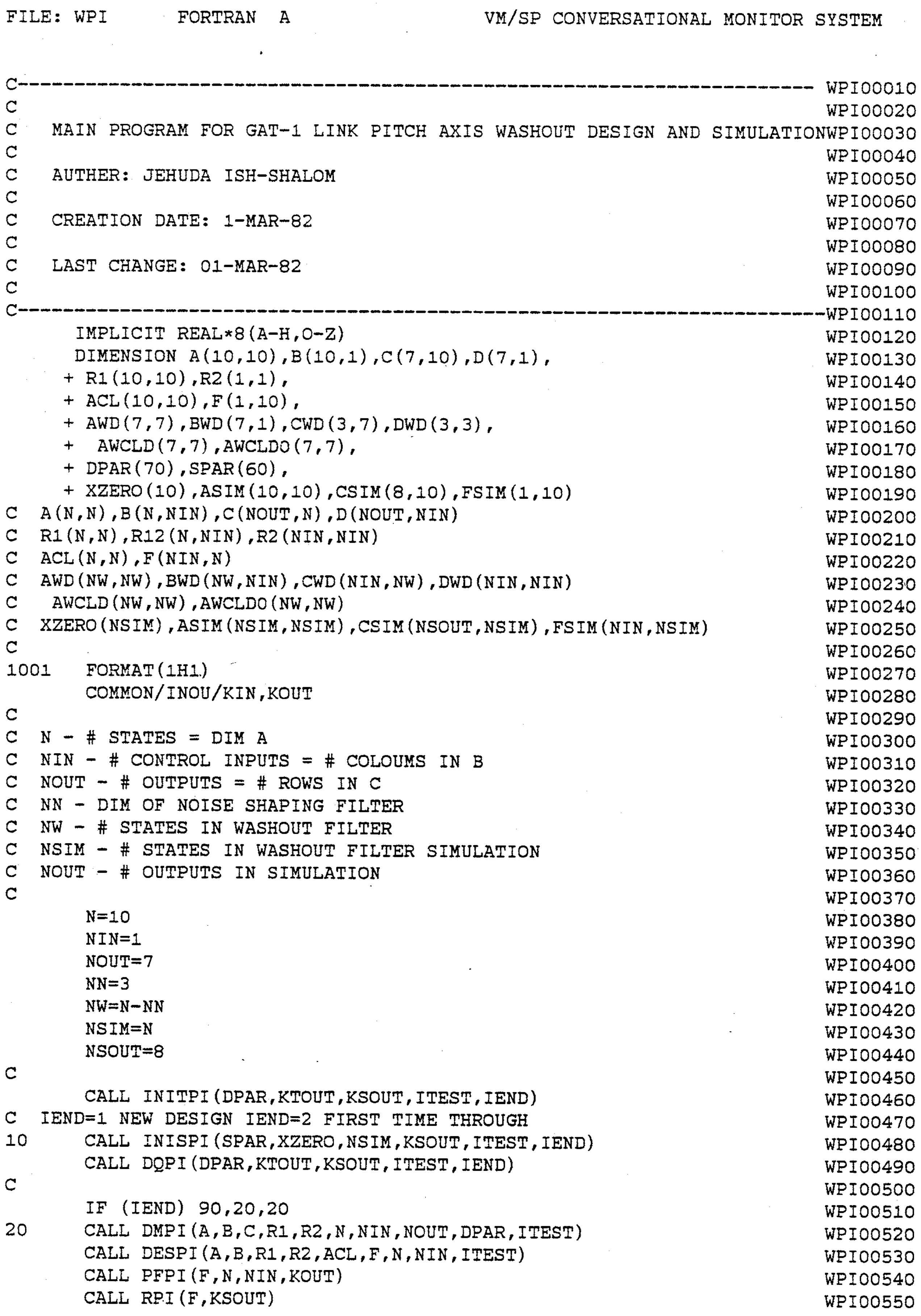
```
FILE: SMP      FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


      DO 33 J=1,NSIM                                                    SMP00560
33    ASIM(I,J)=0.D0                                                    SMP00570
      DO 34 I=1,NW                                                      SMP00580
      DO 34 J=1,N                                                       SMP00590
34    ASIM(I,J)=ACL(I,J)                                                SMP00600
      ASIM(7,9)=SLL                                                     SMP00610
      ASIM(8,9)=1.D0                                                    SMP00620
      ASIM(5,5)=0.D0                                                    SMP00630
      CALL TPRINT(NSIM,NSIM,NSIM,ASIM,'ASIM $',ITEST)                   SMP00640
      DO 35 I=1,NIN                                                     SMP00650
      DO 35 J=1,N                                                       SMP00660
35    FSIM(I,J)=F(I,J)                                                  SMP00670
      CALL TPRINT(NIN,NIN,NSIM,FSIM,'FSIM $',ITEST)                     SMP00680
90    CONTINUE                                                          SMP00690
      RETURN                                                            SMP00700
      END                                                               SMP00710
```

```
C-----------------------------------------------------------------------
C
C   SUBROUTINE THAT SIMULATS THE DESIGNED WASHOUT FILETER
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 15-APR-81
C
C   INPUT:
C     SYSTEM MATRICES:    ASIM(NSIM,CSIM(NSOUT,NSIM),FSIM(NIN,NSIM)
C     INITIAL CONDITIONS: XZERO(NSIM)
C     SYSTEM DIMENTIONS:  NSIM,NIN,NSOUT
C     OTHER SIMULATION PARAMETERS: SPAR(60)
C     TEST FLAG: ITEST (PRINT OUT TEST PRINTOUT IF ITEST>0)
C
C   OUTPUT:
C     PRINTPLOTS OF OUTPUTS AND CONTROLS(COMPUTED FROM FSIM)
C
C   SUBROUTINES CALLED:
C     MATIO,REGSIM
C
C-----------------------------------------------------------------------
      SUBROUTINE SIMP(ASIM,CSIM,FSIM,XZERO,SPAR,NSIM,NIN,NSOUT,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ASIM(NSIM,NSIM),CSIM(NSOUT,NSIM),FSIM(NIN,NSIM),
     + XZERO(NSIM),SPAR(60),
     + DUM(9,46),IDUM(9),DUM2(51,9)
C  DUM(NSIM,(1+5*NSIM)),IDUM(NSIM),DUM2(NPTS,NIN+NSOUT)
C
      COMMON/INOU/KIN,KOUT
C
C  UNITS USED     METERS, RADENS, SECONDS,
C                 OTHER NORMALIZED UNITS
C
1001  FORMAT(1H1)
1002  FORMAT(/,/,' ORDER OF OUTPUTS Y(I)',/,
     + ' Y(1)=EOTO,    Y(2)=ESCC,    Y(3)=TETAM,  Y(4)=D_TETAM',/,
     + ' Y(5)=YAOTO,   Y(6)=YSOTO,   Y(7)=YASCC,  Y(8)=YSSCC')
1003  FORMAT(/,' NPTS= ',I5,' NPRPL= ',I3,' DT= ',D20.15,' (SEC)')
C
      NPTS=IDINT(SPAR(1)+0.5D0)
      NPRPL=IDINT(SPAR(2)+0.5D0)
      DT=SPAR(3)
      IF (ITEST.GT.0) WRITE(KOUT,1003) NPTS,NPRPL,DT
      CALL TPRINT(NSIM,NSIM,NSIM,ASIM,'ASIM IN SIM SUB.$',ITEST)
      CALL TPRINT(NSOUT,NSOUT,NSIM,CSIM,'CSIM IN SIM SUB.$',ITEST)
      CALL TPRINT(NIN,NIN,NSIM,FSIM,'FSIM IN SIM SUB.$',ITEST)
      WRITE(KOUT,1002)
      CALL MATIO(NSIM,NSIM,1,XZERO,3)
C
      CALL REGSIM(NSIM,NSOUT,NIN,ASIM,CSIM,FSIM,XZERO,DT,NPTS,NPRPL,
     +  DUM,IDUM,DUM2)
C
      WRITE(KOUT,1001)
```



```
FILE: SIMP      FORTRAN  A                  VM/SP CONVERSATIONAL MONITOR SYSTEM


      CALL TPRINT(NSIM,NSIM,1,XZERO,'XZERO AT END OF SIMULATION$',ITEST)SIM00560
      RETURN                                                            SIM00570
      END                                                               SIM00580
```

```
FILE: WPI      EXEC     A                   VM/SP CONVERSATIONAL MONITOR SYSTEM


FILEDEF 7 TERM
FILEDEF 5 TERM
FILEDEF 6 DISK O DATA E (RECFM F LRECL 132 BLKSIZE 132
FILEDEF 8 DISK S DATA E (RECFM F LRECL 132 BLKSIZE 132
LOAD WPI (NOMAP START
PRINT S DATA E (CC)
PRINT O DATA E (CC)
CP SPOOL PRINT CLOSE
QPRINT S DATA E (SIDES 2 CC
COST
```



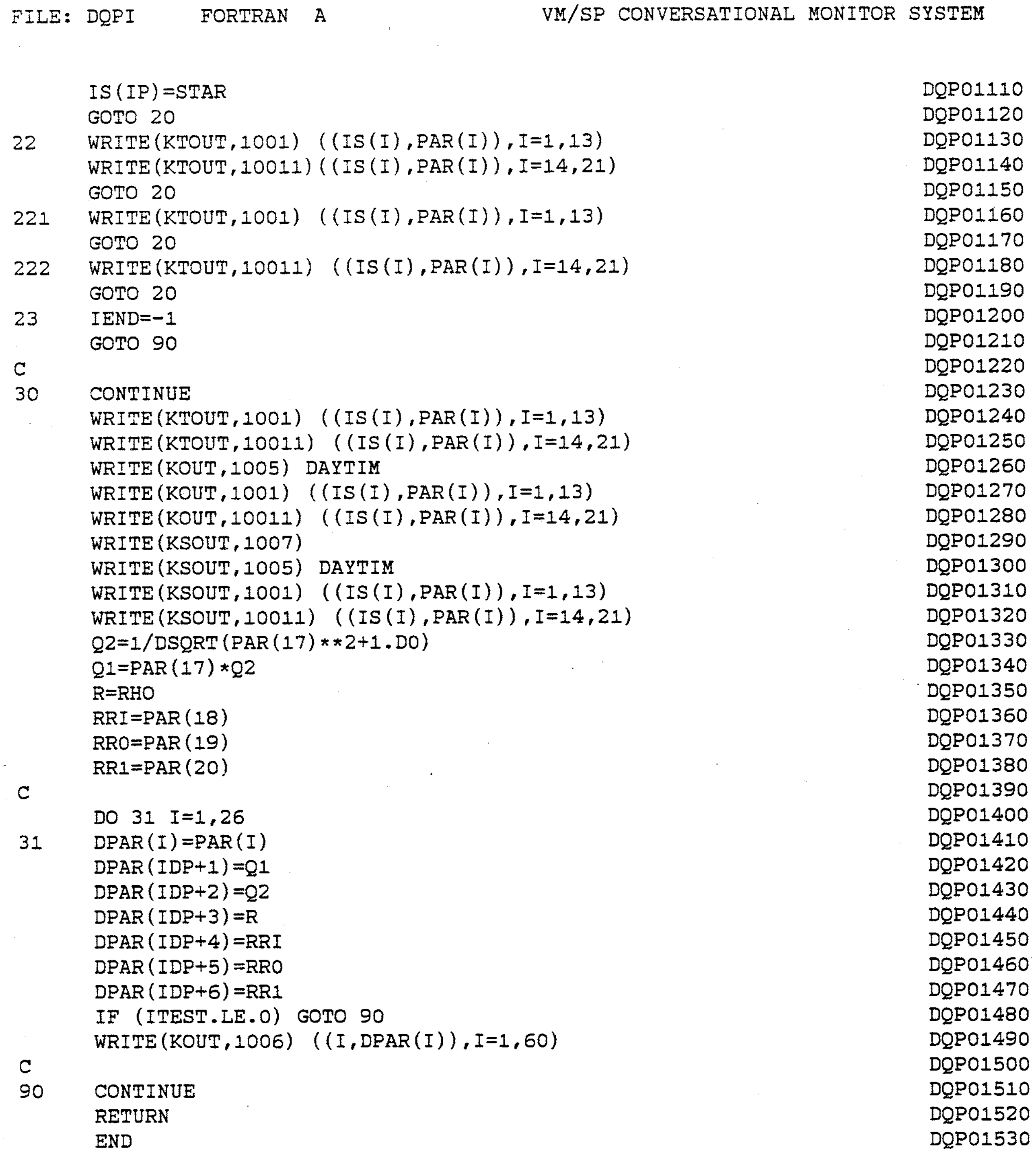
```
FILE: WPI      FORTRAN  A                 VM/SP CONVERSATIONAL MONITOR SYSTEM


C--------------------------------------------------------------------
C
C   MAIN PROGRAM FOR GAT-1 LINK PITCH AXIS WASHOUT DESIGN AND SIMULATION
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 1-MAR-82
C
C   LAST CHANGE: 01-MAR-82
C
C--------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(10,10),B(10,1),C(7,10),D(7,1),
     + R1(10,10),R2(1,1),
     + ACL(10,10),F(1,10),
     + AWD(7,7),BWD(7,1),CWD(3,7),DWD(3,3),
     +  AWCLD(7,7),AWCLDO(7,7),
     + DPAR(70),SPAR(60),
     + XZERO(10),ASIM(10,10),CSIM(8,10),FSIM(1,10)
C  A(N,N),B(N,NIN),C(NOUT,N),D(NOUT,NIN)
C  R1(N,N),R12(N,NIN),R2(NIN,NIN)
C  ACL(N,N),F(NIN,N)
C  AWD(NW,NW),BWD(NW,NIN),CWD(NIN,NW),DWD(NIN,NIN)
C   AWCLD(NW,NW),AWCLDO(NW,NW)
C  XZERO(NSIM),ASIM(NSIM,NSIM),CSIM(NSOUT,NSIM),FSIM(NIN,NSIM)
C
1001   FORMAT(1H1)
       COMMON/INOU/KIN,KOUT
C
C  N - # STATES = DIM A
C  NIN - # CONTROL INPUTS = # COLOUMS IN B
C  NOUT - # OUTPUTS = # ROWS IN C
C  NN - DIM OF NOISE SHAPING FILTER
C  NW - # STATES IN WASHOUT FILTER
C  NSIM - # STATES IN WASHOUT FILTER SIMULATION
C  NOUT - # OUTPUTS IN SIMULATION
C
       N=10
       NIN=1
       NOUT=7
       NN=3
       NW=N-NN
       NSIM=N
       NSOUT=8
C
       CALL INITPI(DPAR,KTOUT,KSOUT,ITEST,IEND)
C  IEND=1 NEW DESIGN IEND=2 FIRST TIME THROUGH
10     CALL INISPI(SPAR,XZERO,NSIM,KSOUT,ITEST,IEND)
       CALL DQPI(DPAR,KTOUT,KSOUT,ITEST,IEND)
C
       IF (IEND) 90,20,20
20     CALL DMPI(A,B,C,R1,R2,N,NIN,NOUT,DPAR,ITEST)
       CALL DESPI(A,B,R1,R2,ACL,F,N,NIN,ITEST)
       CALL PFPI(F,N,NIN,KOUT)
       CALL RPI(F,KSOUT)
      WRITE(KSOUT,1001)
      CALL PFPI(F,N,NIN,KTOUT)
      CALL RPI(F,KOUT)
C  CHECK FOR ERROR (ITEST<0)
      IF (ITEST.LT.0) GOTO 90
      CALL STPI(ACL,B,F,DTW,AWD,BWD,CWD,DWD,AWCLD,AWCLDO,DPAR,
     +  N,NIN,NW,NN,KTOUT,ITEST)
30    CALL SQP(SPAR,XZERO,NSIM,KTOUT,KSOUT,ITEST,IEND)
C  IEND=0 CONTINUE, IEND>0 NEW DESIGN, IEND<0 END
      IF (IEND) 90,40,10
40    CALL SMPI(ACL,C,D,F,ASIM,CSIM,FSIM,XZERO,DPAR,SPAR,
     +  N,NIN,NOUT,NSIM,NSOUT,NW,ITEST)
      CALL SIMPI(ASIM,CSIM,FSIM,XZERO,SPAR,NSIM,NIN,NSOUT,ITEST)
C  IF ITEST<0 ERROR IN SIMULATION => END PROGRAM
      IF (ITEST.GE.0) GOTO 30
90    CONTINUE
      END
```

```
FILE: INITPI   FORTRAN  A                 VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------INI00010
C                                                                       INI00020
C   SUBROUTINE TO INITIALIZE QUERY VARIABELS AND SIMULATION STATE       INI00030
C   FOR THE LINK GAT-1 PITCH AXIS                                       INI00040
C                                                                       INI00050
C   INPUT: ITEST                                                        INI00060
C     ITEST - IF ITEST>0 PRINT DEBUGING PRINTOUTS                       INI00070
C                                                                       INI00080
C   OUTPUT: DPAR(60),KTOUT,KSOUT,IEND                                   INI00090
C     DPAR - DESIGN PARAMETERS                                          INI00100
C     KTOUT - FILE NUMBER OF TERMINAL                                   INI00110
C     KSOUT - FILE NUMBER OF SUMMERY FILE                               INI00120
C     IEND  - =2 TO SIGNAL NEW DESIGN THE FIRST TIME THROUGH TO SQP SUB.INI00130
C                                                                       INI00140
C   AUTHOR: JEHUDA ISH-SHALOM                                           INI00150
C                                                                       INI00160
C   CREATION DATE: 1-MAR-82                                             INI00170
C   LAST CHANGED: 1-MAR-82                                              INI00180
C                                                                       INI00190
C-----------------------------------------------------------------------INI00200
C                                                                       INI00210
      SUBROUTINE INITPI(DPAR,KTOUT,KSOUT,ITEST,IEND)                    INI00220
      IMPLICIT REAL*8(A-H,O-Z)                                          INI00230
      DIMENSION DPAR(60),SPAR(60),PAR(25),ZIN(9)                        INI00240
      EQUIVALENCE (GO,PAR(1)),(BO,PAR(2)),(AO,PAR(3)),(GS,PAR(4)),      INI00250
     + (AS,PAR(5)),(BNL,PAR(6)),(BNR,PAR(7)),(BNI,PAR(8)),(BR,PAR(9)),  INI00260
     + (BBR,PAR(10)),(PIP,PAR(11)),(BI,PAR(12)),                        INI00270
     + (GEE,PAR(13)),(PKO,PAR(14)),(PKS,PAR(15)),(PQ,PAR(16)),          INI00280
     + (RHO,PAR(21))                                                    INI00290
C                                                                       INI00300
      COMMON/INOU/KIN,KOUT                                              INI00310
      KIN=5                                                             INI00320
      KOUT=6                                                            INI00330
      KTOUT=7                                                           INI00340
      KSOUT=8                                                           INI00350
C                                                                       INI00360
C     FILE ASSIGNMENT                                                   INI00370
C                                                                       INI00380
C     FILE 5: INPUT FROM THE TERMINAL (KIN)                             INI00390
C     FILE 6: OUTPUT TO THE LINE PRINTER (KOUT)                         INI00400
C     FILE 7: OUTPUT TO THE TERMINAL  (KTOUT)                           INI00410
C     FILE 8: SUMMERY OUTPUT TO THE LINE PRINTER (KSOUT)                INI00420
C                                                                       INI00430
C    UNITS USED     G'S,RADENS,SECANDS;                                 INI00440
C                    OTHER UNIT ARE NORMALIZED                          INI00450
C                                                                       INI00460
1001  FORMAT(' SCALERS EXITING SUB. INIT',/,' IEND= ',I3,' ITEST= ',I3, INI00470
     + ' KTOUT= ',I3,' KIN= ',I3,' KOUT= ',I3,' KSOUT= ',I3)            INI00480
1002  FORMAT(1H1,' GAT-1 LINK PITCH AXIS WASHOUT DESIGN PROGRAM',/,/,   INI00490
     + 4X,' WRITTEN BY JEHUDA ISH-SHALOM (MARCH 1982)',/)               INI00500
1003  FORMAT(/,' MOTION BASE COMMAND INPUT: ANGULER VELOCITY',/,        INI00510
     + ' MEASURED STATES: ANGULER POSTION AND VELOCITY, AND LINEAR',    INI00520
     + ' ACCELERATION IN MOTION BASE INIRIAL AXES',/,                   INI00530
     + ' VESTIBULAR MODEL INPUTS:',/,                                   INI00540
     + 10X,' LINEAR  = OTOLITH : LINEAR ACCELERATION',/,                INI00550
     + 10X,' ANGULER = SEMICIRCULAR CANAL: ANGULER VELOCITY',/)
C
      WRITE(KTOUT,1002)
      WRITE(KOUT,1002)
      WRITE(KOUT,1003)
C
      ITEST=0
      IEND=2
C    OTOLITH MODEL  OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD).
      GO=21.168DO
      BO=.076DO
      AO=.19DO
C    SEMICIRCULAR CANAL MODEL OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD).
      AS=1/5.9DO
      GS=40.DO
C
      GEE=-1.DO
C
C   NOISE FILTER
C
      BNL=0.1DO
      BNR=0.1DO
      BNI=1.D-5
C   MOTION BASE
      BR=3.13DO
      BBR=0.167DO
      PIP=0.74DO
      BI=0.DO
C   COST VALUES
      PKO=1.DO
      PKS=1.DO
      PQ=0.DO
      PAR(17)=1.DO
C
      PAR(18)=0.DO
      PAR(19)=0.DO
      PAR(20)=0.DO
C
      RHO=0.04DO
C
      DO 25 I=1,60
25    DPAR(I)=0.
      DO 30 I=1,21
30    DPAR(I)=PAR(I)
C
      CALL TPRINT(1,1,60,DPAR,'DPAR AFTER SUB. INIT $',ITEST)
      IF (ITEST.GT.0) WRITE(KOUT,1001) IEND,ITEST,KTOUT,KIN,KOUT,KSOUT
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C
C   SUBROUTINE TO INITIALIZE QUERY SIMULATION VARIABELS AND STATES
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   INPUT: NSIM,ITEST,IEND
C     NSIM - DIMESION OF SIMULATION MATRIX ASIM
C     ITEST - IF ITEST>0 PRINT DEBUGING PRINTOUTS
C     IEND - IF IEND¬= 1 PRINT NEW SIMULATION IN FILE KSOUT
C
C   OUTPUT: SPAR(60),XZERO(NSIM)
C     SPAR - SIMULATION PARAMETERS
C     XZERO - SIMULATION INITIAL STATE
C
C   AUTHOR: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 1-MAR-82
C   LAST CHANGED: 01-MAR-82
C
C-----------------------------------------------------------------------
C
      SUBROUTINE INISPI(SPAR,XZERO,NSIM,KSOUT,ITEST,IEND)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION SPAR(60),PAR(25),ZIN(9),XZERO(NSIM)
      EQUIVALENCE
     + (XZL,ZIN(1)),(SLL,ZIN(2)),(PLMAX,ZIN(3)),
     + (XZR,ZIN(4)),(SLR,ZIN(5)),(RMAX,ZIN(6)),
     + (DT,ZIN(7)),(T,ZIN(8)),(XZRV,ZIN(9))
      COMMON /INOU/KIN,KOUT
C
C    UNITS USED     G'S,RADENS,SECANDS;
C                    OTHER UNIT ARE NORMALIZED
C
1001  FORMAT(' NEW SIMULATION',//)
      IF (IEND.EQ.1) WRITE(KSOUT,1001)
1002  FORMAT(' SCALARS EXITING SUB. INITSP',/,
     + ' IEND= ',I3,' ITEST= ',I3,' NSIM= ',I3,' NPTS= ',I5,' NPRPL= ',
     + I2,' DT= ',D20.15,' T= ',D20.15)
C
      XZL=0.2D0
      SLL=0.D0
      PLMAX=0.D0
      XZR=0.D0
      SLR=0.D0
      RMAX=0.D0
      XZRV=0.D0
      DO 20 I=1,NSIM
20    XZERO(I)=0.D0
      XZERO(8)=XZL
      XZERO(9)=XZR
      XZERO(10)=XZRV
C
      DT=.2D0
      NPTS=51
C  NPRPL > 0 PRINT AND PLOT
```

```
C  NPRPL = 0 PRINT ONLY
C  NPRPL < 9 PLOT ONLY
      NPRPL=-10
      T=DT*(NPTS-1)
C
      DO 25 I=1,60
25    SPAR(I)=0.
      SPAR(1)=DFLOAT(NPTS)
      SPAR(2)=DFLOAT(NPRPL)
      SPAR(3)=DT
      DO 32 I=1,9
32    SPAR(I+3)=ZIN(I)
C
      CALL TPRINT(1,1,60,SPAR,'SPAR AFTER SUB. INIT $',ITEST)
      CALL TPRINT(NSIM,NSIM,1,XZERO,'XZERO AFTER SUB. INIT $',ITEST)
      IF (ITEST.GT.0) WRITE(KOUT,1002) IEND,ITEST,NSIM,NPTS,NPRPL,DT,T
      RETURN
      END
```

```
FILE: DQPI     FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------DQP00010
C   SUBROUTINE THAT QUERYS THE PARAMETERS FOR THE WASHOUT DESIGN        DQP00020
C              FOR THE PITCH AXIS OF THE LINK GAT-1 FLIGHT SIMULATOR    DQP00030
C                                                                       DQP00040
C   AUTHER: JEHUDA ISH-SHALOM                                           DQP00050
C                                                                       DQP00060
C   CREATION DATE: 1-MAR-82                                             DQP00070
C   LAST CHANGE: 01-MAR-82                                              DQP00080
C                                                                       DQP00090
C-----------------------------------------------------------------------DQP00100
      SUBROUTINE DQPI(DPAR,KTOUT,KSOUT,ITEST,IEND)                      DQP00110
      IMPLICIT REAL*8(A-H,O-Z)                                          DQP00120
      DIMENSION PAR(25),DPAR(60)                                        DQP00130
      INTEGER*2 STAR,BLANK,IS(25)                                       DQP00140
      REAL*4 DAYTIM(5)                                                  DQP00150
C   DAYTIM HOLDS THE DAY AND TIME AFTER THE CALL WHEN(DAYTIM)           DQP00160
C      MO/DY/YR HR*MN*SC.SC(*10**-2)                                    DQP00170
C                                                                       DQP00180
      EQUIVALENCE (GO,PAR(1)),(BO,PAR(2)),(AO,PAR(3)),(GS,PAR(4)),      DQP00190
     + (AS,PAR(5)),(BNL,PAR(6)),(BNR,PAR(7)),(BNI,PAR(8)),              DQP00200
     + (BR,PAR(9)),(BBR,PAR(10)),(PIP,PAR(11)),(BI,PAR(12)),            DQP00210
     + (GEE,PAR(13)),(PKO,PAR(14)),(PKS,PAR(15)),(PQ,PAR(16)),          DQP00220
     + (RHO,PAR(21))                                                    DQP00230
C                                                                       DQP00240
      COMMON/INOU/KIN,KOUT                                              DQP00250
C                                                                       DQP00260
      DATA STAR/2H */,BLANK/2H  /                                       DQP00270
C                                                                       DQP00280
C    UNITS USED     METERS. RADENS,SECANDS;                             DQP00290
C                    OTHER UNIT ARE NORMALIZED                          DQP00300
C                                                                       DQP00310
C  FORMAT STATMETS                                                      DQP00320
1001  FORMAT(' OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK',    DQP00330
     + ' LOAD)',/,' (THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))',/,           DQP00340
     + A2,'  1  GO=',F9.3,' (1/G)',/,                                   DQP00350
     + A2,'  2  BO=',F9.3,' (RAD/SEC)',/,                               DQP00360
     + A2,'  3  AO=',F9.3,' (RAD/SEC)',/,                               DQP00370
     + ' SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS',          DQP00380
     + ' (WITH WORK LOAD)',/,                                           DQP00390
     + ' (THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))',/,               DQP00400
     + A2,'  4  GS=',F9.3,' (1/RAD)',/,                                 DQP00410
     + A2,'  5  AS=',F9.3,' (RAD/SEC)',/,                               DQP00420
     + ' NOISE FILTER',/,                                               DQP00430
     + A2,'  6  BNL=',F8.3,' (RAD/SEC)',/,                              DQP00440
     + A2,'  7  BNR=',F8.3,' (RAD/SEC)',/,                              DQP00450
     + A2,'  8  BNI=',D8.2,' (1/SEC)',/,                                DQP00460
     + ' MOTION BASE MODEL',/,                                          DQP00470
     + A2,'  9  BR=',F9.3,' (1/SEC)',/,                                 DQP00480
     + A2,'  10 BBR=',F8.3,' (RAD/(SEC**2*VOLT))',/,                    DQP00490
     + A2,'  11 PIP=',D8.2,' (1/SEC**2)',/,                             DQP00500
     + A2,'  12 BI=',D9.2,' (1/SEC)',/,' GRAVITY CONSTANT',/,           DQP00510
     + A2,'  13 GEE=',F8.3,' (G)')                                      DQP00520
10011 FORMAT( ' COST FUNCTION PRAMETERS',/,                             DQP00530
     + '  ERROR',/,                                                     DQP00540
     + A2,'  14 PKO=',F12.10,' OTO SCALING OF REQ. RESPONSE PKO>0',/,   DQP00550
     + A2,'  15 PKS=',F12.10,' SCC SCALING OF REQ. RESPONSE PKS>0',/,
     + A2,'  16 PQ=',F13.10,' PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL',/,
     + A2,'  17 QO/QS=',D10.2,' OTOLITH/SEMICIRCULAR CANAL',/,
     + '  ROTATION MOTIONS',/,
     + A2,'  18 RRINT=',D10.2,' ANGLE INTEGRAL (RAD/SEC)',/,
     + A2,'  19 RR0  =',D10.2,' ANGLE',/,
     + A2,'  20 RR1  =',D10.2,' VELOCITY',/,
     + '  GLOBAL SCALING',/,
     + A2,'  21 RHO=',D12.2,' WEIGHT OF MOTION OVER ERROR',/)
1002  FORMAT(' CHANGE PAR # (-1 EXIT; 0 CONTINUE; 80,81,82 PRINT;',
     + '90,91,92 TEST) : ')
1003  FORMAT(' PAR(',I2,')=',D20.9,' NEW VALUE :')
1004  FORMAT(D20.9)
1005  FORMAT(/,' WAHOUT FILTER DESIGN PARAMETERS (',5A4,')',/)
1006  FORMAT(' DPAR(I)',/,
     + ' DPAR(',I2,')=GO=',D15.9,' DPAR(',I2,')=BO=',D15.9,
     + ' DPAR(',I2,')=AO=',D15.9,/,
     + ' DPAR(',I2,')=GS=',D15.9,' DPAR(',I2,')=AS=',D15.9,/,
     + ' DPAR(',I2,')=BNL=',D15.9,' DPAR(',I2,')=BNR=',D15.9,
     + ' DPAR(',I2,')=BNI=',D15.9,/,
     + ' DPAR(',I2,')=BR=',D15.9,' DPAR(',I2,')=BBR=',D15.9,
     + ' DPAR(',I2,')=PIP=',D15.9,' DPAR(',I2,')=BI=',D15.9,/,
     + ' DPAR(',I2,')=GEE=',D15.9,/,
     + ' DPAR(',I2,')=PKO=',D15.9,' DPAR(',I2,')=PKS=',D15.9,
     + ' DPAR(',I2,')=PQ=',D15.9,/,
     + ' DPAR(',I2,')=QO/QS=',D15.9,' DPAR(',I2,')=RRINT=',D15.9,
     + ' DPAR(',I2,')=RR0  =',D15.9,' DPAR(',I2,')=RR1  =',D15.9,/,
     + ' DPAR(',I2,')=RHO=',D15.9,/,
     + 38(' DPAR(',I2,')=',D20.9,/))
1007  FORMAT(1H1)
C
      CALL WHEN(DAYTIM)
C
C  IDP OFFSET OF BEGINING OF COMPUTED COST PARAMETERS
      IDP=30
      DO 17 I=1,25
17    IS(I)=BLANK
      DO 18 I=1,25
18    PAR(I)=DPAR(I)
C
20    WRITE(KTOUT,1002)
      IP=0
      READ(KIN,*,ERR=20) IP
      IF (IP.EQ.-1) GOTO 23
      IF (IP.EQ.0) GOTO 30
      IF (IP.EQ.80) GOTO 22
      IF (IP.EQ.81) GOTO 221
      IF (IP.EQ.82) GOTO 222
      IF (IP.EQ.90) ITEST=0
      IF (IP.EQ.91) ITEST=1
      IF (IP.EQ.92) ITEST=2
      IF (IP.LT.0) GOTO 20
      IF (IP-21) 21,21,20
21    WRITE(KTOUT,1003) IP,PAR(IP)
      READ(KIN,1004,ERR=21) PAR(IP)
```

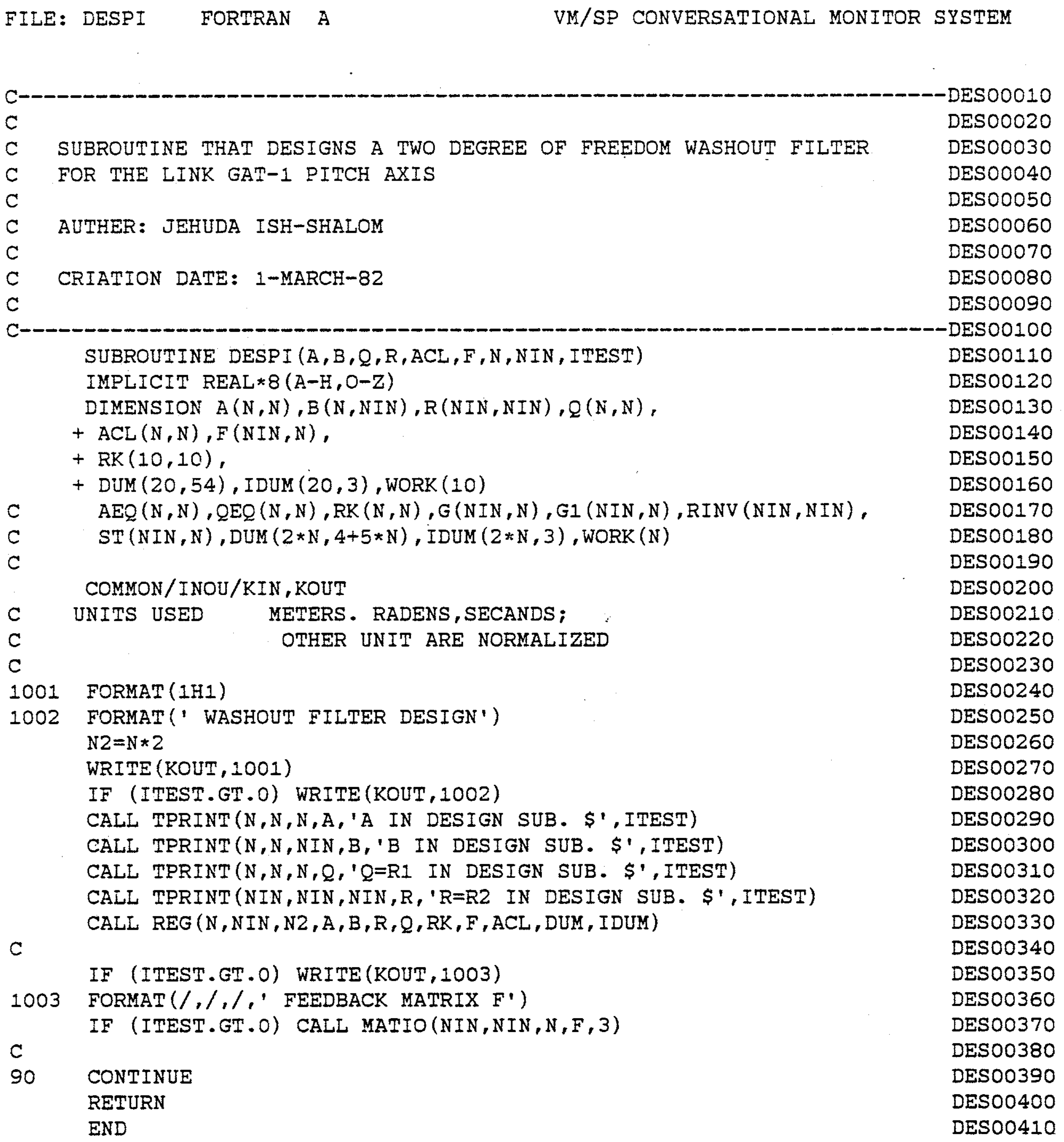


```
FILE: DQPI     FORTRAN  A              VM/SP CONVERSATIONAL MONITOR SYSTEM


      IS(IP)=STAR
      GOTO 20
22    WRITE(KTOUT,1001) ((IS(I),PAR(I)),I=1,13)
      WRITE(KTOUT,10011)((IS(I),PAR(I)),I=14,21)
      GOTO 20
221   WRITE(KTOUT,1001) ((IS(I),PAR(I)),I=1,13)
      GOTO 20
222   WRITE(KTOUT,10011) ((IS(I),PAR(I)),I=14,21)
      GOTO 20
23    IEND=-1
      GOTO 90
C
30    CONTINUE
      WRITE(KTOUT,1001) ((IS(I),PAR(I)),I=1,13)
      WRITE(KTOUT,10011) ((IS(I),PAR(I)),I=14,21)
      WRITE(KOUT,1005) DAYTIM
      WRITE(KOUT,1001) ((IS(I),PAR(I)),I=1,13)
      WRITE(KOUT,10011) ((IS(I),PAR(I)),I=14,21)
      WRITE(KSOUT,1007)
      WRITE(KSOUT,1005) DAYTIM
      WRITE(KSOUT,1001) ((IS(I),PAR(I)),I=1,13)
      WRITE(KSOUT,10011) ((IS(I),PAR(I)),I=14,21)
      Q2=1/DSQRT(PAR(17)**2+1.D0)
      Q1=PAR(17)*Q2
      R=RHO
      RRI=PAR(18)
      RR0=PAR(19)
      RR1=PAR(20)
C
      DO 31 I=1,26
31    DPAR(I)=PAR(I)
      DPAR(IDP+1)=Q1
      DPAR(IDP+2)=Q2
      DPAR(IDP+3)=R
      DPAR(IDP+4)=RRI
      DPAR(IDP+5)=RR0
      DPAR(IDP+6)=RR1
      IF (ITEST.LE.0) GOTO 90
      WRITE(KOUT,1006) ((I,DPAR(I)),I=1,60)
C
90    CONTINUE
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C
C   SUBROUTINE THAT FILLS THE MATRICES FOR THE DESIGN OF THE
C              PITCH AXIS OF THE LINK GAT-1 FLIGHT SIMULATOR
C              (USING A VELOCITY CONTROL)
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CRIATION DATE: 1-MAR-82
C    LAST CHANGE: 12-MAR-82
C
C-----------------------------------------------------------------------
      SUBROUTINE DMPI(A,B,C,Q,R,N,NIN,NOUT,DPAR,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(N,N),B(N,NIN),C(NOUT,N),
     + Q(N,N),R(NIN,NIN),DPAR(60),PAR(25),
     + QQ(7,7),WORK(14)
C       QQ(NOUT,NOUT),WORK(N)
C
      EQUIVALENCE (GO,PAR(1)),(BO,PAR(2)),(AO,PAR(3)),(GS,PAR(4)),
     + (AS,PAR(5)),(BNL,PAR(6)),(BNR,PAR(7)),(BNI,PAR(8)),
     + (BR,PAR(9)),(BBR,PAR(10)),(PIP,PAR(11)),(BI,PAR(12)),
     + (GEE,PAR(13)),
     + (PKO,PAR(14)),(PKS,PAR(15)),(PQ,PAR(16))
C
      COMMON/INOU/KIN,KOUT
C
C
      DO 20 I=1,26
20    PAR(I)=DPAR(I)
      IDP=30
      Q1=DPAR(IDP+1)
      Q2=DPAR(IDP+2)
      R1=DPAR(IDP+3)
      RRI=DPAR(IDP+4)
      RRO=DPAR(IDP+5)
      RR1=DPAR(IDP+6)
C
      DO 21 I=1,N
      DO 21 J=1,N
21    A(I,J)=0.0D0
      A(1,1)=-AO
      A(1,8)=-(AO-BO)
      A(1,9)=-GEE*A(1,8)
      A(2,2)=-AS
      A(2,10)=-AS
      A(3,3)=A(1,1)
      A(3,6)=A(1,9)
      A(4,4)=A(2,2)
      A(4,7)=A(2,2)
      A(5,5)=-0.1D0*BNI
      A(5,6)=1.D0
      A(6,6)=-BI
      A(6,7)=1.D0
      A(7,6)=PIP
```

```
FILE: DMPI     FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


      A(7,7)=-BR                                                        DMP00560
      A(8,8)=-BNL                                                       DMP00570
      A(9,9)=-BNI                                                       DMP00580
      A(9,10)=1.D0                                                      DMP00590
      A(10,10)=-BNR                                                     DMP00600
      CALL TPRINT(N,N,N,A,'A AUGMENTED $',ITEST)                        DMP00610
C                                                                       DMP00620
      DO 22 I=1,N                                                       DMP00630
      DO 22 J=1,NIN                                                     DMP00640
22    B(I,J)=0.0D0                                                      DMP00650
      B(7,1)=BBR                                                        DMP00660
      CALL TPRINT(N,N,NIN,B,'B AUGMENTED $',ITEST)                      DMP00670
C                                                                       DMP00680
      DO 23 I=1,NOUT                                                    DMP00690
      DO 23 J=1,N                                                       DMP00700
23    C(I,J)=0.0D0                                                      DMP00710
      C(1,1)=PKO*GO                                                     DMP00720
      C(1,8)=C(1,1)                                                     DMP00730
      C(1,9)=-GEE*C(1,1)                                                DMP00740
      C(2,2)=PKS*GS                                                     DMP00750
      C(2,10)=C(2,2)                                                    DMP00760
      C(3,3)=GO                                                         DMP00770
      C(3,6)=-GEE*C(3,3)                                                DMP00780
      C(4,4)=GS                                                         DMP00790
      C(4,7)=C(4,4)                                                     DMP00800
      C(5,5)=1.D0                                                       DMP00810
      C(6,6)=1.D0                                                       DMP00820
      C(7,7)=1.D0                                                       DMP00830
C                                                                       DMP00840
      CALL TPRINT(NOUT,NOUT,N,C,'C AUGMENTED $',ITEST)                  DMP00850
C                                                                       DMP00860
      DO 25 I=1,NOUT                                                    DMP00870
      DO 25 J=1,NOUT                                                    DMP00880
25    QQ(I,J)=0.0D0                                                     DMP00890
      QQ(1,1)=Q1*(1.D0-PQ)                                              DMP00900
      QQ(2,2)=Q2*(1.D0-PQ)                                              DMP00910
      QQ(1,3)=-(1.D0+PQ)*Q1                                             DMP00920
      QQ(2,4)=-(1.D0+PQ)*Q2                                             DMP00930
      QQ(3,3)=QQ(1,1)                                                   DMP00940
      QQ(4,4)=QQ(2,2)                                                   DMP00950
      QQ(3,1)=QQ(1,3)                                                   DMP00960
      QQ(4,2)=QQ(2,4)                                                   DMP00970
      QQ(5,5)=RRI                                                       DMP00980
      QQ(6,6)=RRO                                                       DMP00990
      QQ(7,7)=RR1                                                       DMP01000
      CALL TPRINT(NOUT,NOUT,NOUT,QQ,'QQ EXTENDED COST MATRIX $',ITEST)  DMP01010
C                                                                       DMP01020
      CALL MQF(NOUT,NOUT,N,NOUT,N,QQ,C,Q,WORK)                          DMP01030
      CALL TPRINT(N,N,N,Q,'R1 = Q IN PROGRAM $',ITEST)                  DMP01040
C                                                                       DMP01050
C     R12=0                                                             DMP01060
C                                                                       DMP01070
      R(1,1)=R1                                                         DMP01080
C                                                                       DMP01090
90    CONTINUE                                                          DMP01100
```

```
      RETURN                                                            DMP01110
      END                                                               DMP01120
```



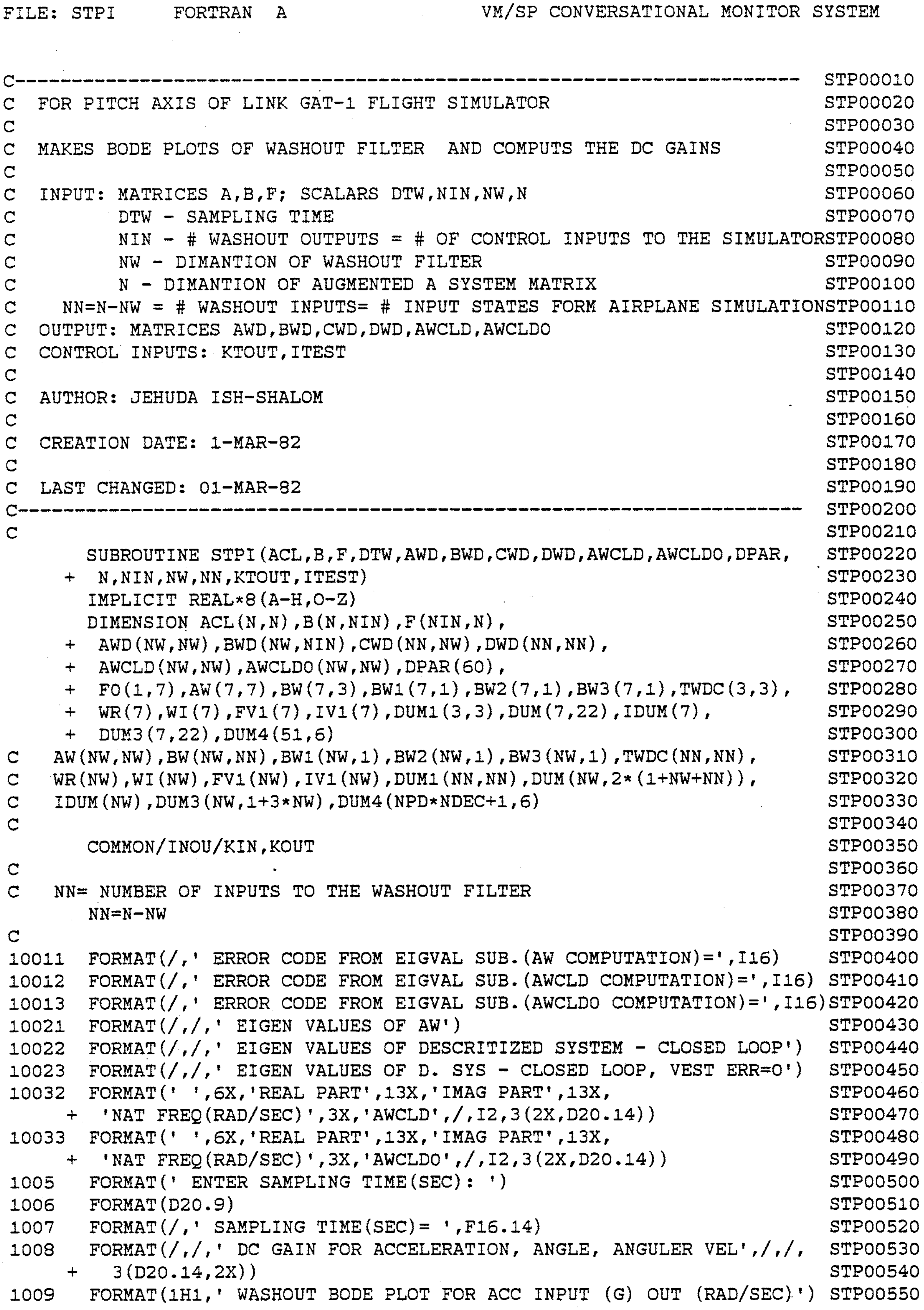
```
FILE: DESPI    FORTRAN  A                  VM/SP CONVERSATIONAL MONITOR SYSTEM


C---------------------------------------------------------------------------
C
C   SUBROUTINE THAT DESIGNS A TWO DEGREE OF FREEDOM WASHOUT FILTER
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CRIATION DATE: 1-MARCH-82
C
C---------------------------------------------------------------------------
      SUBROUTINE DESPI(A,B,Q,R,ACL,F,N,NIN,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(N,N),B(N,NIN),R(NIN,NIN),Q(N,N),
     + ACL(N,N),F(NIN,N),
     + RK(10,10),
     + DUM(20,54),IDUM(20,3),WORK(10)
C      AEQ(N,N),QEQ(N,N),RK(N,N),G(NIN,N),G1(NIN,N),RINV(NIN,NIN),
C      ST(NIN,N),DUM(2*N,4+5*N),IDUM(2*N,3),WORK(N)
C
      COMMON/INOU/KIN,KOUT
C    UNITS USED     METERS. RADENS,SECANDS;
C                    OTHER UNIT ARE NORMALIZED
C
1001  FORMAT(1H1)
1002  FORMAT(' WASHOUT FILTER DESIGN')
      N2=N*2
      WRITE(KOUT,1001)
      IF (ITEST.GT.0) WRITE(KOUT,1002)
      CALL TPRINT(N,N,N,A,'A IN DESIGN SUB. $',ITEST)
      CALL TPRINT(N,N,NIN,B,'B IN DESIGN SUB. $',ITEST)
      CALL TPRINT(N,N,N,Q,'Q=R1 IN DESIGN SUB. $',ITEST)
      CALL TPRINT(NIN,NIN,NIN,R,'R=R2 IN DESIGN SUB. $',ITEST)
      CALL REG(N,NIN,N2,A,B,R,Q,RK,F,ACL,DUM,IDUM)
C
      IF (ITEST.GT.0) WRITE(KOUT,1003)
1003  FORMAT(/,/,/,' FEEDBACK MATRIX F')
      IF (ITEST.GT.0) CALL MATIO(NIN,NIN,N,F,3)
C
90    CONTINUE
      RETURN
      END
```

```
C-----------------------------------------------------------------------RPI00010
C                                                                       RPI00020
C   SUBROUTINE RPI USED TO PRINT THE 10 FEEDFORWARD AND FEEDBACK RESISTORRPI00030
C   FOR THE LINK GAT-1 PITCH AXIS WASHOUT                               RPI00040
C                                                                       RPI00050
C   AUTHER: JEHUDA ISH-SHALOM                                           RPI00060
C                                                                       RPI00070
C   CRIATION DATE: 1-MAR-82                                             RPI00080
C    LAST CHANGE: 12-MAR-82                                             RPI00090
C                                                                       RPI00100
C-----------------------------------------------------------------------RPI00110
C                                                                       RPI00120
      SUBROUTINE RPI(F,KOUT)                                            RPI00130
      IMPLICIT REAL*8(A-H,O-Z)                                          RPI00140
      DIMENSION F(10),SCALE(10),R(10),K(10)                             RPI00150
      DATA SCALE(1)/0.1D0/,                                             RPI00160
     +     SCALE(2)/0.0551D0/,                                          RPI00170
     +     SCALE(3)/0.0268D0/,                                          RPI00180
     +     SCALE(4)/0.015D0/,                                           RPI00190
     +     SCALE(5)/0.3300D0/,                                          RPI00200
     +     SCALE(6)/0.033D0/,                                           RPI00210
     +     SCALE(7)/0.015D0/,                                           RPI00220
     +     SCALE(8)/0.1D0/,                                             RPI00230
     +     SCALE(9)/0.066D0/,                                           RPI00240
     +     SCALE(10)/0.0524D0/,                                         RPI00250
     +     K(1)/9/,                                                     RPI00260
     +     K(2)/10/,                                                    RPI00270
     +     K(3)/8/,                                                     RPI00280
     +     K(4)/1/,                                                     RPI00290
     +     K(5)/2/,                                                     RPI00300
     +     K(6)/6/,                                                     RPI00310
     +     K(7)/7/,                                                     RPI00320
     +     K(8)/3/,                                                     RPI00330
     +     K(9)/4/,                                                     RPI00340
     +     K(10)/5/                                                     RPI00350
1001  FORMAT(/,5X,' FEEDFORWARD RESISTORS')                             RPI00360
1002  FORMAT(/,5X,' FEEDBACK RESISTORS')                                RPI00370
1003  FORMAT(/,' R',I2,'=',F12.3,' KOHM')                               RPI00380
1004  FORMAT(/,' ')                                                     RPI00390
C                                                                       RPI00400
      DO 20 I=1,10                                                      RPI00410
20    R(I)=100.D0/SCALE(I)/F(I)                                         RPI00420
      DO 25 I=1,5                                                       RPI00430
25    R(K(I))=-2.0D0*R(K(I))                                            RPI00440
C 2.*R FOR I 1 TO 5 DONE TO ACOMODATE INCRASE GAIN OF FEEDFORWARD BY 2 RPI00450
C  HARDWARE CHANGED ON 7-MARCH-82                                       RPI00460
      WRITE(KOUT,1004)                                                  RPI00470
      WRITE(KOUT,1001)                                                  RPI00480
      DO 30 I=1,5                                                       RPI00490
30    WRITE(KOUT,1003) I,R(K(I))                                        RPI00500
      WRITE(KOUT,1002)                                                  RPI00510
      DO 40 I=6,10                                                      RPI00520
40    WRITE(KOUT,1003) I,R(K(I))                                        RPI00530
      WRITE(KOUT,1004)                                                  RPI00540
      RETURN                                                            RPI00550
```

```
FILE: RPI      FORTRAN  A                VM/SP CONVERSATIONAL MONITOR SYSTEM

      END                                                               RPI00560
```

```
FILE: PFPI     FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM
```

```
C-----------------------------------------------------------------------
C
C   SUBROUTINE PFP USED TO PRINT THE FEEDBACK GAINS FA, FS, FM, FN
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CRIATION DATE: 1-MAR-82
C   LAST CHANGED: 02-MAR-82
C
C-----------------------------------------------------------------------
C
      SUBROUTINE PFPI(F,N,NIN,KOUT)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION F(NIN,N),LABLE(5),NBA(6)
      DATA LABLE(1)/'FA  '/,LABLE(2)/'FS  '/,LABLE(3)/'FM  '/,
     +                     LABLE(4)/'FN L'/,LABLE(5)/'FN R'/
1001  FORMAT(1H1,10X,' FEEDBACK GAINS')
1002  FORMAT(/,5X,A4)
1003  FORMAT(/,' ',3(D20.13,3X))
C
      NBLOCK=5
      NV=2
      NMR=3
      NNL=1
      NNR=2
      NBA(1)=0
      NBA(2)=NV
      NBA(3)=NBA(2)+NV
      NBA(4)=NBA(3)+NMR
      NBA(5)=NBA(4)+NNL
      NBA(6)=NBA(5)+NNR
      WRITE(KOUT,1001)
      DO 20 K=1,NBLOCK
      WRITE(KOUT,1002) LABLE(K)
      DO 20 I=1,NIN
      J0=NBA(K)+1
      JE=NBA(K+1)
20    WRITE(KOUT ,1003) (F(I,J),J=J0,JE)
      RETURN
      END
```

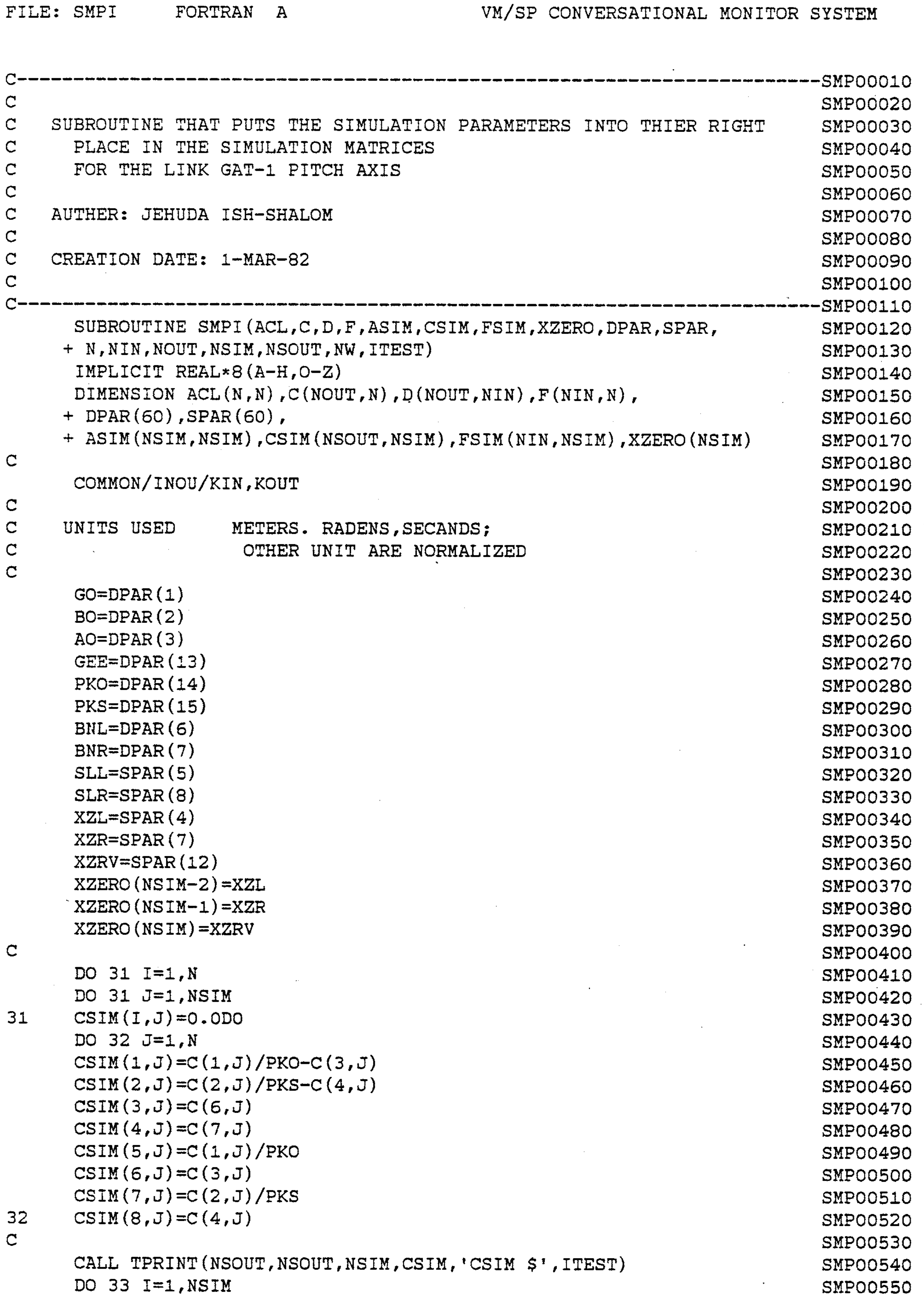


```
FILE: STPI     FORTRAN  A                   VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------
C  FOR PITCH AXIS OF LINK GAT-1 FLIGHT SIMULATOR
C
C  MAKES BODE PLOTS OF WASHOUT FILTER  AND COMPUTS THE DC GAINS
C
C  INPUT: MATRICES A,B,F; SCALARS DTW,NIN,NW,N
C         DTW - SAMPLING TIME
C         NIN - # WASHOUT OUTPUTS = # OF CONTROL INPUTS TO THE SIMULATOR
C         NW - DIMANTION OF WASHOUT FILTER
C         N - DIMANTION OF AUGMENTED A SYSTEM MATRIX
C    NN=N-NW = # WASHOUT INPUTS= # INPUT STATES FORM AIRPLANE SIMULATION
C  OUTPUT: MATRICES AWD,BWD,CWD,DWD,AWCLD,AWCLDO
C  CONTROL INPUTS: KTOUT,ITEST
C
C  AUTHOR: JEHUDA ISH-SHALOM
C
C  CREATION DATE: 1-MAR-82
C
C  LAST CHANGED: 01-MAR-82
C-----------------------------------------------------------------------
C
      SUBROUTINE STPI(ACL,B,F,DTW,AWD,BWD,CWD,DWD,AWCLD,AWCLDO,DPAR,
     + N,NIN,NW,NN,KTOUT,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ACL(N,N),B(N,NIN),F(NIN,N),
     + AWD(NW,NW),BWD(NW,NIN),CWD(NN,NW),DWD(NN,NN),
     + AWCLD(NW,NW),AWCLDO(NW,NW),DPAR(60),
     + FO(1,7),AW(7,7),BW(7,3),BW1(7,1),BW2(7,1),BW3(7,1),TWDC(3,3),
     + WR(7),WI(7),FV1(7),IV1(7),DUM1(3,3),DUM(7,22),IDUM(7),
     + DUM3(7,22),DUM4(51,6)
C    AW(NW,NW),BW(NW,NN),BW1(NW,1),BW2(NW,1),BW3(NW,1),TWDC(NN,NN),
C    WR(NW),WI(NW),FV1(NW),IV1(NW),DUM1(NN,NN),DUM(NW,2*(1+NW+NN)),
C    IDUM(NW),DUM3(NW,1+3*NW),DUM4(NPD*NDEC+1,6)
C
      COMMON/INOU/KIN,KOUT
C
C    NN= NUMBER OF INPUTS TO THE WASHOUT FILTER
      NN=N-NW
C
10011 FORMAT(/,' ERROR CODE FROM EIGVAL SUB.(AW COMPUTATION)=',I16)
10012 FORMAT(/,' ERROR CODE FROM EIGVAL SUB.(AWCLD COMPUTATION)=',I16)
10013 FORMAT(/,' ERROR CODE FROM EIGVAL SUB.(AWCLDO COMPUTATION)=',I16)
10021 FORMAT(/,/,' EIGEN VALUES OF AW')
10022 FORMAT(/,/,' EIGEN VALUES OF DESCRITIZED SYSTEM - CLOSED LOOP')
10023 FORMAT(/,/,' EIGEN VALUES OF D. SYS - CLOSED LOOP, VEST ERR=0')
10032 FORMAT(' ',6X,'REAL PART',13X,'IMAG PART',13X,
     + 'NAT FREQ(RAD/SEC)',3X,'AWCLD',/,I2,3(2X,D20.14))
10033 FORMAT(' ',6X,'REAL PART',13X,'IMAG PART',13X,
     + 'NAT FREQ(RAD/SEC)',3X,'AWCLDO',/,I2,3(2X,D20.14))
1005  FORMAT(' ENTER SAMPLING TIME(SEC): ')
1006  FORMAT(D20.9)
1007  FORMAT(/,' SAMPLING TIME(SEC)= ',F16.14)
1008  FORMAT(/,/,' DC GAIN FOR ACCELERATION, ANGLE, ANGULER VEL',/,/,
     +  3(D20.14,2X))
1009  FORMAT(1H1,' WASHOUT BODE PLOT FOR ACC INPUT (G) OUT (RAD/SEC)')
```

```
1010  FORMAT(1H1,' WASHOUT BODE PLOT FOR D_TETA INPUT,UNITS (RAD/SEC)')
C
      GO=DPAR(1)
      GS=DPAR(4)
      GEE=DPAR(13)
C
      DO 20 I=1,NW
      DO 20 J=1,NW
20    AW(I,J)=ACL(I,J)
      AW(5,5)=0.D0
      AW(6,6)=0.D0
      DO 21 I=1,NW
      DO 21 J=1,NN
21    BW(I,J)=ACL(I,J+NW)
      DO 22 J=1,NW
      CWD(1,J)=-F(1,J)
      DO 22 I=2,NN
22    CWD(I,J)=0.D0
      CWD(2,6)=1.D0
      CWD(3,7)=1.D0
      DO 23 I=1,NIN
      DO 23 J=1,NN
23    DWD(I,J)=-F(I,J+NW)
      DO 24 I=2,3
      DO 24 J=1,NN
24    DWD(I,J)=0.D0
C
      CALL TPRINT(NW,NW,NW,AW,'AW TOP LEFT OF NWXNW OF ACL $',ITEST)
      CALL TPRINT(NW,NW,NN,BW,'BW TOP NW ROWS NW+1 TO N COL OF ACL $',
     +  ITEST)
      CALL TPRINT(NN,NN,NW,CWD,'CW=CWD OF WASHOUT FILTER $',ITEST)
      CALL TPRINT(NN,NN,NN,DWD,'DW=DWD OF WASHOUT FILTER $',ITEST)
C
C  CALL TO MSCALE ADDED TO FIXE UP ERROR IN DCGAIN SUB.
C   COMPUTES D+C.A**-1 .B INSTEAD OF D-C.A**-1 .B
C
      CALL MSCALE(NW,NW,NN,-1.0D0,BW)
      CALL DCGAIN(NW,NN,AW,BW,CWD,DWD,TWDC,DUM1,IDUM)
      CALL MSCALE(NW,NW,NN,-1.0D0,BW)
C  RESATING THE VALUE OF MATRIX BW
C
      CALL TPRINT(NN,NN,NN,TWDC,'DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:CO
     +M,ANGLE,D_ANGLE $',1)
      CALL TPRINT(NN,NN,NN,TWDC,'DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:CO
     +M,ANGLE,D_ANGLE $',2)
C
      DO 25 I=1,NW
      BW1(I,1)=BW(I,1)
      BW2(I,1)=BW(I,2)
25    BW3(I,1)=BW(I,3)
C
C     CALL DCGAIN(NW,NIN,AW,BW1,CWD,DWD(1,1),TWDC(1,1),DUM1,IDUM)
C     CALL DCGAIN(NW,NIN,AW,BW2,CWD,DWD(1,2),TWDC(1,2),DUM1,IDUM)
C     CALL DCGAIN(NW,NIN,AW,BW3,CWD,DWD(1,3),TWDC(1,3),DUM1,IDUM)
C     WRITE(KOUT,1008)  TWDC
```

```
FILE: STPI     FORTRAN  A                 VM/SP CONVERSATIONAL MONITOR SYSTEM


C     WRITE(KTOUT,1008) TWDC
C
C     IP=1000
C     NPD=10
C     NDEC=5
C     WZERO=0.01
C     NPTS=51
C     WRITE(KOUT,1009)
C     CALL BNIN(NW,NPTS,AW,BW1,CWD,DWD(1,1),WZERO,NDEC,NPD,IP,DUM3,
C    +  IDUM,DUM4)
C     WRITE(KOUT,1010)
C     CALL BNIN(NW,NPTS,AW,BW3,CWD,DWD(1,3),WZERO,NDEC,NPD,IP,DUM3,
C    +  IDUM,DUM4)
C30   WRITE(KTOUT,1005)
C     READ(KIN,1006,ERR=30) DTW
C     IF (DTW.EQ.0.D0) GOTO 90
C     WRITE(KOUT,1007) DTW
C     CALL DLIN(NW,NIN,AW,BW,AWD,BWD,DTW,DUM,IDUM)
C     CALL TPRINT(NW,NW,NW,AWD,'AWD $',ITEST)
C     CALL TPRINT(NW,NW,NIN,BWD,'BWD $',ITEST)
C     CALL MMUL(NW,NIN,NW,NW,NW,NIN,BWD,CWD,AWCLD)
C     CALL TPRINT(NW,NW,NW,AWCLD,'-BWD.F $',ITEST)
C     CALL MMUL(NW,NIN,NW,NW,NW,NIN,BWD,F0,AWCLD0)
C     CALL TPRINT(NW,NW,NW,AWCLD0,'-BWD.F0 FOR 0 VESTIBULAR FEEDBACK $'
C    +  ,ITEST)
C     DO 31 I=1,NW
C     DO 31 J=1,NW
C     AWCLD(I,J)=AWD(I,J)+AWCLD(I,J)
C31   AWCLD0(I,J)=AWD(I,J)+AWCLD0(I,J)
C     CALL TPRINT(NW,NW,NW,AWCLD,'AWCLD $',ITEST)
C     CALL TPRINT(NW,NW,NW,AWCLD0,'AWCLD0 $',ITEST)
C
C     IF (ITEST.EQ.0) GOTO 50
C     WRITE(KOUT,10021)
C  DUM DISTROYED BY EIGVAL
C     CALL SAVE(NW,NW,NW,NW,AW,DUM)
C     CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)
C     IF (IERR.NE.0) WRITE(KOUT,10011) IERR
C     CALL SAVE(NW,NW,NW,NW,AWD,DUM)
C     CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)
C
C50   WRITE(KOUT,10022)
C     CALL SAVE(NW,NW,NW,NW,AWCLD,DUM)
C     CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)
C     IF (IERR.EQ.0) GOTO 51
C     WRITE(KTOUT,10012) IERR
C     WRITE(KOUT,10012) IERR
C51   DO 52 I=1,NW
C     FREQN=DSQRT(WR(I)**2+WI(I)**2)
C52   IF (FREQN.GT.1.D0) WRITE(KTOUT,10032) I,WR(I),WI(I),FREQN
C
C     WRITE(KOUT,10023)
C     CALL SAVE(NW,NW,NW,NW,AWCLD0,DUM)
C     CALL EIGVAL(NW,NW,DUM,WR,WI,IV1,FV1,IERR)
C     IF (IERR.EQ.0) GOTO 61
```

```
C       WRITE(KTOUT,10013) IERR                                         STP01660
C       WRITE(KOUT,10013) IERR                                          STP01670
C61     DO 62 I=1,NW                                                    STP01680
C       FREQN=DSQRT(WR(I)**2+WI(I)**2)                                  STP01690
C62     IF (FREQN.GT.1.D0) WRITE(KTOUT,10033) I,WR(I),WI(I),FREQN       STP01700
C                                                                       STP01710
C       GOTO 30                                                         STP01720
90      CONTINUE                                                        STP01730
        RETURN                                                          STP01740
        END                                                             STP01750
```



```
FILE: SMPI     FORTRAN  A               VM/SP CONVERSATIONAL MONITOR SYSTEM


C-----------------------------------------------------------------------
C
C   SUBROUTINE THAT PUTS THE SIMULATION PARAMETERS INTO THIER RIGHT
C     PLACE IN THE SIMULATION MATRICES
C     FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 1-MAR-82
C
C-----------------------------------------------------------------------
      SUBROUTINE SMPI(ACL,C,D,F,ASIM,CSIM,FSIM,XZERO,DPAR,SPAR,
     + N,NIN,NOUT,NSIM,NSOUT,NW,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ACL(N,N),C(NOUT,N),D(NOUT,NIN),F(NIN,N),
     + DPAR(60),SPAR(60),
     + ASIM(NSIM,NSIM),CSIM(NSOUT,NSIM),FSIM(NIN,NSIM),XZERO(NSIM)
C
      COMMON/INOU/KIN,KOUT
C
C    UNITS USED     METERS. RADENS,SECANDS;
C                    OTHER UNIT ARE NORMALIZED
C
      GO=DPAR(1)
      BO=DPAR(2)
      AO=DPAR(3)
      GEE=DPAR(13)
      PKO=DPAR(14)
      PKS=DPAR(15)
      BNL=DPAR(6)
      BNR=DPAR(7)
      SLL=SPAR(5)
      SLR=SPAR(8)
      XZL=SPAR(4)
      XZR=SPAR(7)
      XZRV=SPAR(12)
      XZERO(NSIM-2)=XZL
      XZERO(NSIM-1)=XZR
      XZERO(NSIM)=XZRV
C
      DO 31 I=1,N
      DO 31 J=1,NSIM
31    CSIM(I,J)=0.0D0
      DO 32 J=1,N
      CSIM(1,J)=C(1,J)/PKO-C(3,J)
      CSIM(2,J)=C(2,J)/PKS-C(4,J)
      CSIM(3,J)=C(6,J)
      CSIM(4,J)=C(7,J)
      CSIM(5,J)=C(1,J)/PKO
      CSIM(6,J)=C(3,J)
      CSIM(7,J)=C(2,J)/PKS
32    CSIM(8,J)=C(4,J)
C
      CALL TPRINT(NSOUT,NSOUT,NSIM,CSIM,'CSIM $',ITEST)
      DO 33 I=1,NSIM
      DO 33 J=1,NSIM
33    ASIM(I,J)=0.D0
      DO 34 I=1,NW
      DO 34 J=1,N
34    ASIM(I,J)=ACL(I,J)
      ASIM(NW+1,N)=SLL
      ASIM(NW+2,N)=1.D0
      ASIM(5,5)=0.D0
      ASIM(6,6)=0.D0
      CALL TPRINT(NSIM,NSIM,NSIM,ASIM,'ASIM $',ITEST)
      DO 35 I=1,NIN
      DO 35 J=1,N
35    FSIM(I,J)=F(I,J)
      CALL TPRINT(NIN,NIN,NSIM,FSIM,'FSIM $',ITEST)
90    CONTINUE
      RETURN
      END
```

```
FILE: SIMPI    FORTRAN  A                 VM/SP CONVERSATIONAL MONITOR SYSTEM

C-----------------------------------------------------------------------
C
C   SUBROUTINE THAT SIMULATS THE DESIGNED WASHOUT FILETER
C   FOR THE LINK GAT-1 PITCH AXIS
C
C   AUTHER: JEHUDA ISH-SHALOM
C
C   CREATION DATE: 1-MAR-82
C
C   INPUT:
C     SYSTEM MATRICES:    ASIM(NSIM,CSIM(NSOUT,NSIM),FSIM(NIN,NSIM)
C     INITIAL CONDITIONS: XZERO(NSIM)
C     SYSTEM DIMENTIONS:  NSIM,NIN,NSOUT
C     OTHER SIMULATION PARAMETERS: SPAR(60)
C     TEST FLAG: ITEST (PRINT OUT TEST PRINTOUT IF ITEST>0)
C
C   OUTPUT:
C     PRINTPLOTS OF OUTPUTS AND CONTROLS(COMPUTED FROM FSIM)
C
C   SUBROUTINES CALLED:
C     MATIO,REGSIM
C
C-----------------------------------------------------------------------
      SUBROUTINE SIMPI(ASIM,CSIM,FSIM,XZERO,SPAR,NSIM,NIN,NSOUT,ITEST)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ASIM(NSIM,NSIM),CSIM(NSOUT,NSIM),FSIM(NIN,NSIM),
     + XZERO(NSIM),SPAR(60),
     + DUM(10,51),IDUM(10),DUM2(51,9)
C  DUM(NSIM,(1+5*NSIM)),IDUM(NSIM),DUM2(NPTS,NIN+NSOUT)
C
      COMMON/INOU/KIN,KOUT
C
C  UNITS USED    METERS, RADENS, SECONDS,
C                OTHER NORMALIZED UNITS
C
1001  FORMAT(1H1)
1002  FORMAT(/,/,' ORDER OF OUTPUTS Y(I)',/,
     + ' Y(1)=EOTO,    Y(2)=ESCC,    Y(3)=TETAM,  Y(4)=D_TETAM',/,
     + ' Y(5)=YAOTO,   Y(6)=YSOTO,   Y(7)=YASCC,  Y(8)=YSSCC')
1003  FORMAT(/,' NPTS= ',I5,' NPRPL= ',I3,' DT= ',D20.15,' (SEC)')
C
      NPTS=IDINT(SPAR(1)+0.5D0)
      NPRPL=IDINT(SPAR(2)+0.5D0)
      DT=SPAR(3)
      IF (ITEST.GT.0) WRITE(KOUT,1003) NPTS,NPRPL,DT
      CALL TPRINT(NSIM,NSIM,NSIM,ASIM,'ASIM IN SIM SUB.$',ITEST)
      CALL TPRINT(NSOUT,NSOUT,NSIM,CSIM,'CSIM IN SIM SUB.$',ITEST)
      CALL TPRINT(NIN,NIN,NSIM,FSIM,'FSIM IN SIM SUB.$',ITEST)
      WRITE(KOUT,1002)
      CALL MATIO(NSIM,NSIM,1,XZERO,3)
C
      CALL REGSIM(NSIM,NSOUT,NIN,ASIM,CSIM,FSIM,XZERO,DT,NPTS,NPRPL,
     +   DUM,IDUM,DUM2)
C
      WRITE(KOUT,1001)
```

```
FILE: SIMPI    FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


      CALL TPRINT(NSIM,NSIM,1,XZERO,'XZERO AT END OF SIMULATION$',ITEST)SIM00560
      RETURN                                                            SIM00570
      END                                                               SIM00580
```

```
FILE: TPRINT   FORTRAN  A                    VM/SP CONVERSATIONAL MONITOR SYSTEM


       SUBROUTINE TPRINT(NN,N,M,MATRIX,TITLE,ITEST)                     TPR00010
C  SUBROTINE TO PRINT A MATRIX WHEN TEST NOT EQUAL 0                    TPR00020
C    TITLE LENTH LIMETED TO 60 CHARECTERS (INPUT AS A LITERAL CONSTANT) TPR00030
C                                                                       TPR00040
C   AUTHOR: JEHUDA ISH-SHALOM                                           TPR00050
C                                                                       TPR00060
C   CREATED: 31-OCT-80                                                  TPR00070
C                                                                       TPR00080
       INTEGER TITLE(15)                                                TPR00090
       REAL*8 MATRIX(NN,M)                                              TPR00100
       COMMON/INOU/KIN,KOUT                                             TPR00110
       KTOUT=7                                                          TPR00120
       KKOUT=KOUT                                                       TPR00130
       IF (ITEST.EQ.0) GOTO 90                                          TPR00140
       IF (ITEST.EQ.2) KOUT=KTOUT                                       TPR00150
       WRITE(KOUT,10) TITLE                                             TPR00160
10     FORMAT(/,' ',10X,15A4)                                           TPR00170
       CALL MATIO(NN,N,M,MATRIX,3)                                      TPR00180
90     KOUT=KKOUT                                                       TPR00190
       RETURN                                                           TPR00200
       END                                                              TPR00210
```

Appendix C: GAT-1 Optimal Washout System Operating Instructions

Figure 1 describes the pitch optimal washout system implemented in the GAT-1 flight simulator.

1. Inputs, U^a, from airplane dynamic simulation, $\dot{\theta}^{sc}$

$$\bar{U}^a = \begin{pmatrix} a_x^a \\ \dot{\theta}^a \\ \theta^a \end{pmatrix}$$

a_{xa}^a - longitudinal linear acceleration (in airplane body axes)

$\dot{\theta}^a$ - Euler pitch rate

θ^a - Euler pitch angle

$\dot{\theta}^{sc}$ - Extra input for further use for predictive wasgout

2. Feedback, X^m, motion base states:

(In NORMAL mode, feedback from actual motion base.
In TEST mode, feedback from simulated motion base.

$$X^m = \begin{pmatrix} \dot{\theta}^{s'} \\ \theta^{s'} \end{pmatrix}$$

$\dot{\theta}^{s'}$ - Euler pitch rate

$\theta^{s'}$ - Euler pitch angle

3. Output, $\dot{\theta}^{s1}$:

$\dot{\theta}^{s1}$ - simulator pitch axis command

4. Modes of Operation and Controls:

4.1 WASHOUT SELECT - is a thumb wheel located on the side of the washout board; it is a ten position select switch. Positions 0 through 7 select up to 8 different washout gain settings. Positions 8 and 9 correspond to washout gain settings 0 and 1 respectively where $\bar{U}^a = 0$, but $\dot{\theta}^{sc}$ can still be used as an active input (possibly for testing).

4.2 RESET - zeros all the internal states of the optimal washout controller (vestibular models) and simulated motion base (its $\dot{\theta}^{s'}$, $\theta^{s'}$). When power is switched on, the reset circuit initiates an automatic reset. This push button is located on the side of the washout board.

4.3 Pitch Motion ON/OFF - This corresponds to the original GAT-1 switch. It enables us to switch the motion base off at any time (a brake is applied to the pitch axis in the OFF position -- remember it is an inverted pendulum).

4.4 NORMAL/TEST - This switch is located on the side of the washout board. The NORMAL position is down when board is inserted in the GAT-1. This position should always be used when Pitch Motion is ON! The TEST position is used to replace the actual pitch axis motion base by a circuit model for use in testing. It is good practice to check the response of the circuit model first whenever using a new washout!

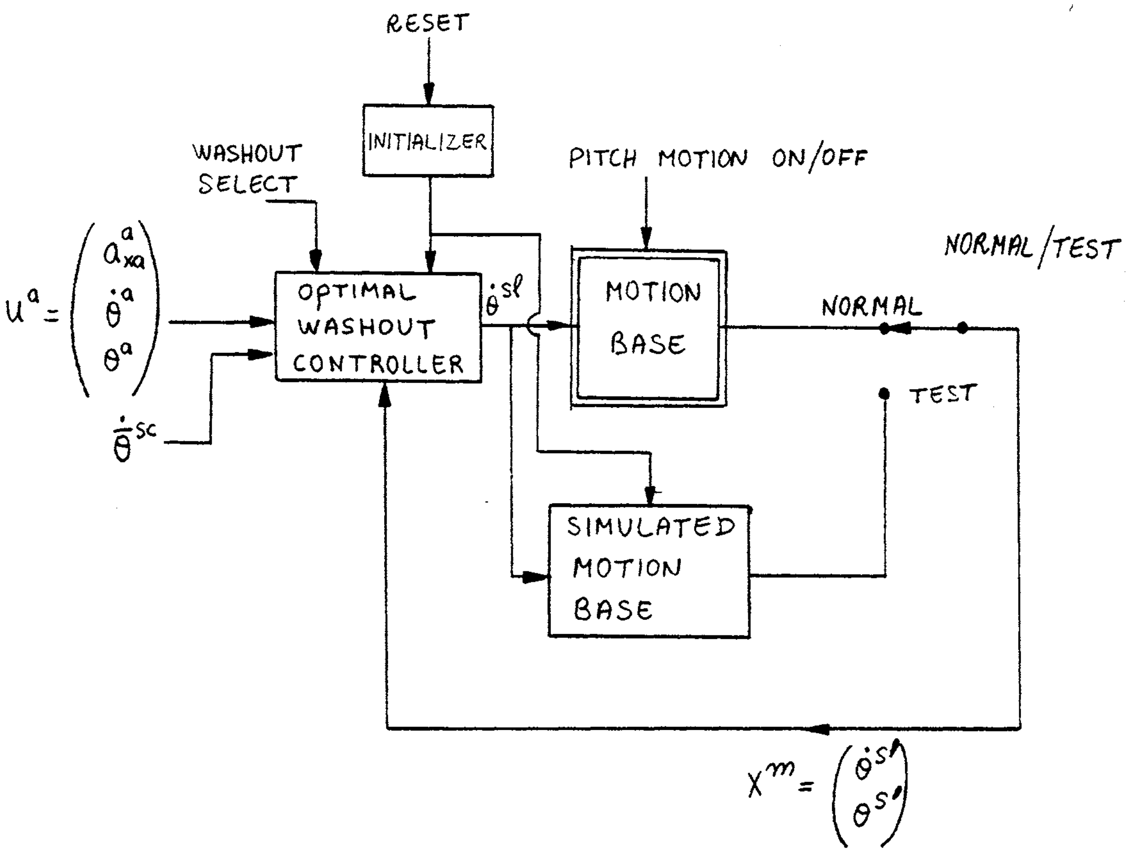
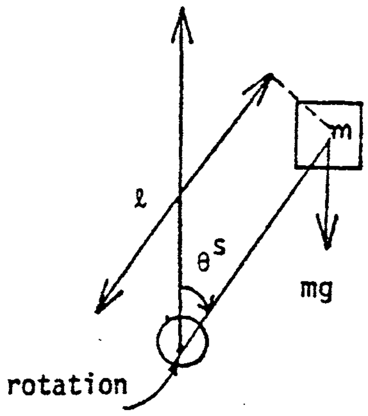

RESET
INITIALIZER
WASHOUT SELECT
PITCH MOTION ON/OFF
NORMAL/TEST
$u^a = \begin{pmatrix} a_{xa}^a \\ \dot{\theta}^a \\ \theta^a \end{pmatrix}$
OPTIMAL WASHOUT CONTROLLER
$\dot{\theta}^{sl}$
MOTION BASE
NORMAL
TEST
$\dot{\theta}^{sc}$
SIMULATED MOTION BASE
$X^m = \begin{pmatrix} \dot{\theta}^{s'} \\ \theta^{s'} \end{pmatrix}$

FIGURE 1 : PITCH WASHOUT SYSTEM

Appendix D: GAT-1 Pitch Motion-Base Model



$$\text{TORQUE} = I_{yy}\ddot{\theta}^S - \ell mg\theta^S + B\dot{\theta}^S$$

$$\therefore \quad \ddot{\theta}^S = (-(B+k_m)/I_{yy})\dot{\theta}^S + (\ell mg/I_{yy})\theta^S + (k_m/I_{yy})(\dot{\theta}^C/k_\Omega)$$

Parameters used for the pitch optimal washout design (the capital letter parameters are the symbols used in the computer program).

$$BR \triangleq \frac{B + k_m}{I_{yy}} = 3.13\ [1/\text{sec}]$$

$$BBR \triangleq \frac{k_m}{I_{yy}k_\Omega} = 0.167\ [\text{rad}/(\text{sec}^2\text{volt})]$$

$$PIP \triangleq \frac{\ell mg}{I_{yy}} = 0.74\ [1/\text{sec}^2]$$

Symbol key:

$\dot{\theta}^C$ = simulator control voltage (volts)
k_m = dc motor torque constant (Newton-m-sec)
k_Ω = dc motor back EMF constant (volts-sec/rad)
B = viscous forces (Newton-m-sec)
I_{yy} = inertia around pivot (kg-m^2)
mg = simulator weight (Newtons)
ℓ = length from pivot to center of gravity of m

Inverted pendulum model of the pitch axis of the GAT-1 flight simulator

Appendix E: Circuit Testing of the Pitch Optimal Washout

The pitch optimal washout has been tested with two sets of optimal gains generated by the washout design program, while using a model for the simulator motion base (Figure VII.3). A comparison of the computer simulated results and the circuit output for a test case of a 0.3 g step in linear acceleration, a^a_{xa} ($\theta^a(t) = 0$, $\dot{\theta}^a(t) = 0$) is given in Figures 1–6. Figures 1 and 2 show the simulator pitch angle and pitch rate response. Figures 3, 4 and 5 show the vestibular model responses of the otolith and semicircular canal for the airplane and simulator pilots. Note that the airplane pilot's semicircular canal output is zero—the airplane has zero pitch rate. The measured results from the simulator are very close to the computer simulations obtained from the washout system design programs.

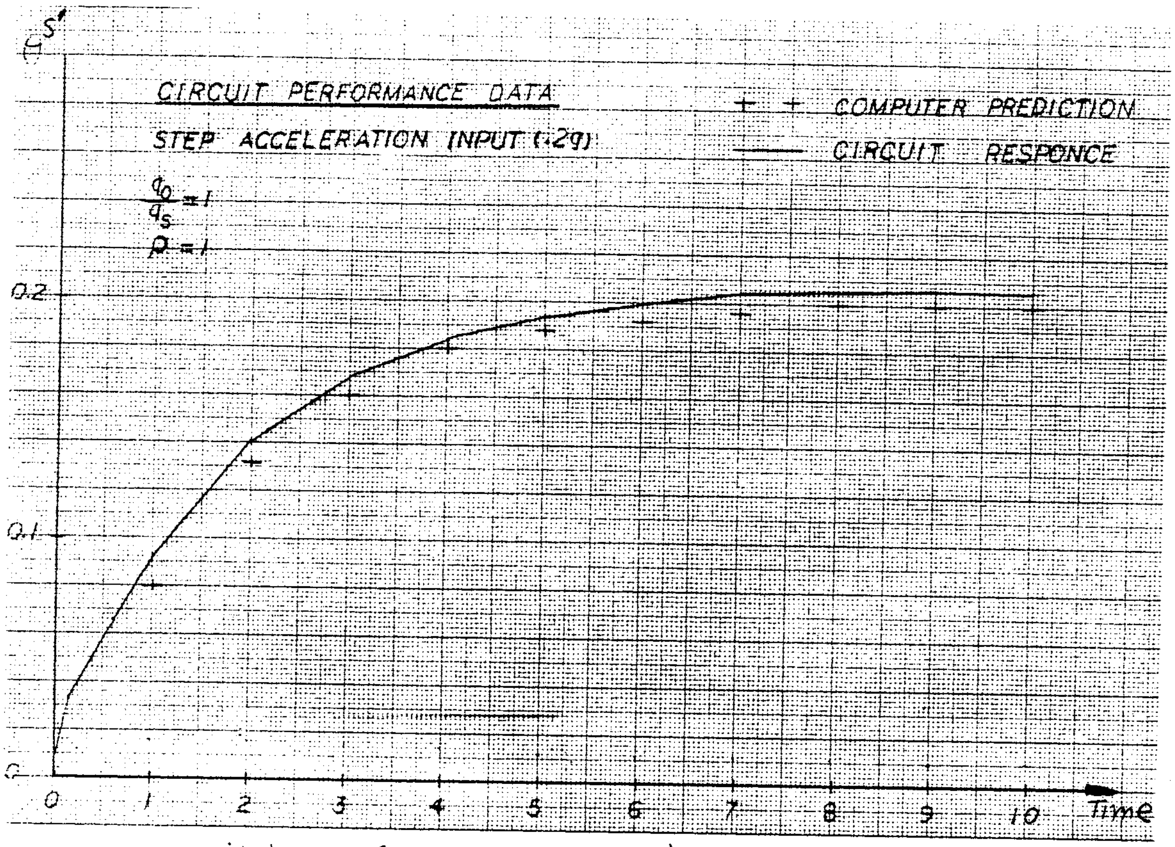


figure 1 : pitch angle of motion base

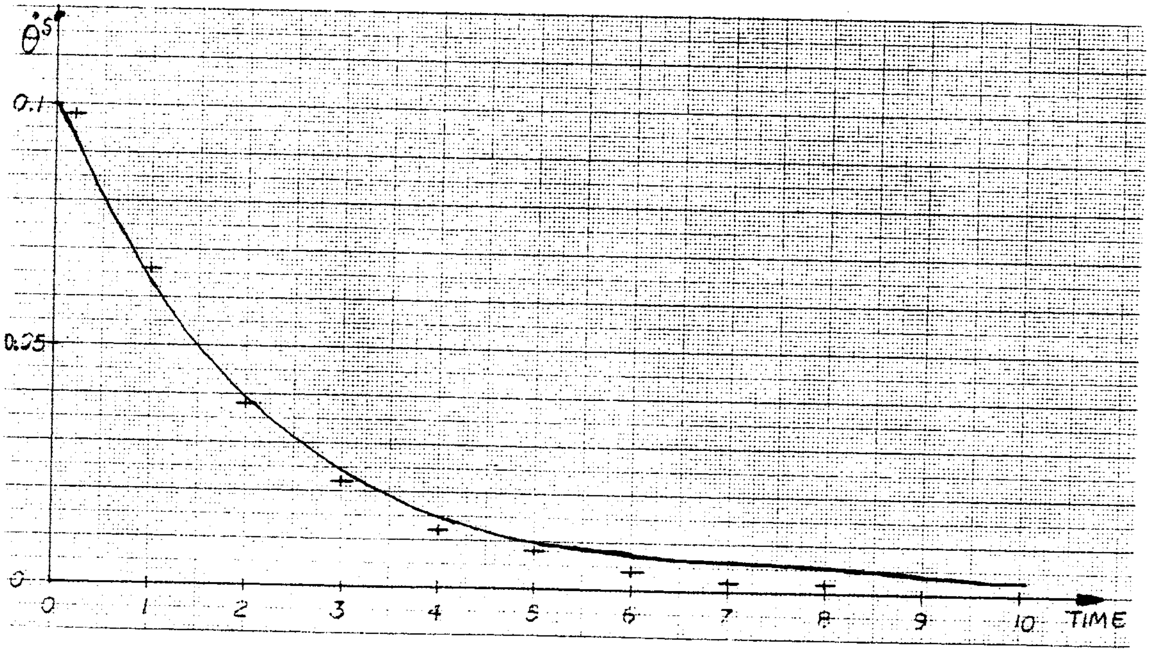


figure 2 : motion-base pitch rate

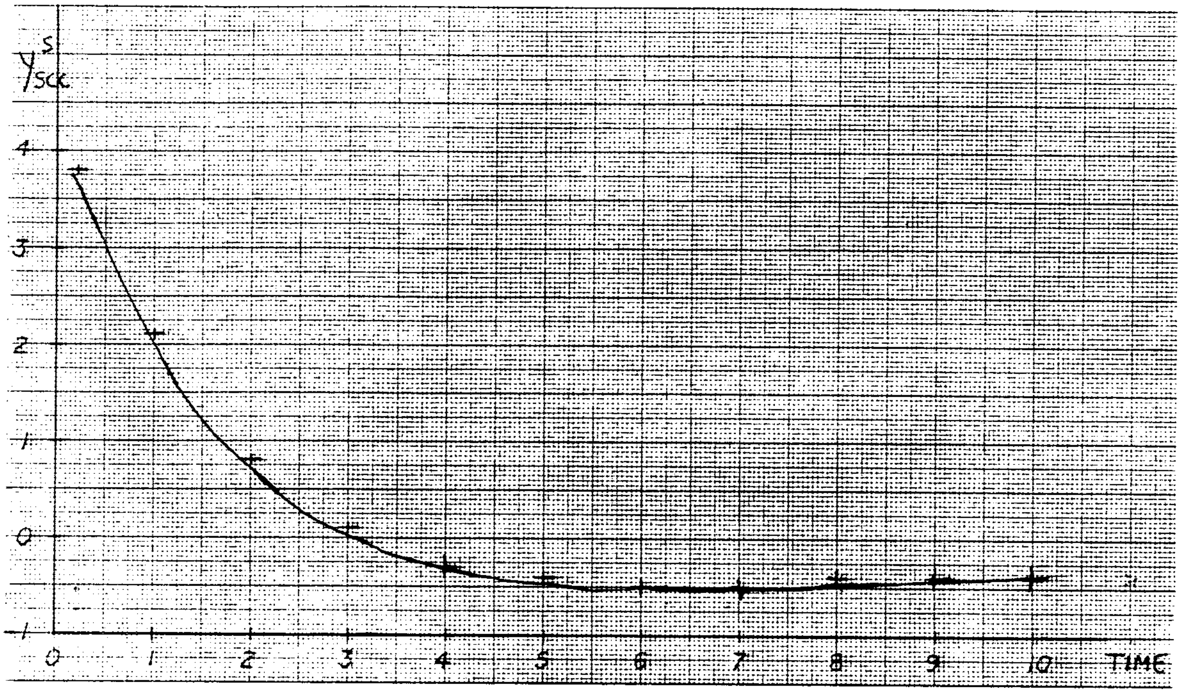


figure 3: semicircular canal vestibular model output of simulator pilot

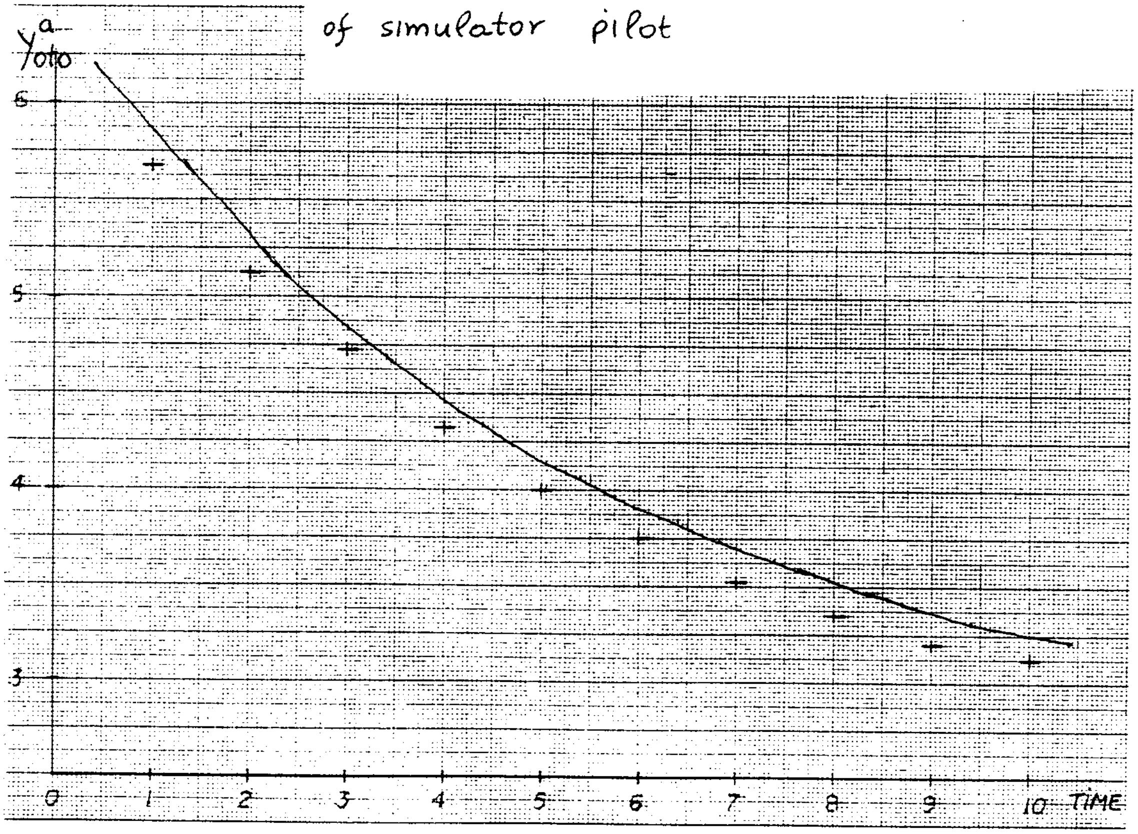


figure 4: otolith vestibular model output of reference airplane pilot

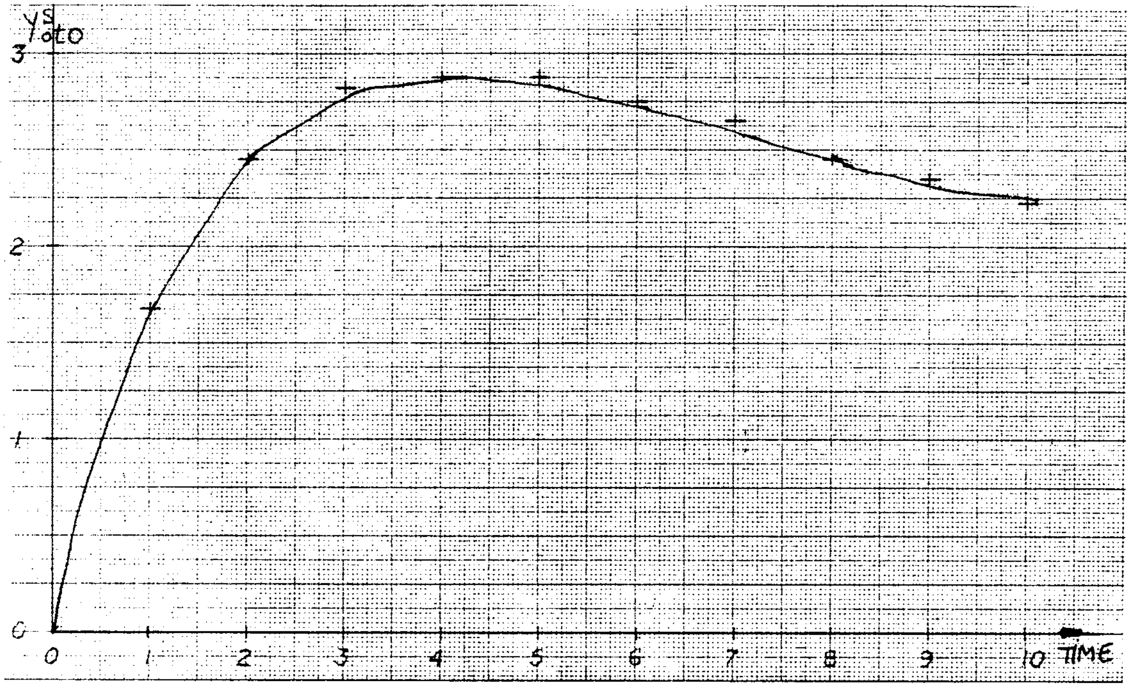
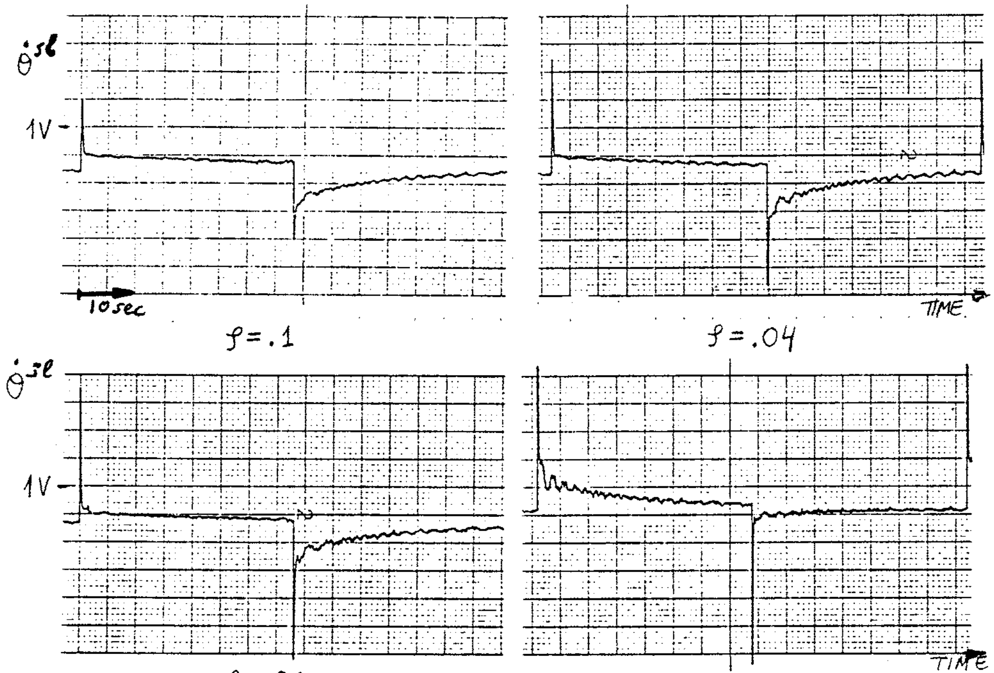


figure 5: otolith vestibular model output of simulator pilot



$\rho = .1$

$\rho = .04$

$\rho = .02$

$\rho = .01$

figure 6: comparison of the simulator command $\dot{\theta}^{sl}$ for 4 different values of ρ. $\rho = .04$ was used for all washouts.

Appendix F: GAT-1 Modifications for Use of the Pitch Washout

Changes made to the GAT-1 were designed in general to be reversible, i.e. one can plug in the original ATTITUDE card (assembly 633745 slot J22) and the GAT-1 would operate as it was originally designed. There are, however, minor circuit changes still in place. When replacing other cards, note changes to TIME DIVISION (assembly 633713, slot J21), and RELATIVE WIND (assembly 633743, slot J20).

Changes to the Link GAT-1 backplane (for use of the optimal washout system)

1.

Current	Changed to	Signal
J21-36 to J22-21 B$\boxed{9}$ B$\boxed{15}$ 633713 633745	J21-36 to J22-18 B$\boxed{9}$ B$\boxed{15}$ 633713 633745	sinθ from potentiometer on motion base from 39-J

Other changes, relevant to the above:

Attitude card (633745 B$\boxed{15}$) for normal link motion needs the addition jumper on the card: J22-21 to J22-18.

Purpose of the above change was to allow the change from the Link washout to the new washout by merely replacing the Link attitude card 633745 by a modified 633745 card. No other changes are necessary.

2. Add jumper J22-V to J

Signal: Command from new washout to pitch power amplifier.

3. Add diode 1N4148 in parallel to C_1 in assembly 633901E (diagram B$\boxed{11}$).

Diode minus (cathode) connected to ground.

Connect switch to choice between external or normal (pilot's) flaps control.

Purpose: To protect capacitor from external input with the wrong polarity.

To enable external flaps control for testing the pilot's ability to detect flaps change as part of the washout validation.

Appendix G: Design Parameters and Simulations of Washouts #0 and #2

In this appendix, we include designs and simulations of two optimal washout systms for the GAT-1 flight simulator pitch axis. The two washout systems are referenced as #0 and #2 according to the slector switch setting on the GAT-1. In washout #0, the design uses equal weights in the cost, J, for one threshold unit error of the semicircular canals and the otoliths. In washout #2, the canal error is weighted ten times more than the otolith error (both in threshold units), which is the same as saying that the semicircular canal sensitivity is increased by a factor of 3.16 (= $\sqrt{10}$, have a threshold of 1/3.16). The design includes the feedback gains, the open and closed-loop poles, and the steady state (dc gain) of the washout system. The simulations are for an input step of 0.2 g acceleration.

The plots include:

e_{oto}	otolith error (threshold units)
e_{scc}	semicircular canal error (threshold units)
θ^s	motion base pitch angle (rad)
$\dot{\theta}^s$	motion base pitch rate (rad/sec)
y^a_{oto}	otolith vestibular model output of reference airplane pilot (threshold units)
y^a_{scc}	semicircular canal model output of reference airplane pilot (threshold units)
y^s_{oto}	otolith vestibular model output of simulator pilot (threshold units)
y^s_{scc}	semicircular canal model output of simulator pilot (threshold units)
$\dot{\theta}^c$	pitch rate motion base command (volts)

The next two pages of this appendix contain a summary of washout #0 and #2 parameters and their correspondingly designed resistors R1 - R9.

WASHOUT #0

```
WAHOUT FILTER DESIGN PARAMETERS (11/16/81 18*09*17.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
*  20 RHO=    0.10D+00 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=       32.640 KOHM

R2=       18.126 KOHM

R3=       25.676 KOHM

R4=       24.661 KOHM

R5=       98.010 KOHM

     FEEDBACK RESISTORS

R6=       58.604 KOHM

R7=       72.525 KOHM

R8=       92.019 KOHM

R9=      360.022 KOHM
```

WASHOUT # 2

```
WAHOUT FILTER DESIGN PARAMETERS (11/16/81 18*17*03.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.10D+00 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
*  20 RHO=    0.10D+00 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=      115.125 KOHM

R2=       16.715 KOHM

R3=      115.594 KOHM

R4=      105.592 KOHM

R5=       40.807 KOHM

     FEEDBACK RESISTORS

R6=      154.204 KOHM

R7=       60.946 KOHM

R8=      394.002 KOHM

R9=      149.899 KOHM
```

WASHOUT #0

```
 GAT-1 LINK PITCH AXIS WASHOUT DESIGN PROGRAM

    WRITTEN BY JEHUDA ISH-SHALOM (OCTOBER 1981)


MOTION BASE COMMAND INPUT: ANGULER VELOCITY
MEASURED STATES: ANGULER POSTION AND VELOCITY, AND LINEAR ACCELERATION IN MOTION BASE INIRIAL AXE
VESTIBULAR MODEL INPUTS:
          LINEAR  = OTOLITH : LINEAR ACCELERATION
          ANGULER = SEMICIRCULAR CANAL: ANGULER VELOCITY


WAHOUT FILTER DESIGN PARAMETERS ( 1/18/82 11*54*06.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
*  20 RHO=     0.10D+00 WEIGHT OF MOTION OVER ERROR


      EXTENDED Q MATRIX



         COL    1           2           3           4           5           6
 ROW

   1       7.0711D-01  0.0         -7.0711D-01  0.0         0.0         0.0

   2       0.0         7.0711D-01  0.0         -7.0711D-01  0.0         0.0

   3      -7.0711D-01  0.0         7.0711D-01   0.0         0.0         0.0

   4       0.0        -7.0711D-01  0.0          7.0711D-01  0.0         0.0

   5       0.0         0.0         0.0          0.0         0.0         0.0

   6       0.0         0.0         0.0          0.0         0.0         0.0
```

```
OPEN LOOP EIGENVALUES


                                    ---EIGENVALUES---
              REAL PART        IMAG PART       NAT FREQ(HZ)        ZETA          FREQ(HZ)

        1     -3.351D+00        0.0            5.333D-01         1.000000        0.0
        2      2.208D-01        0.0            3.515D-02        -1.000000        0.0
        3     -1.900D-01        0.0            3.024D-02         1.000000        0.0
        4     -1.900D-01        0.0            3.024D-02         1.000000        0.0
        5     -1.695D-01        0.0            2.698D-02         1.000000        0.0
        6     -1.695D-01        0.0            2.698D-02         1.000000        0.0
        7     -1.000D-01        0.0            1.592D-02         1.000000        0.0
        8     -1.000D-01        0.0            1.592D-02         1.000000        0.0
        9     -1.000D-05        0.0            1.592D-06         1.000000        0.0
```

note unstable plant

```
                                 CLOSED-LOOP EIGENVALUES

              REAL PART        IMAG PART       NAT FREQ(HZ)        ZETA          FREQ(HZ)

        1     -1.807D+01        0.0            2.876D+00         1.000000        0.0
        2     -5.249D-01        0.0            8.355D-02         1.000000        0.0
        3     ~~-1.900D-01~~    0.0            3.024D-02         1.000000        0.0
        4     ~~-1.695D-01~~    0.0            2.698D-02         1.000000        0.0
        5     -1.664D-01        0.0            2.648D-02         1.000000        0.0
        6     ~~-1.000D-01~~    0.0            1.592D-02         1.000000        0.0
        7     ~~-1.000D-01~~    0.0            1.592D-02         1.000000        0.0
        8     -7.820D-02        0.0            1.245D-02         1.000000        0.0
        9     -1.~~000D-05~~    0.0            1.592D-06         1.000000        0.0
```

due to motion base dynamics

"washout pole"

main show approximate pole-zero cancilations

```
                 FEEDBACK GAINS

      FA

  -0.4054966345364D+02    -0.1851737047396D+02

      FS

   0.4054966345364D+02     0.1851737047398D+02

      FM

   0.5170836290450D+02     0.9192213577129D+02

      FN L

  -0.3894663499026D+02

      FN R

  -0.4642073514027D+02    -0.1052845844949D+03
```

a^a_{xa} θ^a $\dot{\theta}^a$

$-\dot{\theta}^{cl}$ output for unit step input at $t=0^+$

```
     FEEDFORWARD RESISTORS

R1=       32.640 KOHM

R2=       18.126 KOHM

R3=       25.676 KOHM

R4=       24.661 KOHM

R5=       98.010 KOHM

     FEEDBACK RESISTORS

R6=       58.604 KOHM

R7=       72.525 KOHM

R8=       92.019 KOHM

R9=      360.022 KOHM


          DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:COM,ANGLE,D_ANGLE $ DC G
```

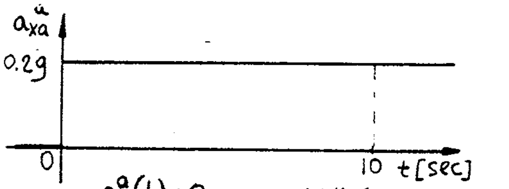
for unity step input $(t \to \infty)$

a^{a}_{xa} θ^{a} $\dot{\theta}^{a}$ ← *inputs*

```
         COL      1              2               3
ROW

     1       -2.8225D+00  -4.2657D+00  -1.6755D+01

     2       |6.3697D-01| |9.6268D-01|  3.7811D+00

     3         0.0           0.0          0.0
```

output θ^{s}

```
SIMULATION CONDITIONS
 1  XZL=    0.200 |G S|
 2  SLL=    0.0   |G S/SEC|
 3  PLMAX=  0.0  |G S|
 4  XZR=    0.0   |RAD|
 5  SLR=    0.0   |RAD/SEC|
 6  RMAX=   0.0   |RAD|
 7  DT=     0.200 [SEC]
 8  T=     10.000 [SEC]
 9  XZRV=    0.0   |RAD/SEC|
```



$\theta^{a}(t)=0$ *zero initial conditions*
$\dot{\theta}^{a}(t)=0$

```
ORDER OF OUTPUTS Y(I)
Y(1)=EOTO,     Y(2)=ESCC,     Y(3)=TETAM,   Y(4)=D_TETAM
Y(5)=YAOTO,    Y(6)=YSOTO,    Y(7)=YASCC,   Y(8)=YSSCC

         COL     1
ROW

   1        0.0

   2        0.0

   3        0.0

   4        0.0

   5        0.0

   6        0.0

   7        2.0000D-01

   8        0.0

   9        0.0
```

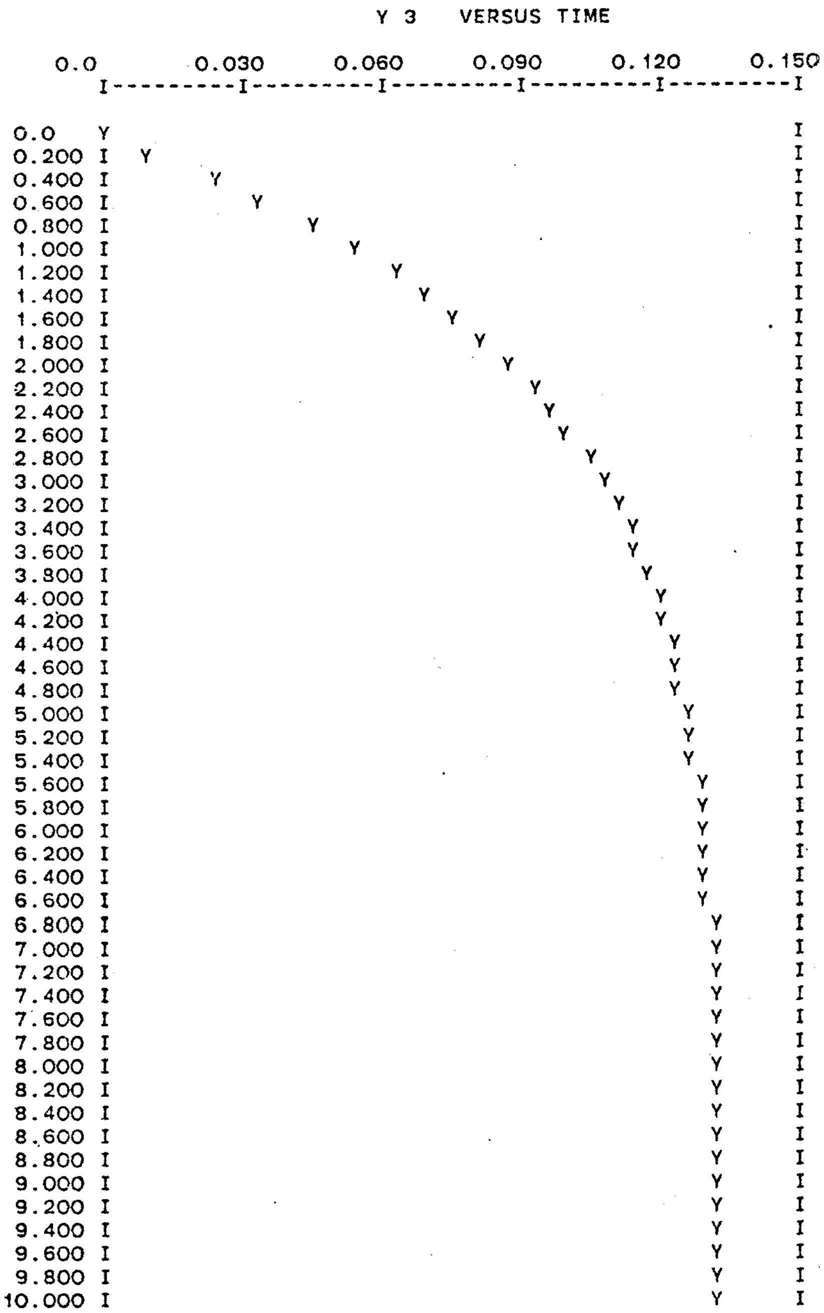
ℓ$_{oto}$ [th]

```
                           Y 1    VERSUS TIME
    0.0       1.000     2.000     3.000     4.000     5.000
       I---------I---------I---------I---------I---------I

 0.0   I                                         Y       I
 0.200 I                                      Y          I
 0.400 I                                   Y             I
 0.600 I                                Y                I
 0.800 I                             Y                   I
 1.000 I                          Y                      I
 1.200 I                        Y                        I
 1.400 I                      Y                          I
 1.600 I                    Y                            I
 1.800 I                  Y                              I
 2.000 I                 Y                               I
 2.200 I                Y                                I
 2.400 I              Y                                  I
 2.600 I             Y                                   I
 2.800 I            Y                                    I
 3.000 I            Y                                    I
 3.200 I           Y                                     I
 3.400 I          Y                                      I
 3.600 I         Y                                       I
 3.800 I         Y                                       I
 4.000 I        Y                                        I
 4.200 I        Y                                        I
 4.400 I        Y                                        I
 4.600 I       Y                                         I
 4.800 I       Y                                         I
 5.000 I       Y                                         I
 5.200 I      Y                                          I
 5.400 I      Y                                          I
 5.600 I      Y                                          I
 5.800 I      Y                                          I
 6.000 I      Y                                          I
 6.200 I      Y                                          I
 6.400 I     Y                                           I
 6.600 I     Y                                           I
 6.800 I     Y                                           I
 7.000 I     Y                                           I
 7.200 I     Y                                           I
 7.400 I     Y                                           I
 7.600 I     Y                                           I
 7.800 I     Y                                           I
 8.000 I     Y                                           I
 8.200 I     Y                                           I
 8.400 I     Y                                           I
 8.600 I     Y                                           I
 8.800 I     Y                                           I
 9.000 I     Y                                           I
 9.200 I     Y                                           I
 9.400 I     Y                                           I
 9.600 I     Y                                           I
 9.800 I     Y                                           I
10.000 I     Y                                           I
```

ℓ_{scc} [th]

```
                           Y 2   VERSUS TIME

   -3.200    -2.400    -1.600    -0.800    -0.000     0.800
       I---------I---------I---------I---------I---------I

 0.0   I                                       Y         I
 0.200 I       Y                                         I
 0.400 I           Y                                     I
 0.600 I               Y                                 I
 0.800 I                  Y                              I
 1.000 I                     Y                           I
 1.200 I                        Y                        I
 1.400 I                           Y                     I
 1.600 I                             Y                   I
 1.800 I                               Y                 I
 2.000 I                                Y                I
 2.200 I                                  Y              I
 2.400 I                                   Y             I
 2.600 I                                    Y            I
 2.800 I                                     Y           I
 3.000 I                                      Y          I
 3.200 I                                       Y         I
 3.400 I                                        Y        I
 3.600 I                                        Y        I
 3.800 I                                         Y       I
 4.000 I                                         Y       I
 4.200 I                                          Y      I
 4.400 I                                          Y      I
 4.600 I                                          Y      I
 4.800 I                                          Y      I
 5.000 I                                           Y     I
 5.200 I                                           Y     I
 5.400 I                                           Y     I
 5.600 I                                           Y     I
 5.800 I                                           Y     I
 6.000 I                                           Y     I
 6.200 I                                           Y     I
 6.400 I                                           Y     I
 6.600 I                                           Y     I
 6.800 I                                           Y     I
 7.000 I                                           Y     I
 7.200 I                                           Y     I
 7.400 I                                           Y     I
 7.600 I                                           Y     I
 7.800 I                                           Y     I
 8.000 I                                           Y     I
 8.200 I                                           Y     I
 8.400 I                                           Y     I
 8.600 I                                          Y      I
 8.800 I                                          Y      I
 9.000 I                                          Y      I
 9.200 I                                          Y      I
 9.400 I                                          Y      I
 9.600 I                                          Y      I
 9.800 I                                          Y      I
10.000 I                                          Y      I
```

θ^{m} [rad]



```
                           Y 3   VERSUS TIME

    0.0        0.030     0.060     0.090     0.120     0.150
       I---------I---------I---------I---------I---------I

   0.0 Y                                                 I
 0.200 I  Y                                              I
 0.400 I       Y                                         I
 0.600 I          Y                                      I
 0.800 I              Y                                  I
 1.000 I                 Y                               I
 1.200 I                    Y                            I
 1.400 I                      Y                          I
 1.600 I                        Y                        I
 1.800 I                          Y                      I
 2.000 I                            Y                    I
 2.200 I                              Y                  I
 2.400 I                               Y                 I
 2.600 I                                Y                I
 2.800 I                                  Y              I
 3.000 I                                   Y             I
 3.200 I                                    Y            I
 3.400 I                                     Y           I
 3.600 I                                     Y           I
 3.800 I                                      Y          I
 4.000 I                                       Y         I
 4.200 I                                       Y         I
 4.400 I                                        Y        I
 4.600 I                                        Y        I
 4.800 I                                        Y        I
 5.000 I                                         Y       I
 5.200 I                                         Y       I
 5.400 I                                         Y       I
 5.600 I                                          Y      I
 5.800 I                                          Y      I
 6.000 I                                          Y      I
 6.200 I                                          Y      I
 6.400 I                                          Y      I
 6.600 I                                          Y      I
 6.800 I                                           Y     I
 7.000 I                                           Y     I
 7.200 I                                           Y     I
 7.400 I                                           Y     I
 7.600 I                                           Y     I
 7.800 I                                           Y     I
 8.000 I                                           Y     I
 8.200 I                                           Y     I
 8.400 I                                           Y     I
 8.600 I                                           Y     I
 8.800 I                                           Y     I
 9.000 I                                           Y     I
 9.200 I                                           Y     I
 9.400 I                                           Y     I
 9.600 I                                           Y     I
 9.800 I                                           Y     I
10.000 I                                           Y     I
```

$\dot{\theta}^m$ [rad /sec]

```
                           Y 4    VERSUS TIME

    0.0       0.015     0.030     0.045     0.060     0.075
       I---------I---------I---------I---------I---------I

 0.0   Y                                                 I
 0.200 I                                          Y      I
 0.400 I                                       Y         I
 0.600 I                                   Y             I
 0.800 I                               Y                 I
 1.000 I                            Y                    I
 1.200 I                         Y                       I
 1.400 I                       Y                         I
 1.600 I                    Y                            I
 1.800 I                  Y                              I
 2.000 I                Y                                I
 2.200 I              Y                                  I
 2.400 I             Y                                   I
 2.600 I           Y                                     I
 2.800 I          Y                                      I
 3.000 I         Y                                       I
 3.200 I        Y                                        I
 3.400 I       Y                                         I
 3.600 I      Y                                          I
 3.800 I     Y                                           I
 4.000 I     Y                                           I
 4.200 I    Y                                            I
 4.400 I    Y                                            I
 4.600 I   Y                                             I
 4.800 I   Y                                             I
 5.000 I  Y                                              I
 5.200 I  Y                                              I
 5.400 I  Y                                              I
 5.600 I Y                                               I
 5.800 I Y                                               I
 6.000 I Y                                               I
 6.200 I Y                                               I
 6.400 IY                                                I
 6.600 IY                                                I
 6.800 IY                                                I
 7.000 IY                                                I
 7.200 IY                                                I
 7.400 IY                                                I
 7.600 IY                                                I
 7.800 IY                                                I
 8.000 Y                                                 I
 8.200 Y                                                 I
 8.400 Y                                                 I
 8.600 Y                                                 I
 8.800 Y                                                 I
 9.000 Y                                                 I
 9.200 Y                                                 I
 9.400 Y                                                 I
 9.600 Y                                                 I
 9.800 Y                                                 I
10.000 Y                                                 I
```

y^{a}_{oto} [th]

```
                           Y 5   VERSUS TIME

    2.000     2.500     3.000     3.500     4.000     4.500
       I---------I---------I---------I---------I---------I

 0.0   I                                            Y    I
 0.200 I                                          Y      I
 0.400 I                                        Y        I
 0.600 I                                      Y          I
 0.800 I                                     Y           I
 1.000 I                                   Y             I
 1.200 I                                 Y               I
 1.400 I                                Y                I
 1.600 I                              Y                  I
 1.800 I                             Y                   I
 2.000 I                            Y                    I
 2.200 I                          Y                      I
 2.400 I                         Y                       I
 2.600 I                        Y                        I
 2.800 I                       Y                         I
 3.000 I                      Y                          I
 3.200 I                     Y                           I
 3.400 I                   Y                             I
 3.600 I                   Y                             I
 3.800 I                  Y                              I
 4.000 I                 Y                               I
 4.200 I                Y                                I
 4.400 I               Y                                 I
 4.600 I              Y                                  I
 4.800 I             Y                                   I
 5.000 I             Y                                   I
 5.200 I            Y                                    I
 5.400 I           Y                                     I
 5.600 I          Y                                      I
 5.800 I          Y                                      I
 6.000 I         Y                                       I
 6.200 I         Y                                       I
 6.400 I        Y                                        I
 6.600 I       Y                                         I
 6.800 I       Y                                         I
 7.000 I      Y                                          I
 7.200 I      Y                                          I
 7.400 I     Y                                           I
 7.600 I     Y                                           I
 7.800 I    Y                                            I
 8.000 I    Y                                            I
 8.200 I    Y                                            I
 8.400 I   Y                                             I
 8.600 I   Y                                             I
 8.800 I  Y                                              I
 9.000 I  Y                                              I
 9.200 I  Y                                              I
 9.400 I Y                                               I
 9.600 I Y                                               I
 9.800 I Y                                               I
10.000 IY                                                I
```

$Y^{s}_{oto}[th]$

```
                           Y 6   VERSUS TIME

    0.0        0.400     0.800     1.200     1.600     2.000
       I---------I---------I---------I---------I---------I

   0.0 Y                                                 I
 0.200 I    Y                                            I
 0.400 I           Y                                     I
 0.600 I                Y                                I
 0.800 I                     Y                           I
 1.000 I                          Y                      I
 1.200 I                              Y                  I
 1.400 I                                 Y               I
 1.600 I                                    Y            I
 1.800 I                                      Y          I
 2.000 I                                        Y        I
 2.200 I                                          Y      I
 2.400 I                                           Y     I
 2.600 I                                            Y    I
 2.800 I                                             Y   I
 3.000 I                                              Y  I
 3.200 I                                               Y I
 3.400 I                                               Y I
 3.600 I                                               Y I
 3.800 I                                               Y I
 4.000 I                                               Y I
 4.200 I                                               Y I
 4.400 I                                               Y I
 4.600 I                                               Y I
 4.800 I                                               Y I
 5.000 I                                               Y I
 5.200 I                                              Y  I
 5.400 I                                              Y  I
 5.600 I                                              Y  I
 5.800 I                                             Y   I
 6.000 I                                             Y   I
 6.200 I                                             Y   I
 6.400 I                                            Y    I
 6.600 I                                            Y    I
 6.800 I                                           Y     I
 7.000 I                                           Y     I
 7.200 I                                          Y      I
 7.400 I                                          Y      I
 7.600 I                                         Y       I
 7.800 I                                         Y       I
 8.000 I                                        Y        I
 8.200 I                                        Y        I
 8.400 I                                        Y        I
 8.600 I                                       Y         I
 8.800 I                                       Y         I
 9.000 I                                      Y          I
 9.200 I                                      Y          I
 9.400 I                                      Y          I
 9.600 I                                     Y           I
 9.800 I                                     Y           I
10.000 I                                    Y            I
```

$y^{a}_{scc}[th]$

```
                          Y 7    VERSUS TIME

        0.0        1.000      2.000      3.000      4.000      5.00
          I----------I----------I----------I----------I----------I

  0.0     Y                                                      I
  0.200   Y                                                      I
  0.400   Y                                                      I
  0.600   Y                                                      I
  0.800   Y                                                      I
  1.000   Y                                                      I
  1.200   Y                                                      I
  1.400   Y                                                      I
  1.600   Y                                                      I
  1.800   Y                                                      I
  2.000   Y                                                      I
  2.200   Y                                                      I
  2.400   Y                                                      I
  2.600   Y                                                      I
  2.800   Y                                                      I
  3.000   Y                                                      I
  3.200   Y                                                      I
  3.400   Y                                                      I
  3.600   Y                                                      I
  3.800   Y                                                      I
  4.000   Y                                                      I
  4.200   Y                                                      I
  4.400   Y                                                      I
  4.600   Y                                                      I
  4.800   Y                                                      I
  5.000   Y                                                      I
  5.200   Y                                                      I
  5.400   Y                                                      I
  5.600   Y                                                      I
  5.800   Y                                                      I
  6.000   Y                                                      I
  6.200   Y                                                      I
  6.400   Y                                                      I
  6.600   Y                                                      I
  6.800   Y                                                      I
  7.000   Y                                                      I
  7.200   Y                                                      I
  7.400   Y                                                      I
  7.600   Y                                                      I
  7.800   Y                                                      I
  8.000   Y                                                      I
  8.200   Y                                                      I
  8.400   Y                                                      I
  8.600   Y                                                      I
  8.800   Y                                                      I
  9.000   Y                                                      I
  9.200   Y                                                      I
  9.400   Y                                                      I
  9.600   Y                                                      I
  9.800   Y                                                      I
 10.000   Y                                                      I
```

y^{s}_{scc} [th]

```
                           Y 8   VERSUS TIME

   -0.800    -0.000     0.800     1.600     2.400     3.200
       I---------I---------I---------I---------I---------I

   0.0 I         Y                                       I
 0.200 I                                         Y       I
 0.400 I                                     Y           I
 0.600 I                                 Y               I
 0.800 I                              Y                  I
 1.000 I                           Y                     I
 1.200 I                        Y                        I
 1.400 I                     Y                           I
 1.600 I                   Y                             I
 1.800 I                 Y                               I
 2.000 I                Y                                I
 2.200 I              Y                                  I
 2.400 I             Y                                   I
 2.600 I            Y                                    I
 2.800 I           Y                                     I
 3.000 I          Y                                      I
 3.200 I         Y                                       I
 3.400 I        Y                                        I
 3.600 I        Y                                        I
 3.800 I       Y                                         I
 4.000 I       Y                                         I
 4.200 I      Y                                          I
 4.400 I      Y                                          I
 4.600 I      Y                                          I
 4.800 I      Y                                          I
 5.000 I     Y                                           I
 5.200 I     Y                                           I
 5.400 I     Y                                           I
 5.600 I     Y                                           I
 5.800 I     Y                                           I
 6.000 I     Y                                           I
 6.200 I     Y                                           I
 6.400 I     Y                                           I
 6.600 I     Y                                           I
 6.800 I     Y                                           I
 7.000 I     Y                                           I
 7.200 I     Y                                           I
 7.400 I     Y                                           I
 7.600 I     Y                                           I
 7.800 I     Y                                           I
 8.000 I     Y                                           I
 8.200 I     Y                                           I
 8.400 I     Y                                           I
 8.600 I      Y                                          I
 8.800 I      Y                                          I
 9.000 I      Y                                          I
 9.200 I      Y                                          I
 9.400 I      Y                                          I
 9.600 I      Y                                          I
 9.800 I      Y                                          I
10.000 I      Y                                          I
```

$\dot{\theta}^{cl}$ [Volt]

```
                           U 1   VERSUS TIME
   -2.000     0.0       2.000     4.000     6.000     8.000
       I---------I---------I---------I---------I---------I
 0.0   I                                                UI
 0.200 I               U                                 I
 0.400 I             U                                   I
 0.600 I            U                                    I
 0.800 I            U                                    I
 1.000 I           U                                     I
 1.200 I           U                                     I
 1.400 I          U                                      I
 1.600 I          U                                      I
 1.800 I         U                                       I
 2.000 I         U                                       I
 2.200 I         U                                       I
 2.400 I        U                                        I
 2.600 I        U                                        I
 2.800 I        U                                        I
 3.000 I        U                                        I
 3.200 I        U                                        I
 3.400 I       U                                         I
 3.600 I       U                                         I
 3.800 I       U                                         I
 4.000 I       U                                         I
 4.200 I       U                                         I
 4.400 I       U                                         I
 4.600 I       U                                         I
 4.800 I       U                                         I
 5.000 I       U                                         I
 5.200 I       U                                         I
 5.400 I      U                                          I
 5.600 I      U                                          I
 5.800 I      U                                          I
 6.000 I      U                                          I
 6.200 I      U                                          I
 6.400 I      U                                          I
 6.600 I      U                                          I
 6.800 I      U                                          I
 7.000 I      U                                          I
 7.200 I      U                                          I
 7.400 I      U                                          I
 7.600 I      U                                          I
 7.800 I      U                                          I
 8.000 I      U                                          I
 8.200 I      U                                          I
 8.400 I      U                                          I
 8.600 I      U                                          I
 8.800 I      U                                          I
 9.000 I      U                                          I
 9.200 I      U                                          I
 9.400 I      U                                          I
 9.600 I      U                                          I
 9.800 I      U                                          I
10.000 I      U                                          I
```

WASHOUT # 2

```
 GAT-1 LINK PITCH AXIS WASHOUT DESIGN PROGRAM

    WRITTEN BY JEHUDA ISH-SHALOM (OCTOBER 1981)


MOTION BASE COMMAND INPUT: ANGULER VELOCITY
MEASURED STATES: ANGULER POSTION AND VELOCITY, AND LINEAR ACCELERATION IN MOTION BASE INIRIAL AXE
VESTIBULAR MODEL INPUTS:
          LINEAR  = OTOLITH : LINEAR ACCELERATION
          ANGULER = SEMICIRCULAR CANAL: ANGULER VELOCITY


WAHOUT FILTER DESIGN PARAMETERS (11/16/81 18*17*03.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.10D+00 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
*  20 RHO=    0.10D+00 WEIGHT OF MOTION OVER ERROR
```

```
OPEN LOOP EIGENVALUES


                              ---EIGENVALUES---
          REAL PART       IMAG PART      NAT FREQ(HZ)       ZETA          FREQ(HZ)

   1     -3.351D+00       0.0             5.333D-01        1.000000       0.0
   2      2.208D-01       0.0             3.515D-02       -1.000000       0.0
   3     -1.900D-01       0.0             3.024D-02        1.000000       0.0
   4     -1.900D-01       0.0             3.024D-02        1.000000       0.0
   5     -1.695D-01       0.0             2.698D-02        1.000000       0.0
   6     -1.695D-01       0.0             2.698D-02        1.000000       0.0
   7     -1.000D-01       0.0             1.592D-02        1.000000       0.0
   8     -1.000D-01       0.0             1.592D-02        1.000000       0.0
   9     -1.000D-05       0.0             1.592D-06        1.000000       0.0


                           CLOSED-LOOP EIGENVALUES
          REAL PART       IMAG PART      NAT FREQ(HZ)       ZETA          FREQ(HZ)

   1     -2.134D+01       0.0             3.396D+00        1.000000       0.0
   2     -2.119D-01       0.0             3.372D-02        1.000000       0.0
   3     ~~-1.900D-01~~       0.0             3.024D-02        1.000000       0.0
   4     ~~-1.695D-01~~       0.0             2.698D-02        1.000000       0.0
   5     -1.038D-01       2.377D-02       1.695D-02        0.974782       3.783D-03
   6     -1.038D-01      -2.377D-02       1.695D-02        0.974782       3.783D-03
   7     ~~-1.000D-01~~       0.0             1.592D-02        1.000000       0.0
   8     ~~-1.000D-01~~       0.0             1.592D-02        1.000000       0.0
   9     ~~-1.000D-05~~       0.0             1.592D-06        1.000000       0.0
OPTIMAL CLOSED LOOP MATRIX ACL



          COL    1               2               3               4               5               6
 ROW
   1        -1.9000D-01   0.0             0.0             0.0             0.0             0.0



                 7               8               9

            -1.1400D-01  -1.1400D-01   0.0
```

```
    FA
-0.9470371705083D+01    -0.4447424996780D+02
    FS
 0.9470371705070D+01     0.4447424996786D+02
    FM
 0.1965131945427D+02     0.1093861176746D+03
```

```
    FN L
```

a^{a}_{xa}

```
-0.8650986146106D+01
    FN R
-0.1316097564195D+02    -0.1141712685073D+03
```

θ^{a} $\dot{\theta}^{a}$

$-\dot{\theta}^{cl}$ outputs FOR UNIT STEP INPUTS AT $t=0^{+}$

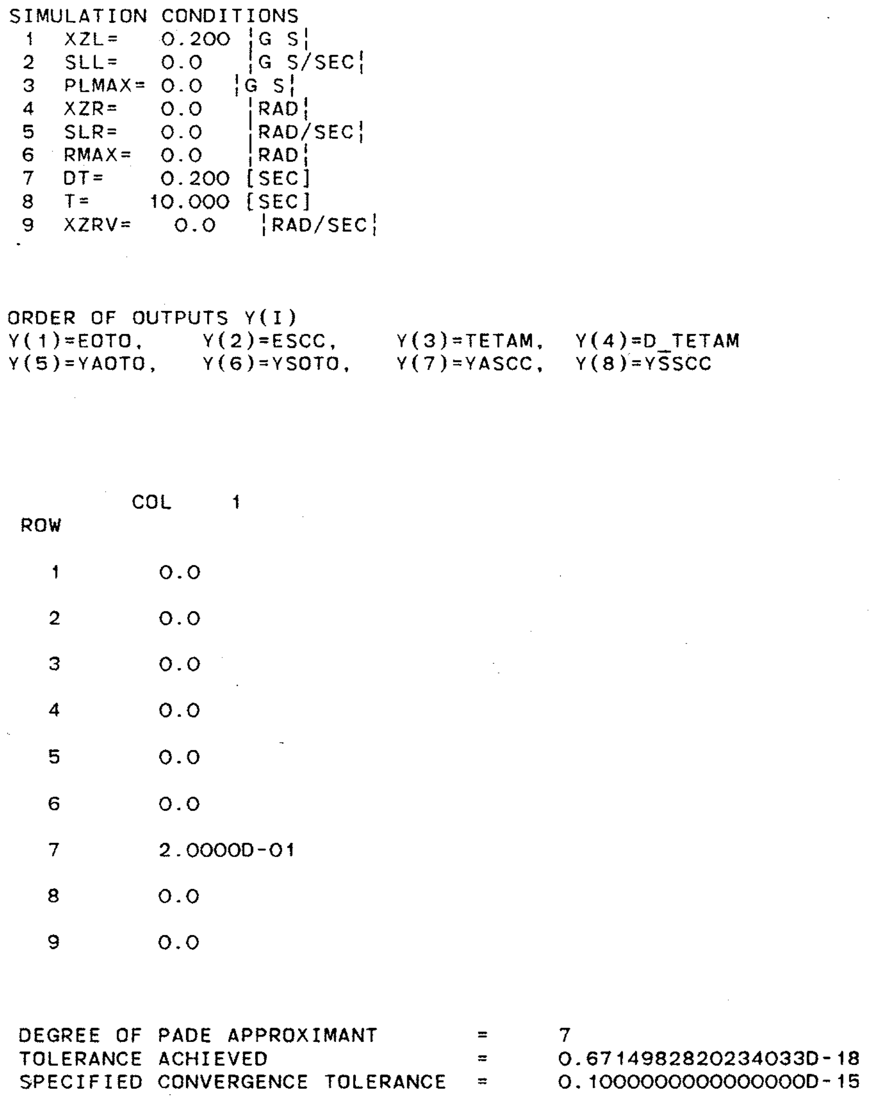
```
      FEEDFORWARD RESISTORS
R1=      115.125 KOHM
R2=       16.715 KOHM
R3=      115.594 KOHM
R4=      105.592 KOHM
R5=       40.807 KOHM
      FEEDBACK RESISTORS
R6=      154.204 KOHM
R7=       60.946 KOHM
R8=      394.002 KOHM
R9=      149.899 KOHM


         DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:COM,ANGLE,D_ANGLE $ DC G
```

FOR UNITY step INPUT AT $T \to \infty$

a^{a}_{xa} θ^{a} $\dot{\theta}^{a}$ ← INPUTS

```
        COL     1             2             3
  ROW
    1       -1.3792D+00  -3.4745D+00  -3.2380D+01
    2        3.1126D-01   7.8410D-01   7.3073D+00
    3        0.0          0.0          0.0
```

θ^{S} (row 2): $\boxed{3.1126D-01}$ $\boxed{7.8410D-01}$ ← OUTPUTS



```
SIMULATION CONDITIONS
 1  XZL=    0.200 ¦G S¦
 2  SLL=    0.0   ¦G S/SEC¦
 3  PLMAX= 0.0  ¦G S¦
 4  XZR=    0.0   ¦RAD¦
 5  SLR=    0.0   ¦RAD/SEC¦
 6  RMAX=   0.0   ¦RAD¦
 7  DT=     0.200 [SEC]
 8  T=     10.000 [SEC]
 9  XZRV=    0.0   ¦RAD/SEC¦


ORDER OF OUTPUTS Y(I)
Y(1)=EOTO,    Y(2)=ESCC,    Y(3)=TETAM,  Y(4)=D_TETAM
Y(5)=YAOTO,   Y(6)=YSOTO,   Y(7)=YASCC,  Y(8)=YSSCC



         COL     1
 ROW

   1        0.0

   2        0.0

   3        0.0

   4        0.0

   5        0.0

   6        0.0

   7        2.0000D-01

   8        0.0

   9        0.0


 DEGREE OF PADE APPROXIMANT          =     7
 TOLERANCE ACHIEVED                  =     0.67149828202340330D-18
 SPECIFIED CONVERGENCE TOLERANCE     =     0.10000000000000000D-15
```

e_{oto} [th]

```
                           Y 1    VERSUS TIME
                                                               X 10**  0
    0.800     1.600     2.400     3.200     4.000     4.800
       I---------I---------I---------I---------I---------I

0.0    I                                          Y      I
 0.200 I                                        Y        I
 0.400 I                                      Y          I
 0.600 I                                     Y           I
 0.800 I                                   Y             I
 1.000 I                                 Y               I
 1.200 I                                Y                I
 1.400 I                               Y                 I
 1.600 I                             Y                   I
 1.800 I                            Y                    I
 2.000 I                           Y                     I
 2.200 I                         Y                       I
 2.400 I                        Y                        I
 2.600 I                       Y                         I
 2.800 I                      Y                          I
 3.000 I                     Y                           I
 3.200 I                    Y                            I
 3.400 I                   Y                             I
 3.600 I                   Y                             I
 3.800 I                  Y                              I
 4.000 I                 Y                               I
 4.200 I                Y                                I
 4.400 I                Y                                I
 4.600 I               Y                                 I
 4.800 I              Y                                  I
 5.000 I              Y                                  I
 5.200 I             Y                                   I
 5.400 I             Y                                   I
 5.600 I            Y                                    I
 5.800 I           Y                                     I
 6.000 I           Y                                     I
 6.200 I           Y                                     I
 6.400 I          Y                                      I
 6.600 I          Y                                      I
 6.800 I         Y                                       I
 7.000 I         Y                                       I
 7.200 I         Y                                       I
 7.400 I        Y                                        I
 7.600 I        Y                                        I
 7.800 I        Y                                        I
 8.000 I       Y                                         I
 8.200 I       Y                                         I
 8.400 I       Y                                         I
 8.600 I       Y                                         I
 8.800 I      Y                                          I
 9.000 I      Y                                          I
 9.200 I      Y                                          I
 9.400 I      Y                                          I
 9.600 I      Y                                          I
 9.800 I     Y                                           I
10.000 I     Y                                           I
```

e_{scc} [th]

```
                           Y 2   VERSUS TIME
                                                               X 10**  0
   -0.500    -0.375    -0.250    -0.125     0.0       0.125
       I---------I---------I---------I---------I---------I

 0.0   I                                       Y         I
 0.200 Y                                                 I
 0.400 I Y                                               I
 0.600 I    Y                                            I
 0.800 I      Y                                          I
 1.000 I         Y                                       I
 1.200 I           Y                                     I
 1.400 I             Y                                   I
 1.600 I               Y                                 I
 1.800 I                 Y                               I
 2.000 I                   Y                             I
 2.200 I                     Y                           I
 2.400 I                      Y                          I
 2.600 I                        Y                        I
 2.800 I                         Y                       I
 3.000 I                           Y                     I
 3.200 I                            Y                    I
 3.400 I                             Y                   I
 3.600 I                               Y                 I
 3.800 I                                Y                I
 4.000 I                                 Y               I
 4.200 I                                  Y              I
 4.400 I                                   Y             I
 4.600 I                                   Y             I
 4.800 I                                    Y            I
 5.000 I                                     Y           I
 5.200 I                                      Y          I
 5.400 I                                      Y          I
 5.600 I                                       Y         I
 5.800 I                                        Y        I
 6.000 I                                        Y        I
 6.200 I                                         Y       I
 6.400 I                                         Y       I
 6.600 I                                          Y      I
 6.800 I                                          Y      I
 7.000 I                                          Y      I
 7.200 I                                           Y     I
 7.400 I                                           Y     I
 7.600 I                                           Y     I
 7.800 I                                           Y     I
 8.000 I                                            Y    I
 8.200 I                                            Y    I
 8.400 I                                            Y    I
 8.600 I                                            Y    I
 8.800 I                                            Y    I
 9.000 I                                             Y   I
 9.200 I                                             Y   I
 9.400 I                                             Y   I
 9.600 I                                             Y   I
 9.800 I                                             Y   I
10.000 I                                             Y   I
```

θ^m [rad]

```
                    Y 3   VERSUS TIME
                                                               X 10**  0
    0.0       0.012     0.025     0.037     0.050     0.062
       I---------I---------I---------I---------I---------I

 0.0   Y                                                 I
 0.200 I Y                                               I
 0.400 I   Y                                             I
 0.600 I     Y                                           I
 0.800 I       Y                                         I
 1.000 I        Y                                        I
 1.200 I          Y                                      I
 1.400 I            Y                                    I
 1.600 I              Y                                  I
 1.800 I               Y                                 I
 2.000 I                 Y                               I
 2.200 I                  Y                              I
 2.400 I                    Y                            I
 2.600 I                     Y                           I
 2.800 I                      Y                          I
 3.000 I                        Y                        I
 3.200 I                         Y                       I
 3.400 I                          Y                      I
 3.600 I                           Y                     I
 3.800 I                            Y                    I
 4.000 I                             Y                   I
 4.200 I                              Y                  I
 4.400 I                               Y                 I
 4.600 I                                Y                I
 4.800 I                                 Y               I
 5.000 I                                  Y              I
 5.200 I                                   Y             I
 5.400 I                                    Y            I
 5.600 I                                    Y            I
 5.800 I                                     Y           I
 6.000 I                                      Y          I
 6.200 I                                       Y         I
 6.400 I                                       Y         I
 6.600 I                                        Y        I
 6.800 I                                        Y        I
 7.000 I                                         Y       I
 7.200 I                                          Y      I
 7.400 I                                          Y      I
 7.600 I                                           Y     I
 7.800 I                                           Y     I
 8.000 I                                            Y    I
 8.200 I                                            Y    I
 8.400 I                                             Y   I
 8.600 I                                             Y   I
 8.800 I                                             Y   I
 9.000 I                                              Y  I
 9.200 I                                              Y  I
 9.400 I                                              Y  I
 9.600 I                                               Y I
 9.800 I                                               Y I
10.000 I                                               Y I
```

$\dot{\theta}^m$ [rad/sec]

```
                           Y 4   VERSUS TIME
                                                               X 10** -3
    0.0       3.000     6.000     9.000    12.000    15.000
       I---------I---------I---------I---------I---------I

 0.0   Y                                                 I
 0.200 I                                          Y      I
 0.400 I                                         Y       I
 0.600 I                                        Y        I
 0.800 I                                      Y          I
 1.000 I                                     Y           I
 1.200 I                                   Y             I
 1.400 I                                  Y              I
 1.600 I                                 Y               I
 1.800 I                                Y                I
 2.000 I                              Y                  I
 2.200 I                             Y                   I
 2.400 I                            Y                    I
 2.600 I                           Y                     I
 2.800 I                          Y                      I
 3.000 I                         Y                       I
 3.200 I                        Y                        I
 3.400 I                       Y                         I
 3.600 I                      Y                          I
 3.800 I                     Y                           I
 4.000 I                     Y                           I
 4.200 I                    Y                            I
 4.400 I                   Y                             I
 4.600 I                  Y                              I
 4.800 I                 Y                               I
 5.000 I                 Y                               I
 5.200 I                Y                                I
 5.400 I               Y                                 I
 5.600 I               Y                                 I
 5.800 I              Y                                  I
 6.000 I              Y                                  I
 6.200 I             Y                                   I
 6.400 I            Y                                    I
 6.600 I            Y                                    I
 6.800 I           Y                                     I
 7.000 I           Y                                     I
 7.200 I          Y                                      I
 7.400 I          Y                                      I
 7.600 I          Y                                      I
 7.800 I         Y                                       I
 8.000 I         Y                                       I
 8.200 I        Y                                        I
 8.400 I        Y                                        I
 8.600 I        Y                                        I
 8.800 I       Y                                         I
 9.000 I       Y                                         I
 9.200 I       Y                                         I
 9.400 I      Y                                          I
 9.600 I      Y                                          I
 9.800 I      Y                                          I
10.000 I     Y                                           I
```

y^{a}_{oto} [th]

```
                           Y 5    VERSUS TIME
                                                               X 10**  0
    2.000     2.500     3.000     3.500     4.000     4.500
       I---------I---------I---------I---------I---------I

 0.0   I                                            Y    I
 0.200 I                                          Y      I
 0.400 I                                        Y        I
 0.600 I                                      Y          I
 0.800 I                                     Y           I
 1.000 I                                   Y             I
 1.200 I                                 Y               I
 1.400 I                                Y                I
 1.600 I                              Y                  I
 1.800 I                             Y                   I
 2.000 I                            Y                    I
 2.200 I                          Y                      I
 2.400 I                         Y                       I
 2.600 I                        Y                        I
 2.800 I                       Y                         I
 3.000 I                      Y                          I
 3.200 I                     Y                           I
 3.400 I                   Y                             I
 3.600 I                   Y                             I
 3.800 I                  Y                              I
 4.000 I                 Y                               I
 4.200 I                Y                                I
 4.400 I               Y                                 I
 4.600 I              Y                                  I
 4.800 I             Y                                   I
 5.000 I             Y                                   I
 5.200 I            Y                                    I
 5.400 I           Y                                     I
 5.600 I          Y                                      I
 5.800 I          Y                                      I
 6.000 I         Y                                       I
 6.200 I         Y                                       I
 6.400 I        Y                                        I
 6.600 I       Y                                         I
 6.800 I       Y                                         I
 7.000 I      Y                                          I
 7.200 I      Y                                          I
 7.400 I     Y                                           I
 7.600 I     Y                                           I
 7.800 I    Y                                            I
 8.000 I    Y                                            I
 8.200 I    Y                                            I
 8.400 I   Y                                             I
 8.600 I   Y                                             I
 8.800 I  Y                                              I
 9.000 I  Y                                              I
 9.200 I  Y                                              I
 9.400 I Y                                               I
 9.600 I Y                                               I
 9.800 I Y                                               I
10.000 IY                                                I
```

y^{s}_{oto} [th]

```
                            Y 6    VERSUS TIME
                                                                X 10**  O
     0.0       0.200     0.400     0.600     0.800     1.000
        I---------I---------I---------I---------I---------I

  0.0   Y                                                 I
  0.200 I Y                                               I
  0.400 I    Y                                            I
  0.600 I      Y                                          I
  0.800 I         Y                                       I
  1.000 I           Y                                     I
  1.200 I             Y                                   I
  1.400 I               Y                                 I
  1.600 I                 Y                               I
  1.800 I                  Y                              I
  2.000 I                    Y                            I
  2.200 I                      Y                          I
  2.400 I                       Y                         I
  2.600 I                        Y                        I
  2.800 I                         Y                       I
  3.000 I                           Y                     I
  3.200 I                            Y                    I
  3.400 I                             Y                   I
  3.600 I                              Y                  I
  3.800 I                              Y                  I
  4.000 I                               Y                 I
  4.200 I                                Y                I
  4.400 I                                 Y               I
  4.600 I                                 Y               I
  4.800 I                                  Y              I
  5.000 I                                  Y              I
  5.200 I                                   Y             I
  5.400 I                                   Y             I
  5.600 I                                    Y            I
  5.800 I                                    Y            I
  6.000 I                                    Y            I
  6.200 I                                    Y            I
  6.400 I                                     Y           I
  6.600 I                                     Y           I
  6.800 I                                     Y           I
  7.000 I                                     Y           I
  7.200 I                                     Y           I
  7.400 I                                      Y          I
  7.600 I                                      Y          I
  7.800 I                                      Y          I
  8.000 I                                      Y          I
  8.200 I                                      Y          I
  8.400 I                                      Y          I
  8.600 I                                      Y          I
  8.800 I                                      Y          I
  9.000 I                                      Y          I
  9.200 I                                      Y          I
  9.400 I                                      Y          I
  9.600 I                                      Y          I
  9.800 I                                      Y          I
 10.000 I                                     Y           I
```

y^{a}_{scc} [th]

```
                            Y 7    VERSUS TIME
                                                                   X 10**  0
        0.0        1.000      2.000      3.000      4.000      5.000
          I---------I---------I---------I---------I---------I

   0.0    Y                                                  I
   0.200  Y                                                  I
   0.400  Y                                                  I
   0.600  Y                                                  I
   0.800  Y                                                  I
   1.000  Y                                                  I
   1.200  Y                                                  I
   1.400  Y                                                  I
   1.600  Y                                                  I
   1.800  Y                                                  I
   2.000  Y                                                  I
   2.200  Y                                                  I
   2.400  Y                                                  I
   2.600  Y                                                  I
   2.800  Y                                                  I
   3.000  Y                                                  I
   3.200  Y                                                  I
   3.400  Y                                                  I
   3.600  Y                                                  I
   3.800  Y                                                  I
   4.000  Y                                                  I
   4.200  Y                                                  I
   4.400  Y                                                  I
   4.600  Y                                                  I
   4.800  Y                                                  I
   5.000  Y                                                  I
   5.200  Y                                                  I
   5.400  Y                                                  I
   5.600  Y                                                  I
   5.800  Y                                                  I
   6.000  Y                                                  I
   6.200  Y                                                  I
   6.400  Y                                                  I
   6.600  Y                                                  I
   6.800  Y                                                  I
   7.000  Y                                                  I
   7.200  Y                                                  I
   7.400  Y                                                  I
   7.600  Y                                                  I
   7.800  Y                                                  I
   8.000  Y                                                  I
   8.200  Y                                                  I
   8.400  Y                                                  I
   8.600  Y                                                  I
   8.800  Y                                                  I
   9.000  Y                                                  I
   9.200  Y                                                  I
   9.400  Y                                                  I
   9.600  Y                                                  I
   9.800  Y                                                  I
  10.000  Y                                                  I
```

$y^{s}_{scc}[th]$

```
                           Y 8   VERSUS TIME
                                                               X 10**  0
   -0.125     0.0       0.125     0.250     0.375     0.500
       I---------I---------I---------I---------I---------I

 0.0   I         Y                                       I
 0.200 I                                                 Y
 0.400 I                                               Y I
 0.600 I                                            Y    I
 0.800 I                                          Y      I
 1.000 I                                       Y         I
 1.200 I                                     Y           I
 1.400 I                                   Y             I
 1.600 I                                 Y               I
 1.800 I                               Y                 I
 2.000 I                             Y                   I
 2.200 I                           Y                     I
 2.400 I                          Y                      I
 2.600 I                        Y                        I
 2.800 I                       Y                         I
 3.000 I                     Y                           I
 3.200 I                    Y                            I
 3.400 I                   Y                             I
 3.600 I                 Y                               I
 3.800 I                Y                                I
 4.000 I               Y                                 I
 4.200 I              Y                                  I
 4.400 I             Y                                   I
 4.600 I             Y                                   I
 4.800 I            Y                                    I
 5.000 I           Y                                     I
 5.200 I          Y                                      I
 5.400 I          Y                                      I
 5.600 I         Y                                       I
 5.800 I        Y                                        I
 6.000 I        Y                                        I
 6.200 I       Y                                         I
 6.400 I       Y                                         I
 6.600 I      Y                                          I
 6.800 I      Y                                          I
 7.000 I      Y                                          I
 7.200 I     Y                                           I
 7.400 I     Y                                           I
 7.600 I     Y                                           I
 7.800 I     Y                                           I
 8.000 I    Y                                            I
 8.200 I    Y                                            I
 8.400 I    Y                                            I
 8.600 I    Y                                            I
 8.800 I    Y                                            I
 9.000 I   Y                                             I
 9.200 I   Y                                             I
 9.400 I   Y                                             I
 9.600 I   Y                                             I
 9.800 I   Y                                             I
10.000 I   Y                                             I
```

$\dot{\theta}^{cl}$ [Volt]

```
                           U 1    VERSUS TIME
                                                              X 10**  0
   -0.500    -0.000     0.500     1.000     1.500    2.000
       I---------I---------I---------I---------I---------I

 0.0   I                                            U    I
 0.200 I              U                                  I
 0.400 I             U                                   I
 0.600 I             U                                   I
 0.800 I            U                                    I
 1.000 I            U                                    I
 1.200 I            U                                    I
 1.400 I           U                                     I
 1.600 I           U                                     I
 1.800 I           U                                     I
 2.000 I          U                                      I
 2.200 I          U                                      I
 2.400 I          U                                      I
 2.600 I          U                                      I
 2.800 I         U                                       I
 3.000 I         U                                       I
 3.200 I         U                                       I
 3.400 I         U                                       I
 3.600 I        U                                        I
 3.800 I        U                                        I
 4.000 I        U                                        I
 4.200 I        U                                        I
 4.400 I        U                                        I
 4.600 I       U                                         I
 4.800 I       U                                         I
 5.000 I       U                                         I
 5.200 I       U                                         I
 5.400 I       U                                         I
 5.600 I       U                                         I
 5.800 I      U                                          I
 6.000 I      U                                          I
 6.200 I      U                                          I
 6.400 I      U                                          I
 6.600 I      U                                          I
 6.800 I      U                                          I
 7.000 I      U                                          I
 7.200 I     U                                           I
 7.400 I     U                                           I
 7.600 I     U                                           I
 7.800 I     U                                           I
 8.000 I     U                                           I
 8.200 I     U                                           I
 8.400 I     U                                           I
 8.600 I     U                                           I
 8.800 I     U                                           I
 9.000 I     U                                           I
 9.200 I     U                                           I
 9.400 I    U                                            I
 9.600 I    U                                            I
 9.800 I    U                                            I
10.000 I    U                                            I
```

Appendix II: Design Parameters of the Washouts Used

Note that the values of the feed-forward resistors that one has to use are twice that computed by the program for washout designs made before March 1982.

WASHOUT #4

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/22/82 16*57*16.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=    21.168 (1/G)
   2  BO=     0.076 (RAD/SEC)
   3  AO=     0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=    40.000 (1/RAD)
   5  AS=     0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=    0.100 (RAD/SEC)
   7  BNR=    0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=     3.130 (1/SEC)
  10  BBR=    0.167 (RAD/(SEC**2*VOLT))
  11  PIP=0.74D+00 (1/SEC**2)
  12  BI= 0.0      (1/SEC)
GRAVITY CONSTANT
  13  GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
  14  PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
  15  PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
  16  PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
  17  QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
  18  RRO/R=  0.0      ANGLE/VELOCITY COMMAND
  19  RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
*  20 RHO=    0.20D-01 WEIGHT OF MOTION OVER ERROR
```

```
     FEEDFORWARD RESISTORS

R1=       14.376 KOHM
R2=        8.006 KOHM
R3=       11.300 KOHM   X2
R4=       10.860 KOHM
R5=       43.185 KOHM

     FEEDBACK RESISTORS

R6=       27.493 KOHM
R7=       29.975 KOHM
R8=       40.523 KOHM
R9=      158.634 KOHM
```

WASHOUT #5

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/16/82 15*13*03.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
*  20 RHO=    0.10D-01 WEIGHT OF MOTION OVER ERROR



     FEEDFORWARD RESISTORS

R1=      10.142 KOHM

R2=       5.658 KOHM

R3=       7.968 KOHM  } X2

R4=       7.659 KOHM

R5=      30.461 KOHM

     FEEDBACK RESISTORS

R6=      19.664 KOHM

R7=      20.789 KOHM

R8=      28.580 KOHM

R9=     111.895 KOHM
```

WASHOUT #6

LINK LIKE WASHOUT

R_1 ;	51.7	49.9 + 1.78
R_2 :	206.32	196 + 10K
R_6 :	24.98	21.5 + 3.48
R_7 :	71.93	68.1 + 3.83

WASHOUT # 7

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 18*02*57.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.10D-01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR



     FEEDFORWARD RESISTORS

R1=      356.971 KOHM

R2=       10.854 KOHM

R3=      531.113 KOHM   X2

R4=      454.411 KOHM

R5=       17.104 KOHM

     FEEDBACK RESISTORS

R6=      255.077 KOHM

R7=       36.641 KOHM

R8=     1695.564 KOHM

R9=       62.830 KOHM
```

WASHOUT # 8

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 18*02*18.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=    21.168 (1/G)
   2  BO=     0.076 (RAD/SEC)
   3  AO=     0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=    40.000 (1/RAD)
   5  AS=     0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=    0.100 (RAD/SEC)
   7  BNR=    0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=     3.130 (1/SEC)
   10 BBR=    0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=   -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.32D-01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=     0.40D-01 WEIGHT OF MOTION OVER ERROR



     FEEDFORWARD RESISTORS

R1=      149.531 KOHM

R2=       10.362 KOHM

R3=      186.719 KOHM

R4=      164.369 KOHM

R5=       19.758 KOHM

     FEEDBACK RESISTORS

R6=      180.233 KOHM

R7=       36.606 KOHM

R8=      613.316 KOHM

R9=       72.577 KOHM
```

WASHOUT # 9

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 18*01*32.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.10D+00 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
*  20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=       69.090 KOHM

R2=        9.990 KOHM

R3=       71.182 KOHM

R4=       64.821 KOHM

R5=       25.020 KOHM

     FEEDBACK RESISTORS

R6=      109.169 KOHM

R7=       36.610 KOHM

R8=      241.870 KOHM

R9=       91.907 KOHM
```

WASHOUT # 10

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 17*59*11.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.32D+00 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
*  20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=       34.523 KOHM )
                      |
R2=        9.950 KOHM |
                      |
R3=       30.339 KOHM > X2
                      |
R4=       28.488 KOHM |
                      |
R5=       35.297 KOHM )

     FEEDBACK RESISTORS

R6=       61.654 KOHM

R7=       37.310 KOHM

R8=      106.298 KOHM

R9=      129.659 KOHM
```

WASHOUT # 11

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 18*03*35.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.32D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=       16.481 KOHM )
                      |
R2=       16.839 KOHM |
                      |
R3=       12.101 KOHM } X2
                      |
R4=       11.803 KOHM |
                      |
R5=      149.805 KOHM )

     FEEDBACK RESISTORS

R6=       31.292 KOHM

R7=       68.248 KOHM

R8=       44.042 KOHM

R9=      550.282 KOHM
```

WASHOUT # 12

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 18*04*29.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
  10 BBR=    0.167 (RAD/(SEC**2*VOLT))
  11 PIP=0.74D+00 (1/SEC**2)
  12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
  13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
  14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
  15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
  16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.10D+02 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
  18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
  19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
  20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=       15.529 KOHM

R2=       26.676 KOHM

R3=       10.974 KOHM  X2

R4=       10.797 KOHM

R5=      436.416 KOHM

     FEEDBACK RESISTORS

R6=       29.574 KOHM

R7=      119.568 KOHM

R8=       40.288 KOHM

R9=     1603.100 KOHM
```

WASHOUT # 13

```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 18*05*27.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=    21.168 (1/G)
   2  BO=     0.076 (RAD/SEC)
   3  AO=     0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=    40.000 (1/RAD)
   5  AS=     0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=    0.100 (RAD/SEC)
   7  BNR=    0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=     3.130 (1/SEC)
  10  BBR=    0.167 (RAD/(SEC**2*VOLT))
  11  PIP=0.74D+00 (1/SEC**2)
  12  BI= 0.0       (1/SEC)
GRAVITY CONSTANT
  13  GEE=   -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
  14  PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
  15  PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
  16  PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.32D+02 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
  18  RRO/R=  0.0      ANGLE/VELOCITY COMMAND
  19  RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
  20  RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=       15.196 KOHM

R2=       37.695 KOHM

R3=       10.540 KOHM   X2

R4=       10.415 KOHM

R5=     1334.665 KOHM

     FEEDBACK RESISTORS

R6=       28.971 KOHM

R7=      191.593 KOHM

R8=       38.862 KOHM

R9=     4902.668 KOHM
```

WASHOUT #14

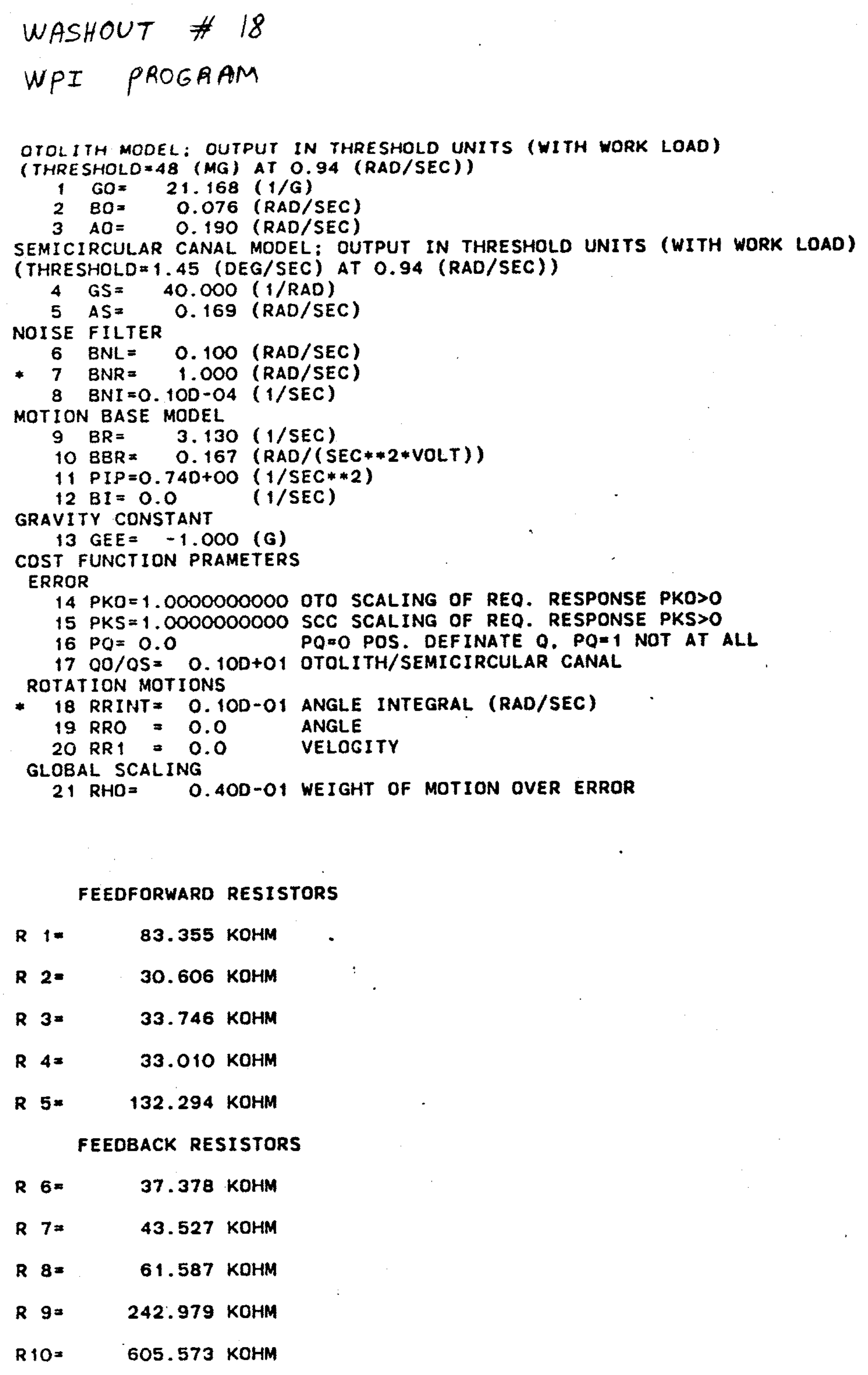
```
WAHOUT FILTER DESIGN PARAMETERS ( 1/19/82 18*06*31.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
*  17 QO/QS=  0.10D+03 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=       15.068 KOHM

R2=       45.464 KOHM

R3=       10.370 KOHM  X2

R4=       10.265 KOHM

R5=     4168.924 KOHM

     FEEDBACK RESISTORS

R6=       28.738 KOHM

R7=      255.172 KOHM

R8=       38.304 KOHM

R9=    15313.848 KOHM
```

WASHOUT # 15

```
WAHOUT FILTER DESIGN PARAMETERS (11/12/81 13*38*57.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.0      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
*  20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=      20.414 KOHM

R2=      11.348 KOHM

R3=      16.054 KOHM

R4=      15.426 KOHM

R5=      61.330 KOHM

     FEEDBACK RESISTORS

R6=      38.261 KOHM

R7=      43.549 KOHM

R8=      57.560 KOHM

R9=     225.284 KOHM
```

WASHOUT # 17

```
WAHOUT FILTER DESIGN PARAMETERS ( 3/13/82  2*58*37.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
*  7  BNR=   1.000 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18 RRINT=  0.80D-01 ANGLE INTEGRAL (RAD/SEC)
   19 RRO   =  0.0      ANGLE
   20 RR1   =  0.0      VELOCITY
 GLOBAL SCALING
   21 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


         FEEDFORWARD RESISTORS

R 1=       92.315 KOHM

R 2=       32.011 KOHM

R 3=       36.558 KOHM

R 4=       36.570 KOHM

R 5=      148.244 KOHM

         FEEDBACK RESISTORS

R 6=       36.044 KOHM

R 7=       43.493 KOHM

R 8=       68.227 KOHM

R 9=      272.275 KOHM

R10=      214.282 KOHM
```



WASHOUT # 18
WPI PROGRAM

```
OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
*  7  BNR=   1.000 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q. PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18 RRINT=  0.10D-01 ANGLE INTEGRAL (RAD/SEC)
   19 RRO  =  0.0      ANGLE
   20 RR1  =  0.0      VELOCITY
 GLOBAL SCALING
   21 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


      FEEDFORWARD RESISTORS

R  1=       83.355 KOHM

R  2=       30.606 KOHM

R  3=       33.746 KOHM

R  4=       33.010 KOHM

R  5=      132.294 KOHM

      FEEDBACK RESISTORS

R  6=       37.378 KOHM

R  7=       43.527 KOHM

R  8=       61.587 KOHM

R  9=      242.979 KOHM

R10=       605.573 KOHM
```

WASHOUT #19

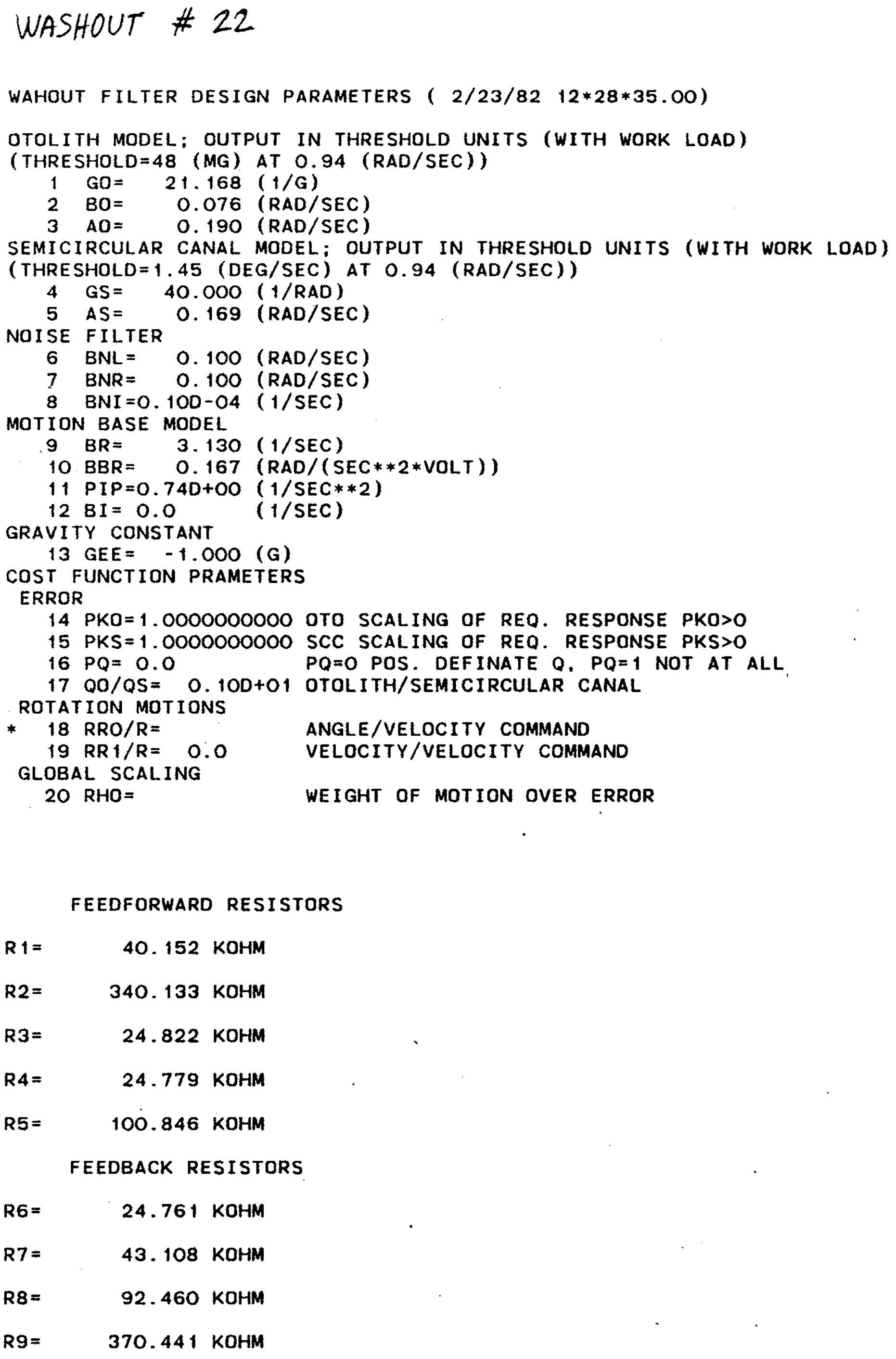
```
 WAHOUT FILTER DESIGN PARAMETERS (11/11/81  2*43*26.00)

 OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
 (THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
    1  GO=   21.168 (1/G)
    2  BO=    0.076 (RAD/SEC)
    3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
    4  GS=   40.000 (1/RAD)
    5  AS=    0.169 (RAD/SEC)
NOISE FILTER
    6  BNL=   0.100 (RAD/SEC)
    7  BNR=   0.100 (RAD/SEC)
    8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
    9  BR=    3.130 (1/SEC)
   10  BBR=   0.167 (RAD/(SEC**2*VOLT))
   11  PIP=0.74D+00 (1/SEC**2)
   12  BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13  GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14  PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15  PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16  PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
   17  QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18  RRO/R=  0.16D+02 ANGLE/VELOCITY COMMAND
   19  RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
   20  RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=      22.333 KOHM )

R2=      14.778 KOHM |

R3=      16.780 KOHM > X2

R4=      16.221 KOHM |

R5=      64.700 KOHM )

     FEEDBACK RESISTORS

R6=      36.734 KOHM

R7=      43.514 KOHM

R8=      60.526 KOHM

R9=     237.665 KOHM
```

WASHOUT #20

```
WAHOUT FILTER DESIGN PARAMETERS (11/11/81  2*44*46.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
*  16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18 RRO/R=  0.30D+02 ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
   20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=      23.790 KOHM

R2=      18.033 KOHM

R3=      17.359 KOHM   X2

R4=      16.850 KOHM

R5=      67.365 KOHM

     FEEDBACK RESISTORS

R6=      35.578 KOHM

R7=      43.486 KOHM

R8=      62.873 KOHM

R9=     247.455 KOHM
```

WASHOUT #21

```
WAHOUT FILTER DESIGN PARAMETERS ( 2/23/82 12*27*39.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18 RRO/R=  0.10D+03 ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
*  20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR



     FEEDFORWARD RESISTORS

R1=      29.457 KOHM

R2=      39.571 KOHM

R3=      19.784 KOHM  } X2

R4=      19.455 KOHM

R5=      78.383 KOHM

     FEEDBACK RESISTORS

R6=      31.301 KOHM

R7=      43.365 KOHM

R8=      72.592 KOHM

R9=     287.926 KOHM
```



WASHOUT # 22

```
WAHOUT FILTER DESIGN PARAMETERS ( 2/23/82 12*28*35.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18 RRO/R=           ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=             WEIGHT OF MOTION OVER ERROR


     FEEDFORWARD RESISTORS

R1=      40.152 KOHM

R2=     340.133 KOHM

R3=      24.822 KOHM

R4=      24.779 KOHM

R5=     100.846 KOHM

     FEEDBACK RESISTORS

R6=      24.761 KOHM

R7=      43.108 KOHM

R8=      92.460 KOHM

R9=     370.441 KOHM
```

Appendix I: Flaps-Down Detection Experiment Results

This appendix presents the data for 5 pilots in the flaps down experiments. The parameter q in the comment column corresponds to Qo/Qs. Run No. labled with & were excluded from the average calculation.

Pilot: JH March/26/82 Visual Pitch and Roll on

RUN NO.	COMMENT	Td sec	Tc sec	Tp sec	ΔH feet	COMMENTS
&1	q=0.32	1.8	1.7	1.9	*	
&2		1.0	2.2	2.3	50	
&3		0.8	2.3	2.4	40	
&4		0.8	0.8	1.2	30	control down up down => large Tp
&5		0.8	2.1	2.6	35	
6		0.5	0.6	0.7	10	control down up down => large Tp
&7		2.2	2.2	2.5	10	
8		0.6	1.0	1.9	10	
9		0.6	0.8	2.1	0	
10	I.C. app. 0	0.6	0.7	1.4	0-5	
11		0.7	1.3	1.5	20	
12	I.C. not 0	2.0	2.1	2.3	10-15	run discarded
13		0.6	0.7	1.3	10	
14	prejudged	*	*	*	20	run discarded
&15		1.9	1.9	2.3	25	
16	q=1.0	0.5	0.6	1.1	15	
17		0.5	0.9	1.1	15	
18		0.5	0.5	0.9	25	
19	prejudged	*	*	*	10	
20	I.C. not 0	0.4	0.6	0.7	4	run discarded
21		0.4	0.5	0.7	0	
&22		0.4	0.5	1.5	10	control down up down but small
23		0.4	0.5	0.8	5	
24		0.5	0.6	1.0	20	
25		0.4	0.5	1.1	0	
26	q=0.32 I.C. not 0	0.5	*	*	50	run discarded
27	I.C. not 0	0.8	0.7	0.9	15	run discarded
28	prejudged	*	*	*	30	
29		0.5	0.7	0.9	20	
30		0.5	0.6	0.9	5	
31		0.7	1.0	1.2	10	
32	I.C. not 0	1.8	1.9	2.3	5	run discarded (EL not 0)
33		0.5	0.6	1.0	10	
&34	Link	2.9	2.8	3.1	50	
35		3.0	2.1	2.4	10	
36	I.C. not 0	1.4	1.5	1.7	15	run discarded
37		3.0	1.6	2.7	25	
38	I.C. not 0	1.0	0.9	1.2	5	run discarded
39	prejudged	*	*	*	5	
40	I.C. not 0	0.7	3.0	3.4	10	run discarded
41	I.C. not 0	2.2	2.0	2.2	5	run discarded
42		2.5	2.1	2.3	40	
43	prejudged	*	*	*	10	
44		2.5	1.6	2.0	20	

&45	q=.1	1.9	1.9	2.0	30	
46		1.5	1.4	1.8	10	
47		1.9	1.3	1.7	20	
48		1.4	1.3	1.5	5	
49		1.5	1.5	1.7	20	
50	I.C. not 0	2.5	1.7	2.2	5	run discarded
&51	q=0.32	0.9	0.9	1.2	5	
52		0.6	0.7	0.9	5	
53	q=1.0	0.3	0.6	0.8	5	
54		0.5	0.6	0.8	15	
55		0.5	0.7	1.0	5	
56	q=3.2	0.4	0.4	0.7	30	
57		0.4	0.5	0.7	5	
58	prejudged	0.1	0.0	0.4	5	small controls, run discarded
59		0.4	0.6	0.7	5	
60		0.4	0.5	0.7	5	
61		0.4	0.5	0.7	5	less over controling

Pilot IM APR.16 visuals on

RUN NO.	COMMENT	Td sec	Tc sec	Tp sec	ΔH feet	COMMENTS
&1	q=.56	1.1	1.7	2.3	20	
&2		1.2	1.7	1.9	10	
3		0.5	0.7	1.0	5	
4		0.7	0.8	1.0	20	
5		0.6	0.8	1.1	0	
&6		1.1	1.3	1.5	20	
7		0.5	0.8	1.1	0	
8		0.7	1.1	1.3	0	
9		0.5	0.6	0.8	20	
10		0.6	0.7	1.1	0	
&11	q=.32	0.6	0.6	0.9	20	
12		0.5	0.7	1.0	0	
13		0.6	0.7	1.1	10	
14		0.9	1.0	1.5	0	
15		1.4	1.5	1.7	0	I.C. not 0
16		0.7	1.0	1.4	0	
17		0.6	0.7	1.1	10	
18		0.7	0.8	1.3	10	
19		0.6	0.8	1.0	10	
20		0.6	0.6	0.8	10	
21	q=1.0	0.5	0.7	1.0	0	
22		0.5	0.8	1.1	0	
23		0.6	0.7	1.0	0	
24		0.5	0.8	1.1	20	I.C. not 0
25		0.5	0.7	0.9	0	
26		0.5	0.6	0.9	10	
27		0.5	0.6	0.9	10	
28		0.6	0.8	1.1	10	I.C. not 0
29		0.6	0.7	2.1	0	I.C. not 0
30		0.5	0.6	0.9	0	I.C. not 0
31	q=3.2	0.5	0.7	1.0	0	
32		0.4	0.5	0.8	0	

33		0.5	0.6	0.8	10	
34		0.4	0.6	0.9	0	I.C. not 0
35		0.5	0.7	0.9	0	
36		0.5	0.7	0.9	10	
37	q=1(RRI)	0.6	0.7	1.0	20	I.C. not 0
38		0.4	0.5	1.0	0	
39		0.7	0.7	0.9	0	
40		0.4	0.7	1.0	0	
41		0.6	0.7	1.8	0	
42		0.5	0.6	0.9	0	
43		0.6	0.8	1.9	10	
44		0.6	0.7	1.3	0	
45		0.6	0.8	1.2	0	
46		0.5	0.6	0.8	10	
&47	Link	1.3	1.9	2.3	10	
48		1.9	0.5	1.4	0	I.C. not 0
49		1.6	1.6	1.8		
50		1.0	1.1	1.4	0	
51		1.2	1.7	2.0	0	
2		3.0	1.9	2.8	60	Pilot not ready
53		1.5	1.7	2.0	0	
&54	no motion	1.7	1.8	2.0	20	
55		1.7	1.4	2.1	0	
56		1.5	1.5	2.0	0	
57		2.0	2.2	2.4	0	
58		1.6	1.7	1.9	10	
59		1.5	1.5	2.0	0	I.C. not 0
60		2.0	2.0	2.5	0	

Note: q=1(RRI) corresponds to runs done with the integrator feedback

Pilot JW Apr.16th visual pitch and roll

RUN	COMMENT	Td	Tc	Tp	ΔH	COMMENTS
NO.		sec	sec	sec	feet	
1	q=1.0	*	0.8	2.8	*	control up; IC not 0
2		0.7	0.6	2.6	*	control up;
3		0.8	1.1	2.3	20	control up; IC not 0
4		1.5	1.7	3.3	20	control up; IC not 0
&5		*	0.6	3.2	30	control up;
&6		1.2	0.7	2.3	10	control up;
7		0.6	0.6	3.0	30	control up; IC not 0
8		0.7	1.5	1.8	50	
9		0.8	1.4	2.0	40	control up down small bump
10		0.6	1.2	1.7	60	control up down small bump
11	ASI cov'd	0.6	0.8	1.9	30	control down up small vally
&12		0.5	0.7	1.5	60	control up;
13		0.6	1.1	1.8	40	IC not 0
14		0.5	1.7	1.9	30	IC not 0
15		0.7	1.4	1.6	40	IC not 0
&16	q=.32	2.1	2.7	3.0	*	
17		1.0	1.7	2.2	50	
18		1.2	3.2	3.4	50	
19		0.7	2.1	2.3	20	

20		1.0	2.3	2.6	50	
&21	q=.1	2.4	2.6	2.9	60	
22		1.1	1.8	2.2	40	
23		0.9	1.6	1.8	40	
24		2.0	2.3	2.6	40	IC not 0
25		1.2	2.0	2.1	20	IC not 0
26	Link	3.0	3.2	3.5	40	IC not 0
27	prejudged	3.0	3.3	3.7	30	IC not 0
28		3.2	3.3	3.7	30	IC not 0
29		2.2	2.5	2.9	40	
30		1.3	1.8	2.3	20	
31	q=3.2	0.6	0.5	3.0	60	control up; I.C. not 0
32		1.1	0.4	3.6	100	control up; I.C. not 0
&33		0.5	0.8	2.8	60	control up;
34		0.4	0.6	2.7	20	control up small; run discarded
35		0.4	0.8	1.0	20	sharp control
36		0.4	0.8	2.6	0	gradual control
37		0.5	0.8	2.7	40	gradual control
38		0.7	1.1	2.4	*	IC not 0; gradual control
39		0.4	0.8	2.6	0	gradual control
40		0.3	0.8	1.6	20	gradual control
41	q=1.0	0.6	2.2	2.5	50	sharp control
42		0.6	1.6	2.4	40	
43		0.7	1.1	2.4	40	
44		0.6	1.1	3.4	20	IC not 0; gradual control
45		0.8	0.4	2.5	40	gradual control
46		0.6	2.2	2.5	30	IC not 0; sharp control
47	Link	2.4	2.5	3.1	30	IC not 0; sharp control
48		3.3	3.5	3.9	20	IC not 0; sharp control
49		2.4	3.0	3.4	30	IC not 0; sharp control
50		2.8	3.0	3.4	10	sharp control
51		1.4	1.8	2.1	5	IC not 0; sharp control
52		2.3	2.7	3.0	10	IC not 0; sharp control
53		1.6	1.7	2.0	20	IC not 0; sharp control
54		2.4	2.3	2.7	10	IC not 0; sharp control
55	q=.56	0.6	0.8	1.1	30	sharp control
56		1.0	1.5	2.6		control up
57		0.8	0.9	2.6	40	control up small
58		0.5	1.3	1.4	20	sharp control
59		0.7	1.8	2.0	30	sharp control
60		0.9	1.5	2.3	50	IC not 0; gradual control

Pilot AE Apr/21/82 Visuals pitch and roll on

RUN NO.	COMMENT	Td sec	Tc sec	Tp sec	ΔH feet	COMMENTS
1	Link			6.4		
2		2.2	2.1	2.9	*	
3		2.8	2.0	2.7	25	
4		1.3	1.5	1.2	25	
5		2.1	0.6	2.3	20	
6		0.8	2.6	2.7	20	
7		3.3	0.8	4.0	40	
8		2.4	0.5	2.9	22	

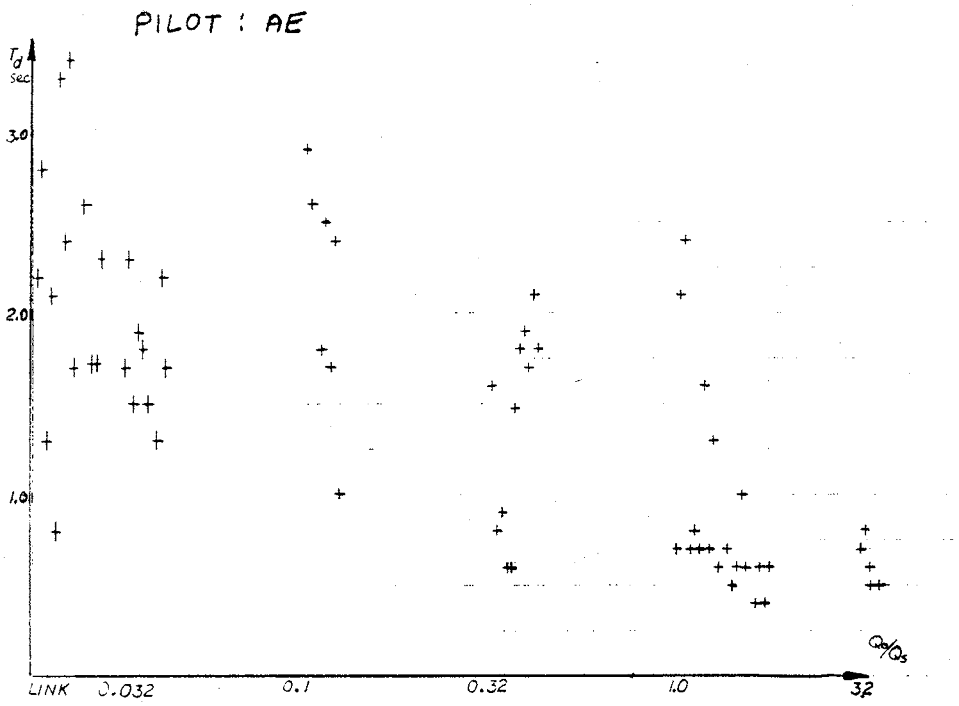
9		3.4	0.8	1.9	*
10		1.7	0.8	1.8	20
11	q=3.2	0.7	0.6	3.3	70
12		0.8	0.6	3.2	85
13		0.5	0.5	3.4	50
14		0.6	0.5	3.5	40
15		0.5	0.8	3.1	40
16	q=1.0	0.7	0.4	1.8	35
17		2.1	0.5	2.8	70
18		2.4	0.2	2.3	60
19		0.7	0.3	1.9	25
20		0.8	0.2	2.3	10
21		0.7	0.8	2.1	25
22		1.6	0.4	2.0	20
23		0.7	0.6	1.7	10
24		1.3	0.5	1.5	20
25		0.6	0.6	1.6	
26	rest	0.7	0.8	2.0	60
27		0.5	0.6	1.3	20
28		0.6	0.9	1.7	
29		1.0	1.8	1.8	20
30		0.6	0.7	1.9	
31		4.6	0.5	2.7	70
32		0.4	0.9	1.9	40
33		0.6	0.9	1.7	10
34		0.4	0.6	1.4	40
35		0.6	0.6	1.4	40
36	q=.32	1.6	1.6	1.9	20
37		0.8	0.8	1.4	10
38		0.9	1.2	2.4	
39		0.6	1.3	1.7	0
40		0.6	0.3	1.8	10
41		1.8	1.7	1.9	40
42		1.9	1.4	2.1	25
43		1.7	1.7	2.2	40
44		2.1	0.7	2.4	25
45		1.8	1.6	2.1	15
46	q=.1	2.9	0.2	3.0	50
47		2.6	1.4	2.9	40
48		1.0	1.3	2.3	35
49		1.8	1.9	2.3	25
50		2.5	0.6	2.8	40
51		1.7	1.8	2.1	0
52		2.2	2.0	2.3	35
53		1.7	1.8	2.0	20
54		2.4	1.4	2.4	0
55		*	*	*	5
56	q=.032	1.7	*	1.9	50
57		2.3	0.6	2.5	15-20
58		1.5	1.6	1.8	25
59		1.9	2.0	2.3	20
60		1.8	1.8	2.1	20
61		1.5	1.0	1.7	25
62		1.3	1.5	1.7	20
63		2.2	2.2	2.5	0
64		0.2	0.2	0.3	20
65		1.7	1.5	1.9	20

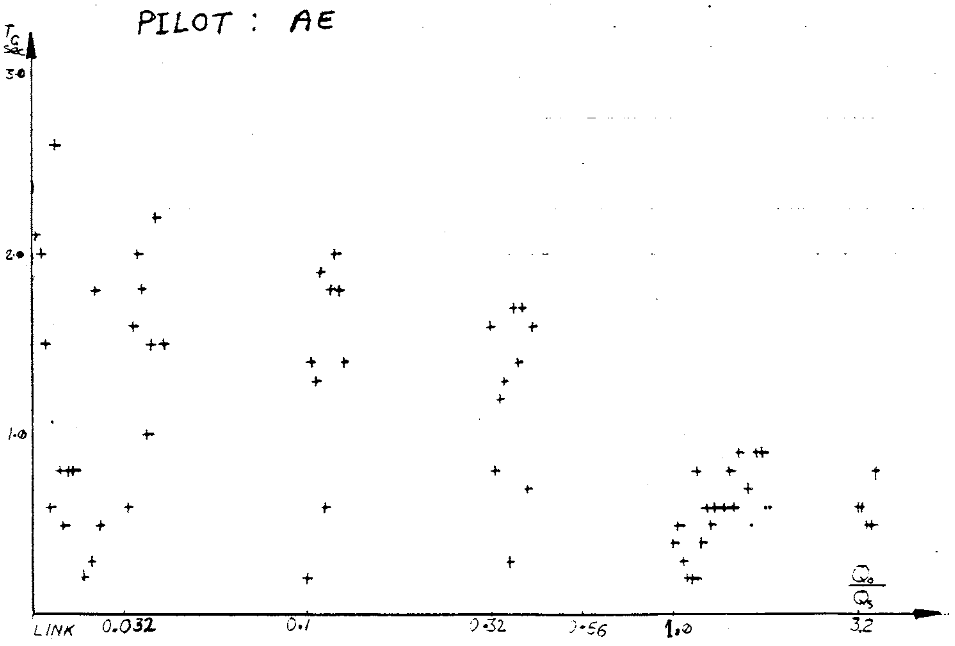
66	Link	2.6	0.2	2.8	35
67		*	*	*	20
68		1.7	1.3	1.7	0 .
69		1.7	1.8	2.1	25
70		2.3	0.5	2.6	40

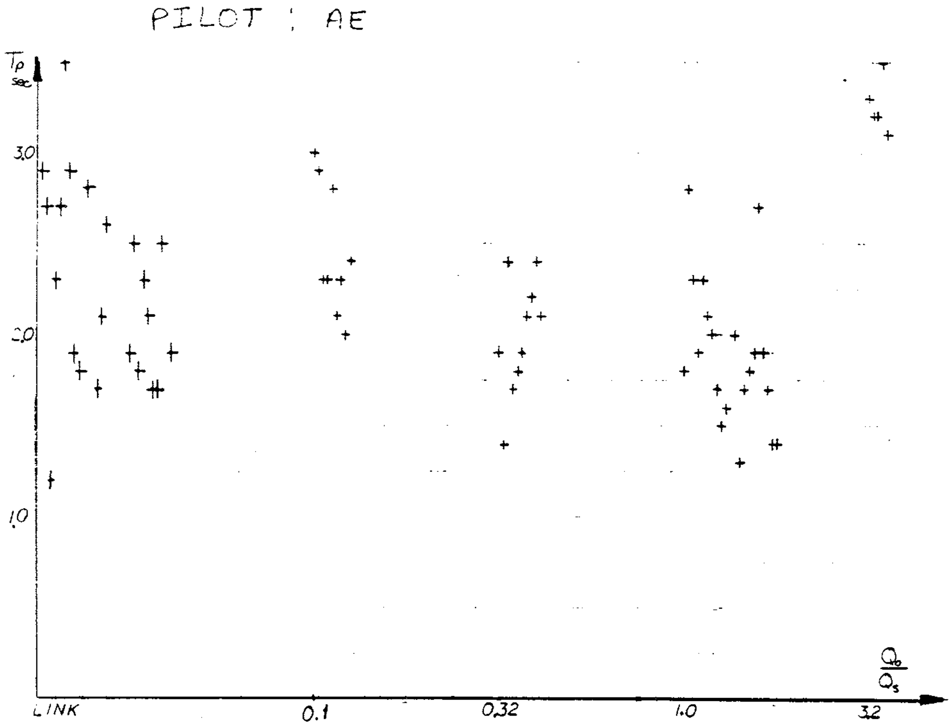
Pilot LH APR/21/82 Visual Pitch and Roll on

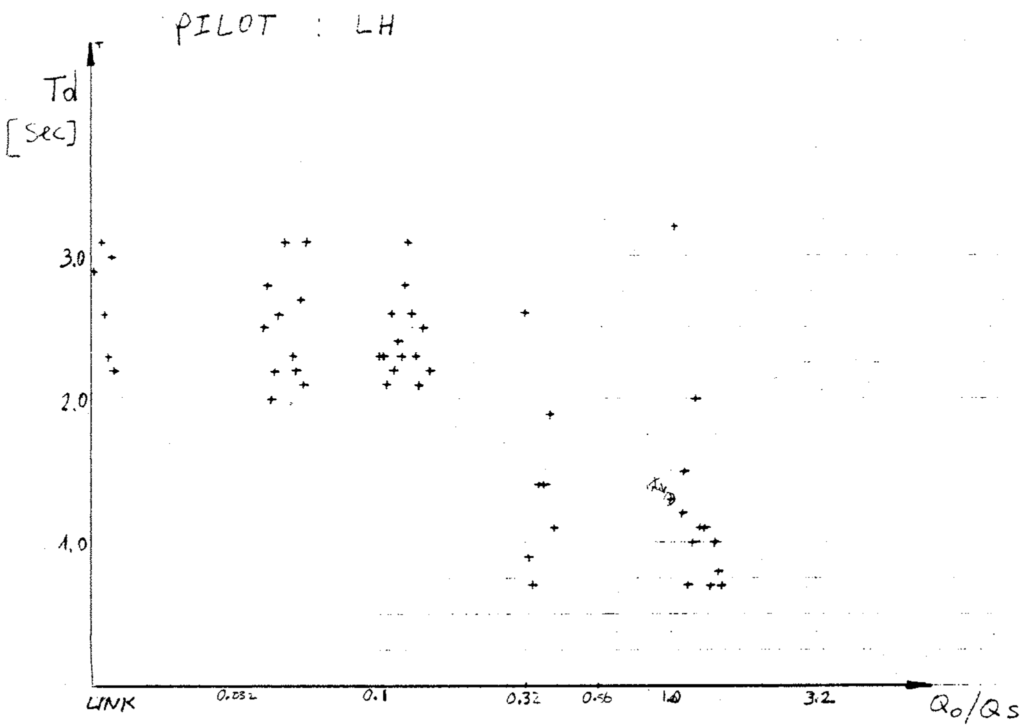
RUN NO.	COMMENT	Td sec	Tc sec	Tp sec	ΔH feet	COMMENTS
1	q=.032	3.9	*	2.8	50	
2		2.7	*	1.8	50	
3		3.2	*	2.9	0	
4		1.5	*	0.3	0	
5		2.7	*	1.5	25	
6		5.5	2.5	2.9	0	
7		0.6	2.8	*	*	
8		*	1.8	2.4	25	
9		2.5	*	2.8	0	
10		2.8	0.4	2.1	50	
11		2.0	0.5	2.1	20	
12		2.2	0.5	1.2	0	
13		2.6	1.7	2.2	0	
14		*	0.9	1.9	10	
15		3.1	0.8	1.7	10	
16	q=.1	2.3	2.0	2.3	30	
17		2.3	2.1	2.3	10	
18		2.1	*	2.3	40	
19		2.6	1.7	2.0	20	
20		2.2	1.5	2.1	100	
21		2.4	2.2	2.6	20	
22		2.3	2.3	2.6	0	
23		2.8	*	2.3	20	
24		3.1	1.9	2.4	50	
25		2.6	*	2.5	10	
26		2.3	1.8	2.2	20	
27		2.1	1.9	2.2	40	
28		2.5	0.9	2.4	20	
29		4.2	0.4	3.1	40	
30		2.2	0.8	2.1	40	
31	q=.32	2.6	0.5	2.7	50	
32		0.9	0.9	1.1	0	
33		0.7	0.8	1.0	0	
34		3.5	0.4	3.0	50	
35		3.5	0.6	2.0	40	
36		1.4	0.3	*	0	
37		1.4	0.5	1.1	40	
38		4.2	0.6	4.0	40	
39		1.9	0.4	2.1	0	
40		1.1	*	1.1	0	
41	q=1.0	*	.8	2.2	100	
42						
43		3.2	0.3	3.1	10	
44		0	0.8	1.2	40	

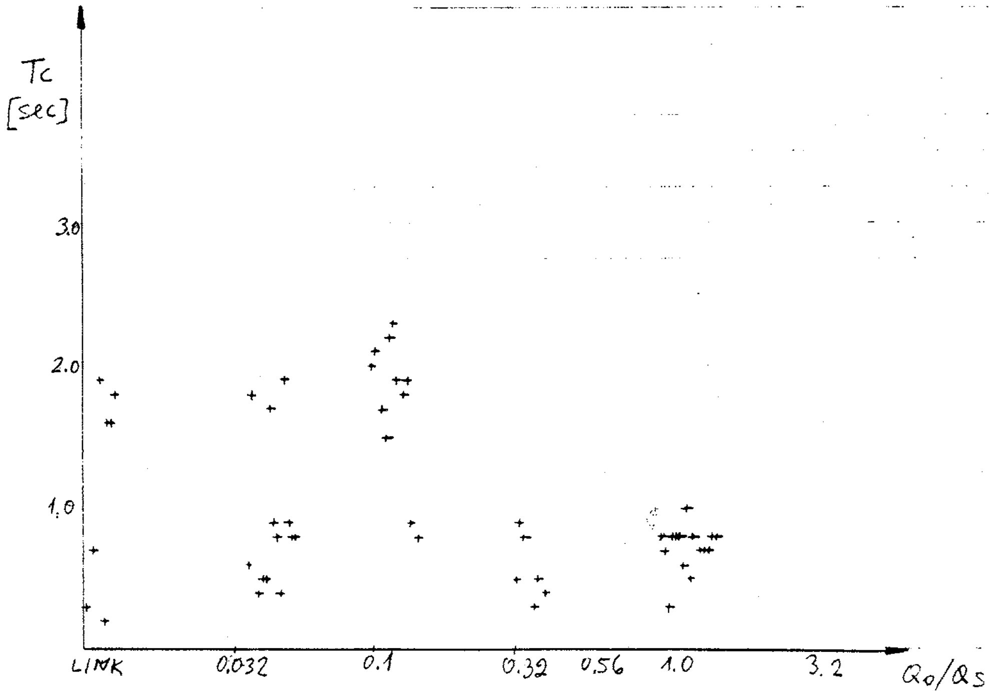
45		1.2	0.8	*	0
46		1.5	0.8	1.3	0
47		4.0	0.7	3.7	80
48		0.7	0.6	*	*
49		1.0	1.0	17	20
50		2.0	0.5	3.6	
51		1.1	0.8	1.5	0
51.5		1.1	0.7	1.0	20
52		1.9	*	1.9	20
53		0.7	0.7	1.0	0
54		1.0	0.7	0.9	10
55		0.8	0.8	1.2	20
56		0.7	0.8	1.0	0
57	Link	2.9	0.3	2.3	20
58		4.5	*	2.6	30
59		3.1	0.7	2.5	0
60		2.6	*	2.7	20
61		2.3	1.9	2.2	10
62		3.7	0.2	2.4	30
63		2.3	1.6	2.0	20
64		3.0	1.6	3.1	0
65		2.2	1.8	2.0	15
66	q=.032	2.3	0.4	2.2	10
67		2.2	1.9	2.3	10
68		2.7	0.9	2.6	10
69		2.1	0.8	2.1	10
70		3.1	0.8	2.8	


PILOT : AE
T_d sec
3.0
2.0
1.0
LINK
0.032
0.1
0.32
1.0
3.2
Q_o/Q_s

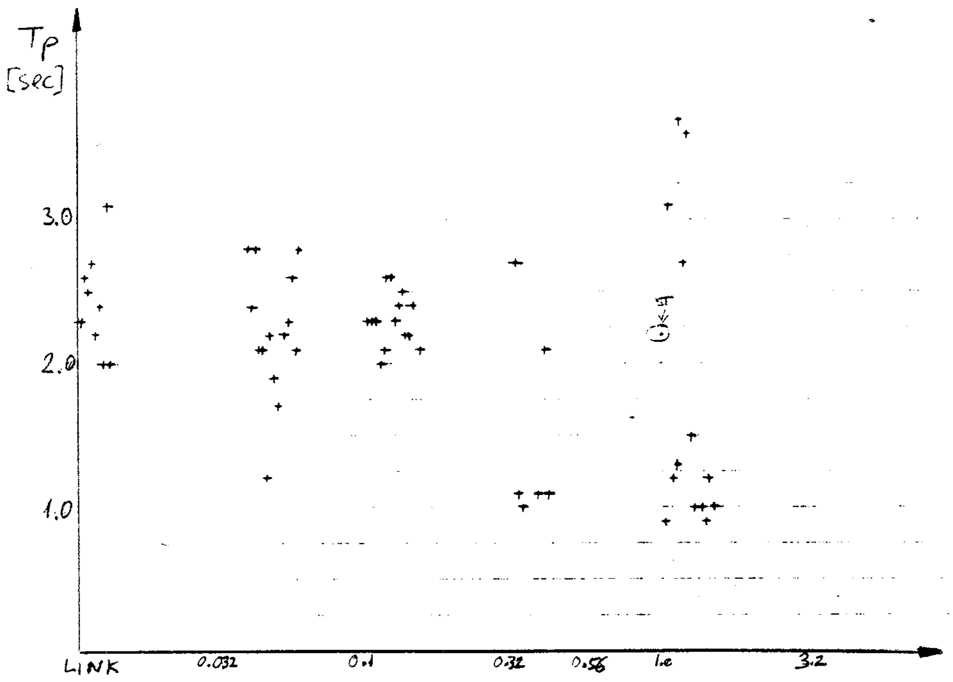

PILOT : AE
T_c sec
3.0
2.0
1.0
LINK
0.032
0.1
0.32
0.56
1.0
3.2
$\frac{Q_o}{Q_s}$


PILOT : AE
Tp sec
3.0
2.0
1.0
LINK
0.1
0.32
1.0
3.2
Qb/Qs


PILOT : LH
Td
[sec]
3.0
2.0
1.0
LINK
0.032
0.1
0.32
0.56
1.0
3.2
Qo/Qs


Tc
[sec]
3.0
2.0
1.0
LINK
0.032
0.1
0.32
0.56
1.0
3.2
Qo/Qs

PILOT : LH



Appendix J: Experience of Pilots Used

0 JI - 1 hour (not pilot, experimenter)
1 IM - 150 hours in Cessna 150, including aerobatics
2 JW - 120 hours in Cessna 150
3 SU - 120 hours in Cessna 150
4 JN - over 1000 hours in single engine airplanes
5 JF - 800 hours in single engine airplanes (has experience on other GAT-1 simulators)
6 JH - 2800 hours, mostly on single engine airplanes, also an instructor on Cessna 150
7 YI - 1500 hours, fighter pilot (Israel AF)
8 CI - 1500 hours, fighter pilot (Israel AF)
9 WH - thousands of hours in single, multi and jet airplanes (was in USAF)
10 DH - 200 hours, mostly in Cessna 150
11 JL - 80 hours, in single engine airplanes
12 CO - 270 hours, mostly in Cessna 150
13 DM - 30 hours, student pilot
14 AE - 320 hours, single engine airplanes (140 in Cessna 150)
15 LN - over 1000 hours, light airplanes
16 GO - 6500 hours DC8 pilot and instructor on Cessna 150
17 PM - 16000 hours DC8 pilot
18 EA - 12000 hours DC8 pilot
19 YM - thousands of hours as fighter pilot in Israel AF
20 EP - 2300 hours (USAF)

Appendix K: Experiments to Demonstrate the Optimal Washout System

The objective of this lab is to expose you to some of the trade-offs and methods of motion design for moving base flight simulators. This will be done as follows:

1) You will use a design program, running on the IPS IBM 370/168, to find several possible optimal washout designs. These designs will be optimal for your choice of cost parameters, as explained in the first lab (3/9/82). Your challenge is to find the best design, using the given cost parameters, for the GAT-1 flight simulator during flights similar to those shown in Figure 5. How could you further improve this design given more free parameters? Which ones would you choose?

2) In the first lab session, you will experience flying the GAT-1 simulator and see what this is really all about. To simplify things, you will fly only with longitudinal motion (pitch, surge, heave only).

3) In the second lab (3/16/82) you will fly the GAT-1 from take-off to landing (approximately 4 minutes each run) with three types of pitch motion:

   a) no pitch motion

   b,c) Two different optimal washouts: One like washout #0 and one like washout #2 (both shown in Figure 5). Which do you think is better and why?

   The tape recorded comments of each student along with his "flight plots" will be used by each student to further analyze the different pitch motions in the lab report. Think of what tests you can do during these flights to enhance your judgement of the longitudinal motion quality.

4) If time permits, the best expected optimal design of the class will be implemented and tested. Be prepared to make convincing arguments for your best design.

5) Several other washouts will be available during the lab for testing, as time allows. This includes one with pitch motion that mimics the original GAT-1 pitch motion $\theta^s(t) = \frac{1}{2}\theta^a(t)$.

Following are four sections to help you understand the lab. First, a general description of the GAT-1 flight simulator; it has more than you need for the lab itself, but it is interesting to know. Second, there is a short description of the Pitch Optimal Washout System being used. Third, a description of what happens during take-off, what is strange about it, and what the problem is in simulating "correct" motion for it. Extrapolating from that, what other maneuvers do you think have interesting trade-offs in their simulator motion design? Fourth, there are some notes on using the optimal pitch washout design program.

1. The LINK GAT-1 Flight Simulator

In order to test the concept of the optimal washout, we intend to use the Link GAT-1 flight simulator. This is a general aviation simulator that resembles a Cessna 150 light aircraft and has three degrees of rotational freedom: yaw, roll, and pitch. It does not have visual out-the-window display capability. The GAT-1 has a very clever design which takes advantage of the simple washout it uses in order to simplify its circuits. The original Link washout design has the following descriptive characteristics:

a. The motion base pitch angle, $\theta^s(t)$ is 1/2 of the computed airplane pitch angle, $\theta^a(t)$, and is limited to +16 degrees up and -8 degrees down.

b. The roll angle is 1/6 of the computed airplane roll angle and is limited to ± 12.5 degrees off the erect position.

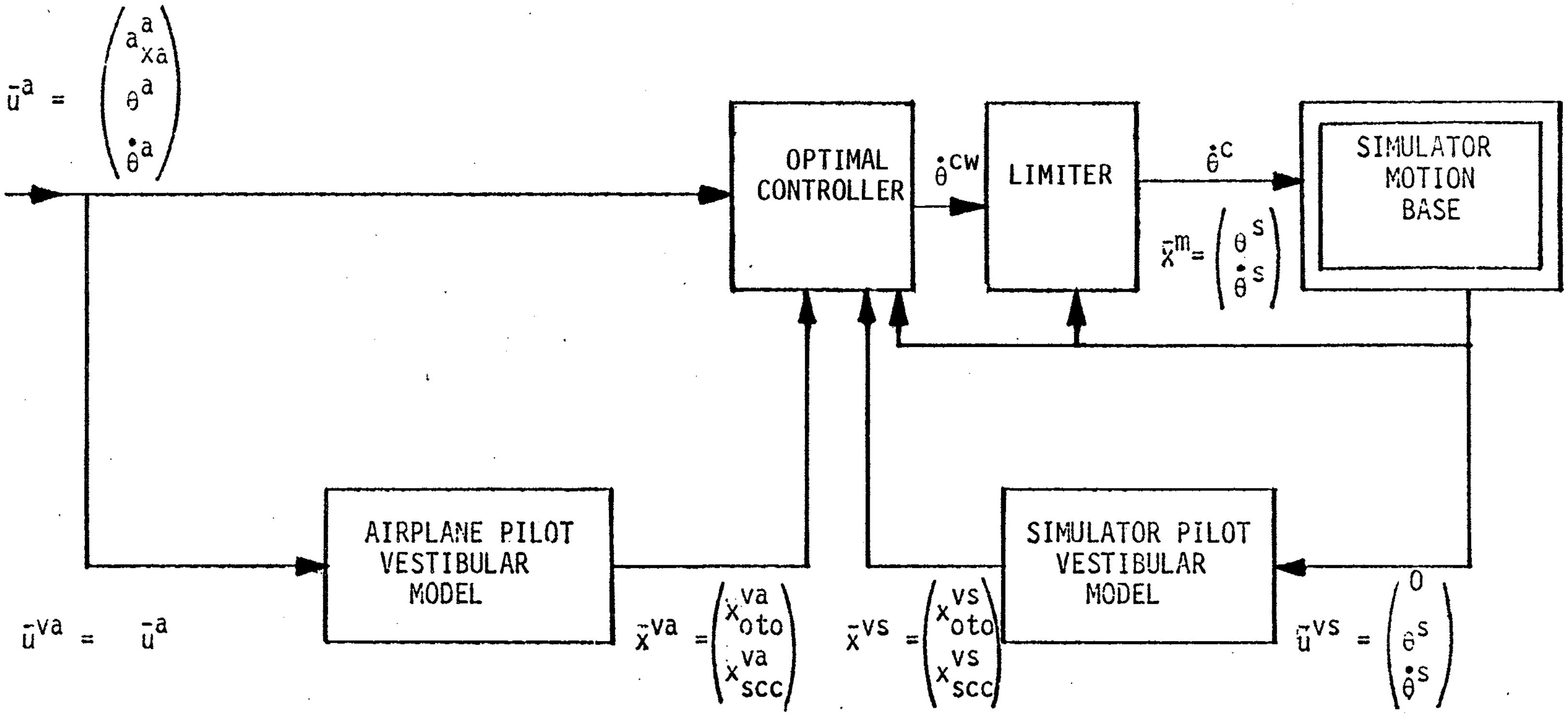
c. The yaw angle is not limited, but its rate is limited to 30 deg/sec.

d. The GAT-1 simulator has no linear motion capabilities. Furthermore, the computed linear motion of the airplane has no influence on the simulator motion.

The original GAT-1 design takes advantage of the simple washout by using the motion base itself as the last step of the integration of the airplane equations of motion. This is done by giving an angular velocity control input to the motion base. Thus the motion base is in a closed loop, since its angles are fed back into the airplane computations. This has the advantage of improving the motion quality as usually considered for feedback systems, but does not enable one to fly the simulator with the motion off. Also, all the flight instruments in the cab are properly scaled measurements of the motion base angles. This makes it very difficult to modify the current GAT-1 washout and thus only the simplest axis was adapted for use with the optimal washout. A point to note is that the pitch and roll rotation axes are below the center of mass of the simulator cab, which makes them open loop unstable. This also implies that the motion base integration of angular rate commands to obtain the angles is only approximate.

2. The Pitch Optimal Washout System

A general block diagram of the pitch optimal washout system is given in Figure 1. This is a closed loop washout system which has the advantage of being relatively simple to implement. The implementation basically requires the implementation of two vestibular models, one for the reference airplane pilot and one for the simulator pilot; beyond that the washout itself is merely a summation of all the states of the system using the pre-computed optimal gains. The vestibular models employed in the control system are the following:


$\bar{u}^a = \begin{pmatrix} a^a_{xa} \\ \theta^a \\ \dot{\theta}^a \end{pmatrix}$
OPTIMAL CONTROLLER
$\dot{\theta}^{cw}$
LIMITER
$\dot{\theta}^c$
SIMULATOR MOTION BASE
$\bar{x}^m = \begin{pmatrix} \theta^s \\ \dot{\theta}^s \end{pmatrix}$
AIRPLANE PILOT VESTIBULAR MODEL
SIMULATOR PILOT VESTIBULAR MODEL
$\bar{u}^{va} = \bar{u}^a$
$\bar{x}^{va} = \begin{pmatrix} x^{va}_{oto} \\ x^{va}_{scc} \end{pmatrix}$
$\bar{x}^{vs} = \begin{pmatrix} x^{vs}_{oto} \\ x^{vs}_{scc} \end{pmatrix}$
$\bar{u}^{vs} = \begin{pmatrix} 0 \\ \theta^s \\ \dot{\theta}^s \end{pmatrix}$

OPTIMAL CONTROLLER: $\dot{\theta}^{cw}(t) = -\bar{f}^n\bar{u}^a(t) - \bar{f}^{va}\bar{x}^{va}(t) - \bar{f}^{vs}\bar{x}^{vs}(t) - \bar{f}^m\bar{x}^m(t)$

KEY: $\bar{u}^a$ airplane motions

- a^a_{xa} - longitudinal linear acceleration (in airplane body axes)
- θ^a - Euler pitch angle
- $\dot{\theta}^a$ - Euler pitch rate

$\bar{u}^{va}$ airplane pilot vestibular input

$\bar{u}^{vs}$ limited simulator pitch axis command

$\dot{\theta}^c$ simulator pitch axis command

$\dot{\theta}^{cw}$ optimal controller pitch axis command

$\bar{x}^{va}$ airplane pilot vestibular model states

- x^{va}_{oto} - of otolith
- x^{va}_{scc} - of semicircular canal

$\bar{x}^{vs}$ simulator pilot vestibular model states (components similar to those of $\bar{x}^{va}$)

$\bar{x}^m$ motion base states

- θ^s - Euler pitch angle
- $\dot{\theta}^s$ - Euler pitch rate

Figure 1: Generalized block diagram of pitch-surge GAT-1 optimal washout

Transfer Function Form | State Space Form

Otolith:

$$y^o(s) = G_o \frac{s + b_o}{s + a_o} u^o(s) \tag{1}$$

u^o = linear acceleration input
y^o = otolith output

$$\dot{x}_{oto}(t) = -a_o x_{oto}(t) - (a_o - b_o) u^o(t) \tag{2}$$

$$y^o(t) = G_o x_{oto}(t) + G_o u^o(t) \tag{3}$$

Semicircular canal:

$$y^s(s) = G_s \frac{s}{s + a_s} u^s(s) \tag{4}$$

u^s = angular velocity input
y^s = semicircular canal output

$$\dot{x}_{scc}(t) = -a_s x_{scc}(t) - a_s u^s(t) \tag{5}$$

$$y^s(t) = G_s x_{scc}(t) + G_s u^s(t) \tag{6}$$

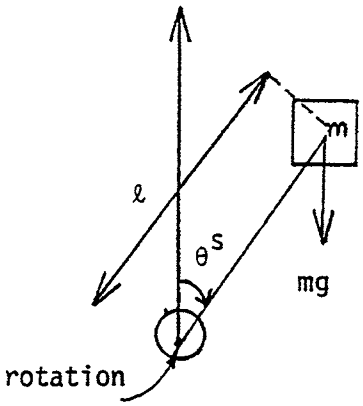
The values used are:

G_o = 21.168 g^{-1}	G_s = 40. sec/rad
a_o = 0.19 rad/sec	a_s = 0.169 rad/sec
b_o = 0.076 rad/sec	g = acceleration of gravity = 9.8 m/s^2

The outputs of these vestibular models are in terms of threshold units, meaning that an output of one unit corresponds to the minimum input the pilot can perceive under the expected pilot workload.

The closed loop washout system design requires a model for the GAT-1 motion base pitch axis. A second order inverted pendulum model with experimentally fitted parameters was used (Figure 2).

The angle limiter used is described in Figure 3. The general idea is that the circuit acts like a short as long as the simulator is within the motion base angle limits, but when commanded to go beyond, it just stays at the limits using an alternative closed loop control system with command $\theta^{c\ell}$.



$$\text{TORQUE} = I_{yy}\ddot{\theta}^s - \ell mg\theta^s + B\dot{\theta}^s$$

$$\therefore \quad \ddot{\theta}^s = (-(B+k_m)/I_{yy})\dot{\theta}^s + (\ell mg/I_{yy})\theta^s + (k_m/I_{yy})(\dot{\theta}^c/k_\Omega)$$

Parameters used for the pitch optimal washout design (the capital letter parameters are the symbols used in the computer program).

$$BR \triangleq \frac{B + k_m}{I_{yy}} = 3.13 \text{ [1/sec]}$$

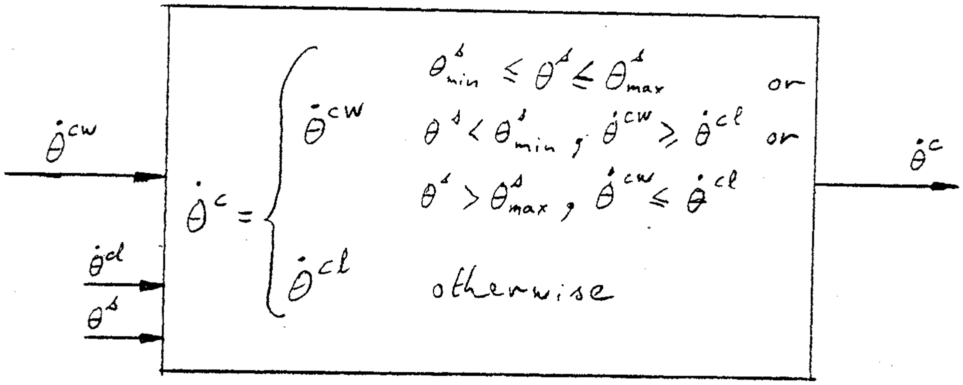
$$BBR \triangleq \frac{k_m}{I_{yy}k_\Omega} = 0.167 \text{ [rad/(sec}^2\text{volt)]}$$

$$PIP \triangleq \frac{\ell mg}{I_{yy}} = 0.74 \text{ [1/sec}^2\text{]}$$

Symbol key:

$\dot{\theta}^c$ = simulator control voltage (volts)
k_m = dc motor torque constant (Newton-m-sec)
k_Ω = dc motor back EMF constant (volts-sec/rad)
B = viscous forces (Newton-m-sec)
I_{yy} = inertia around pivot (kg-m^2)
mg = simulator weight (Newtons)
ℓ = length from pivot to center of gravity of m

Figure 2. Inverted pendulum model of the pitch axis of the GAT-1 flight simulator


$\dot{\theta}^{cw}$
$\dot{\theta}^{cl}$
θ^{s}
$\dot{\theta}^{c}$
$\dot{\theta}^{c} = \begin{cases} \dot{\theta}^{cw} & \theta^{s}_{min} \leq \theta^{s} \leq \theta^{s}_{max} \text{ or} \\ & \theta^{s} < \theta^{s}_{min},\ \dot{\theta}^{cw} \geq \dot{\theta}^{cl} \text{ or} \\ & \theta^{s} > \theta^{s}_{max},\ \dot{\theta}^{cw} \leq \dot{\theta}^{cl} \\ \dot{\theta}^{cl} & \text{otherwise} \end{cases}$

$$\dot{\theta}^{cl} = -f_1 \theta^{s} - f_2 \dot{\theta}^{s} + k\,\theta^{l}(\theta^{s})$$

where

$$\theta^{l}(\theta^{s}) = \begin{cases} \theta^{s}_{max} & \theta^{s} \geq 0 \\ \theta^{s}_{min} & \theta^{s} < 0 \end{cases}$$

Input: $\dot{\theta}^{cw}$ - optimal pitch rate command

$\dot{\theta}^{cl}$ - limitter loop pitch rate command

θ^{s} - motion base pitch angle

Constants: θ^{s}_{max} - maximum allowable motion base pitch angle

θ^{s}_{min} - minimum allowable motion base pitch angle

$\theta^{s}_{max} = 16$ degrees, $\theta^{s}_{min} = -8$ degrees

Figure 3: Block diagram of the pitch angle limiter.

3. Take-off Maneuver

Figure 4 shows an inflight recording of the fore-aft (surge axis) linear acceleration of a single engine Cessna 172 during take-off. The dotted line represents an estimate of the pitch angle. An interesting part of the experiment is to focus on the pitch up during take-off in the time window t = 20 to t = 50 seconds. The interesting point about the pitch up (t = 29 to 33 seconds) is that the pitch up information sensed by the semicircular canal and the otoliths do not correspond to the usual pitch up sensations; the pitch angle increases from approximately 0 to 4.5 degrees (semicircular canal cue) while the linear acceleration sensed in the fore-aft direction decreases from approximately 0.2 g to 0.13 g (otolith cue). The linear acceleration usually increases from 0 to 0.08 g for the given pitch angle. This unusual set of vestibular cues gives the washout design (having only pitch rotation) a difficult decision to make: Give the correct otolith cue by pitching down from, say, 10 degrees to 6.5 degrees -- dashed line in Figure 4 (assuming that we use residual tilt to simulate the linear acceleration at the time before t = 29 seconds) or give the correct pitch rate cue by further pitching up from 10 to 14.5 degrees -- dash-dot line in Figure 4. Since we are constrained in only having pitch rotation and no available linear fore-aft motion, the design of the washout has to make an explicit trade-off between the otolith and the semicircular canal cues. It will be interesting to discover what this trade-off is in terms of the design of the motion washout. Recordings of a take-off in the GAT-1 flight simulator with optimal washouts #0 and #2 are shown in Figure 5 with design parameters given in Table 1. Notice the residual tilt on acceleration and the favoring of semicircular canal cues on take-off.

4. Using the Optimal Washout Design Program

This program runs on the IPS IBM 370/168. Terminals for use with IPS are located on the second floor of Building 39. The dial up number for 300 baud is: x8-7511, which uses half duplex (HDX button on a DEC II printing terminal should be depressed). After dialing up the computer, the example anotated terminal session can be used to guide you through. The computer will automatically get into the design program after printing about half a page of garbage. Be patient!! Do not hit <CR> more than necessary. In case of an input error use @ and the return key (<CR> in the example) to delete the entire line with the error and then reenter the line.

Note that parameter numbers are entered as integers (no decimal point) while parameter values have to be entered as floating point numbers (must have a decimal point). At the end of your design and simulation, exit the program by entering -1 and then LOGOUT of the computer. Your output will appear about 15 minutes later (depending on the load) in the bin marked class 001, in the large metal file cabinets.

Each student has a budget of $40. An extra $20 can be granted to interested students. A typical design costs (after 5 PM) about $3.50 ($1.50 for computer time and $2.00 for printing), thus use your budget with thought, it is very limited.

There will be a demonstration of running the design program during the first lab.

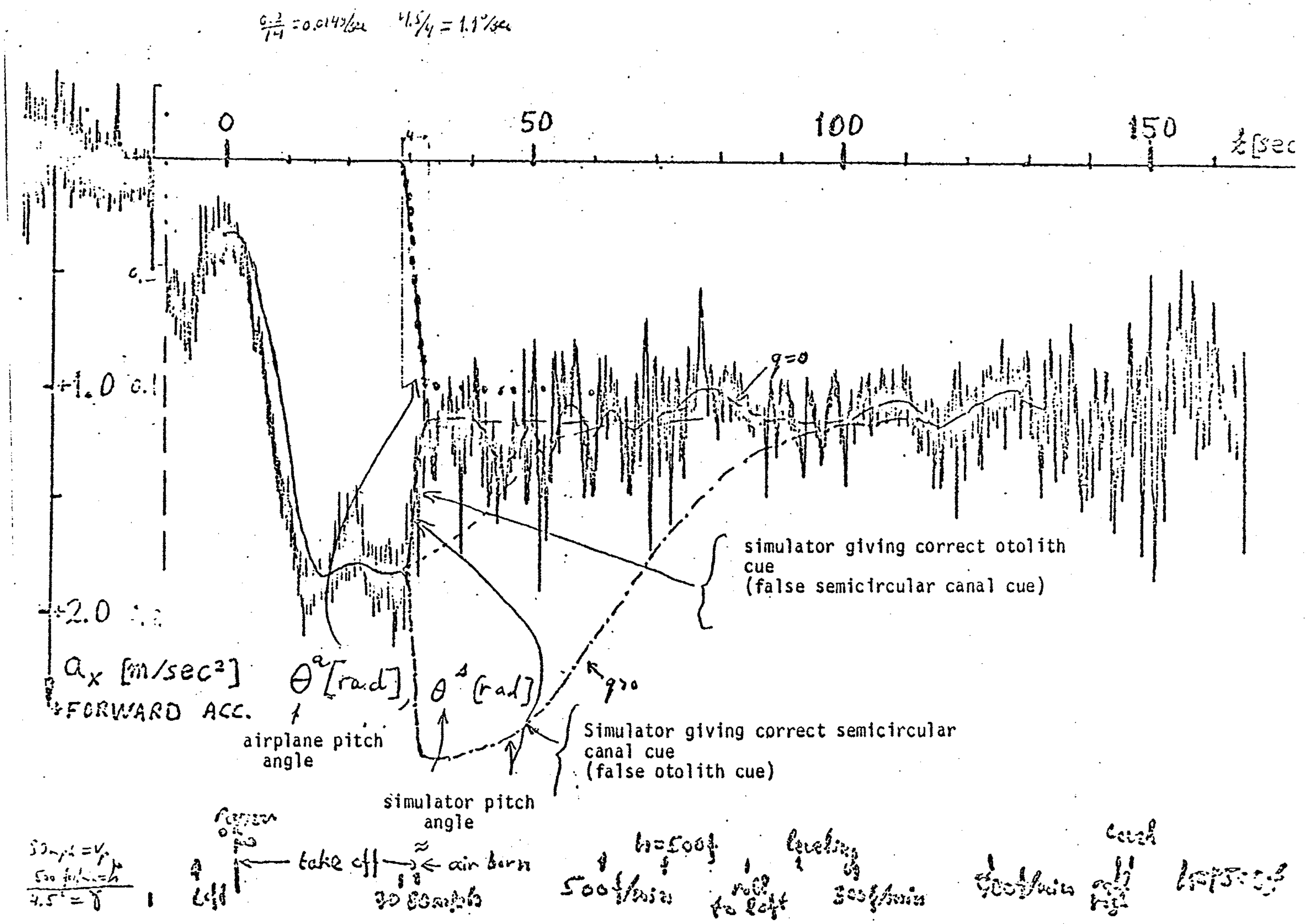


Figure 4: Fore-aft linear acceleration recording during take-off and expected simulator motions

θ^a - pitch angle of simulated airplane

θ^a_{xa} - surge linear acceleration (fore-aft) of simulated airplane in body axes.

θ^s - pitch angle of the simulator motion base

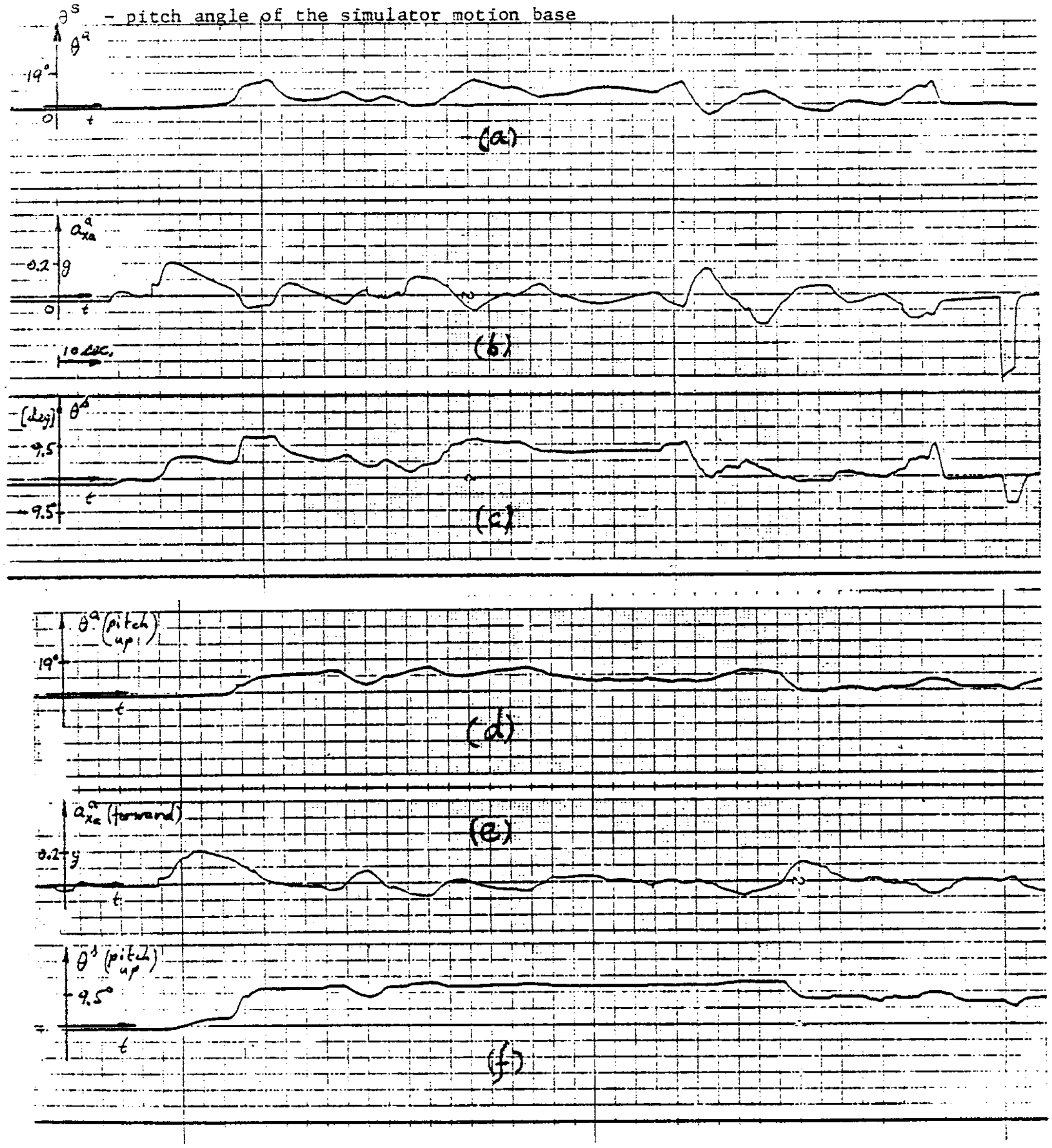


Figure 5. Take-off in the GAT-1 simulator using two different optimal washouts flown by the same pilot.
Traces (a), (b), and (c) are for washout #0 and traces (d), (e) and (f) for washout #2.

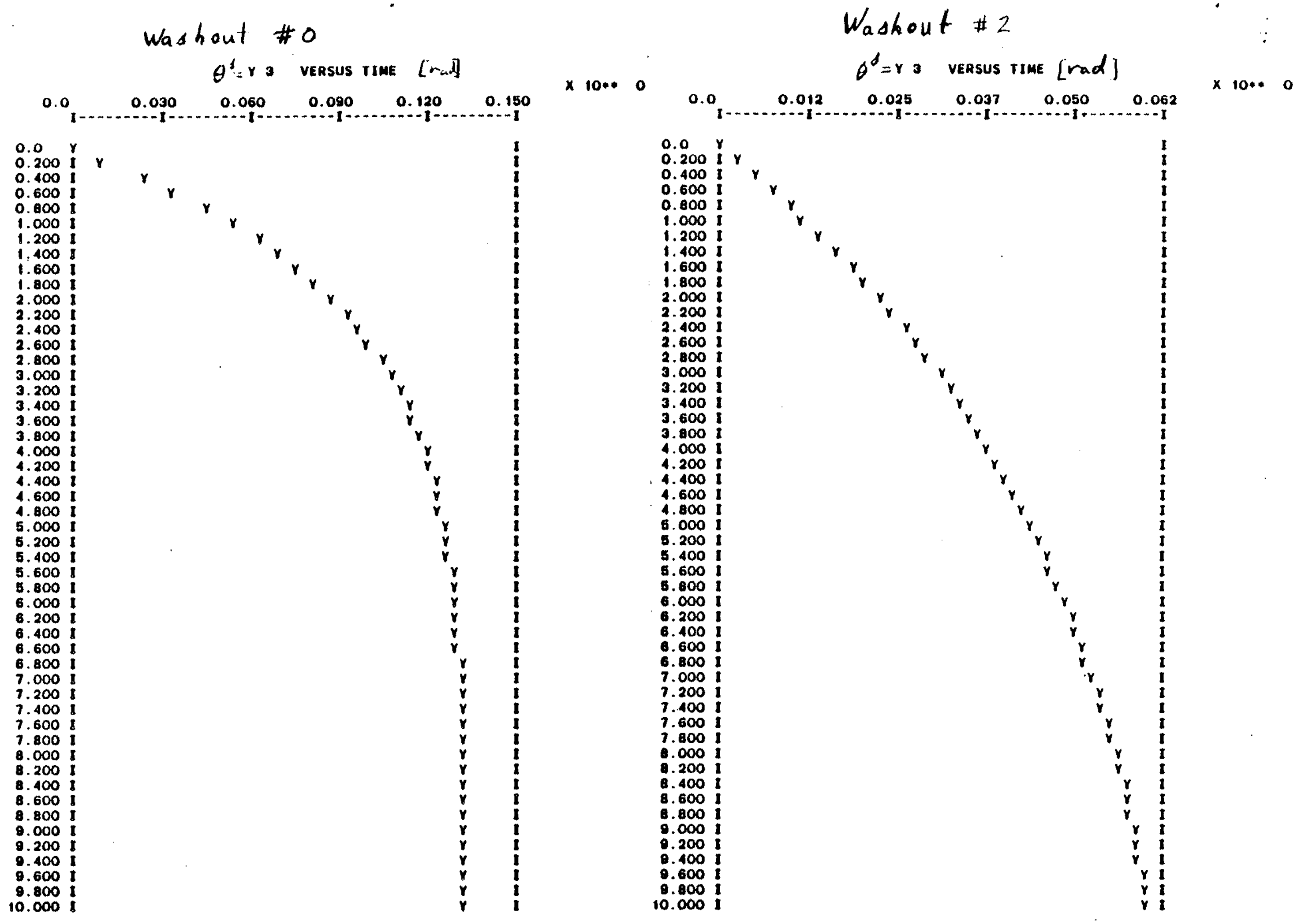
Washout #0

```
WAHOUT FILTER DESIGN PARAMETERS (11/16/81 18*09*17.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.100-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q. PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.O      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.O      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
*  20 RHO=     0.10D+00 WEIGHT OF MOTION OVER ERROR
```

Washout #2

```
WAHOUT FILTER DESIGN PARAMETERS (11/16/81 18*17*03.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.168 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>O
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>O
   16 PQ= 0.0          PQ=O POS. DEFINATE Q. PQ=1 NOT AT ALL
*  17 QO/QS=  0.10D+00 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
   18 RRO/R=  0.O      ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.O      VELOCITY/VELOCITY COMMAND
 GLOBEL SCALING
*  20 RHO=     0.10D+00 WEIGHT OF MOTION OVER ERROR
```

Table 1: Washouts #0 and #2 design parameters.



Simulation for a step 0.2g acc. input in a_{xa}^{a}, with washouts #0 and #2

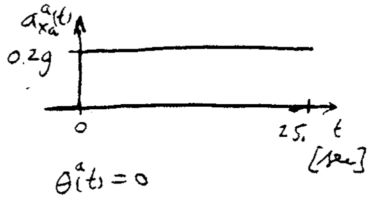
Call X8-7511 for 300 baud and HDX (half duplex)
press data button when you here the beep

```
VM/370 ONLINE
!

.login jeh
ENTER PASSWORD:
.[redacted]
LOGMSG - 21:04:51 EST THURSDAY 03/04/82
* NOTICES ON CALCOMP PLOTTER REMOVAL, XEROX 9700 FEATURES,
*   SERIES/1 CRT SURVEY, NEW MATRIX PACKAGE--ENTER: QUERY LOGMSG.
LOGON AT 16:28:56 EST SUNDAY 03/07/82
CMS/SP V1.8104 02/08/82
.
Y (19E) R/O
D (192) R/O
CP LINK JEHUDA 192 193 READ
ACCESS 193 C
C (193) R/O
CP LINK POTLUCK 200 200 RR
ACCESS 200 Z
Z (200) R/O
EXEC TDISK 10 E 194
DASD 194 DEFINED
DASD 194 - 10 CYL - ACCESSED AS E DISK
GLOBAL TXTLIB LIDS LAUB EISPACK FORTLIB FORTMOD2
CP SPOOL PRT CLASS H CONT
EXEC MAIL
NO MESSAGES IN READER.
EXEC WP
FILEDEF 7 TERM
FILEDEF 5 TERM
FILEDEF 6 DISK O DATA E ( RECFM F LRECL 132 BLKSIZE 132
FILEDEF 8 DISK S DATA E ( RECFM F LRECL 132 BLKSIZE 132
LOAD WP ( NOMAP START
EXECUTION BEGINS...
1 GAT-1 LINK PITCH AXIS WASHOUT DESIGN PROGRAM

     WRITTEN BY JEHUDA ISH-SHALOM (FEB 1982)

 COST FUNCTION PRAMETERS
  ERROR
    16 PQ= 0.0           PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
    17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
  ROTATION MOTIONS
    18 RR0/R=  0.0       ANGLE/VELOCITY COMMAND
    19 RR1/R=  0.0       VELOCITY/VELOCITY COMMAND
  GLOBAL SCALING
    20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR

 CHANGE PAR # (-1 EXIT; 0 CONTINUE; 80 PRINT;) :
?
.18   <-- note integer value!!!
 PAR(18)=      0.0            NEW VALUE :
.50.  <-- note decimal point, cannot do without it!!
 CHANGE PAR # (-1 EXIT; 0 CONTINUE; 80 PRINT;) :  <-- design parameters querry
?
.0  <-- continue and do design
 OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
 (THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
    1  GO=    21.168 (1/G)
    2  BO=     0.076 (RAD/SEC)
    3  AO=     0.190 (RAD/SEC)
```

Note
In case of error, delet the line by typing @<CR>, and reenter.

<-- be patient !!, wait till the computer finishes

```
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18 RR0/R=  0.50D+02 ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR

1           FEEDBACK GAINS

    FA

-0.5658993119416D+02   -0.2561924280317D+02

    FS

 0.5658993119415D+02    0.2561924280321D+02

    FM

 0.8875382240134D+02    0.1534389887258D+03

    FN L

-0.5519126471792D+02

    FN R

 0.5911603519738D+02   -0.8231072945845D+02

          DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:COM,ANGLE,D_ANGLE $


          COL    1           2

    1      -1.8683D+00 -2.2136D+00 -4.9874D+00
    2       4.2164D-01  4.9956D-01  1.1255D+00
    3       0.0         0.0         0.0
 SIMULATION CONDITIONS   ← step a^a_xa of 0.2g   a^a_xa(t) = { 0 t<0 ; 0.2g t≥0
  1   XZL=    0.200 |G S|
  2   SLL=    0.0    |G S/SEC|
  3   PLMAX= 0.0   |G S|
  4   XZR=    0.0     |RAD|  ← step θ^a=0.
  5   SLR=    0.0     |RAD/SEC|  ← step θ̇^a(t), ramp in θ^a(t)
  6   RMAX=   0.0     |RAD|
  7   DT=      0.200 [SEC]
  8   T=     10.000 [SEC]  ← length of simulation

 (-1 EXIT;0 CONTINUE;22 NEW SIM;55 NEW DESIGN;80 PRINT;90,91,92 TEST)
 CHANGE PARAMETER # :  ← simulation parameters querry.
?
.8
 ZIN( 8)=.1000000000000000D+02 NEW VALUE:
.25.
 (-1 EXIT;0 CONTINUE;22 NEW SIM;55 NEW DESIGN;80 PRINT;90,91,92 TEST)
 CHANGE PARAMETER # :
?
.80  ← check parameters of simulation


 SIMULATION CONDITIONS
  1   XZL=    0.200 |G S|
  2   SLL=    0.0    |G S/SEC|
  3   PLMAX= 0.0   |G S|
  4   XZR=    0.0     |RAD|
  5   SLR=    0.0     |RAD/SEC|
  6   RMAX=   0.0     |RAD|
  7   DT=      0.500 [SEC]
  8   T=     25.000 [SEC]

 (-1 EXIT;0 CONTINUE;22 NEW SIM;55 NEW DESIGN;80 PRINT;90,91,92 TEST)
 CHANGE PARAMETER # :
?
.0
```



```
 SIMULATION CONDITIONS
  1   XZL=    0.200 |G S|
  2   SLL=    0.0    |G S/SEC|
  3   PLMAX= 0.0   |G S|
  4   XZR=    0.0     |RAD|
  5   SLR=    0.0     |RAD/SEC|
  6   RMAX=   0.0     |RAD|
  7   DT=      0.500 [SEC]
  8   T=     25.000 [SEC]

                                          next simulation condition defult
 SIMULATION CONDITIONS
  1   XZL=    0.200 |G S|
  2   SLL=    0.0    |G S/SEC|
  3   PLMAX= 0.0   |G S|
  4   XZR=    0.0     |RAD|
  5   SLR=    0.0     |RAD/SEC|
  6   RMAX=   0.0     |RAD|
  7   DT=      0.500 [SEC]
  8   T=     25.000 [SEC]

 (-1 EXIT;0 CONTINUE;22 NEW SIM;55 NEW DESIGN;80 PRINT;90,91,92 TEST)
 CHANGE PARAMETER # :
?
.-1  ← exit program
PRINT S DATA E ( CC )
PRINT O DATA E ( CC )
CP SPOOL PRINT CLOSE
COST
COST AT 16:36:36 EST SUNDAY 03/07/82
--------------------------------------------------
TOTAL CPU TIME            5 SECONDS         $0.34
CONNECT TIME              7 MINUTES         $0.10
PAGE SWAPPING           228 PAGES           $0.00
NON-SPOOLED I/O        1139 SIOS            $0.27
SPOOLED I/O             842 RECORDS         $0.10
--------------------------------------------------
TOTAL CMS CHARGES      440K WEEKEND         $0.82
R; T=2.27/4.76 16:36:51

.logout
CONNECT= 00:08:05 VIRTCPU= 000:02.34 TOTCPU= 000:05.21
LOGOFF AT 16:37:03 EST SUNDAY 03/07/82
```

Sample Optimal Washout Design Program Output

```
WAHOUT FILTER DESIGN PARAMETERS ( 3/04/82  2*49*13.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=    21.168 (1/G)
   2  BO=     0.076 (RAD/SEC)
   3  AO=     0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=    40.000 (1/RAD)
   5  AS=     0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=    0.100 (RAD/SEC)
   7  BNR=    0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=     3.130 (1/SEC)
   10 BBR=    0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
 * 18 RRO/R=  0.50D+02 ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR
```

mark * for changing the default

```
      FEEDFORWARD RESISTORS

R1=       25.630 KOHM

R2=       23.185 KOHM

R3=       18.119 KOHM

R4=       17.671 KOHM

R5=       70.841 KOHM

      FEEDBACK RESISTORS

R6=       34.143 KOHM

R7=       43.448 KOHM

R8=       65.937 KOHM

R9=      260.221 KOHM
```

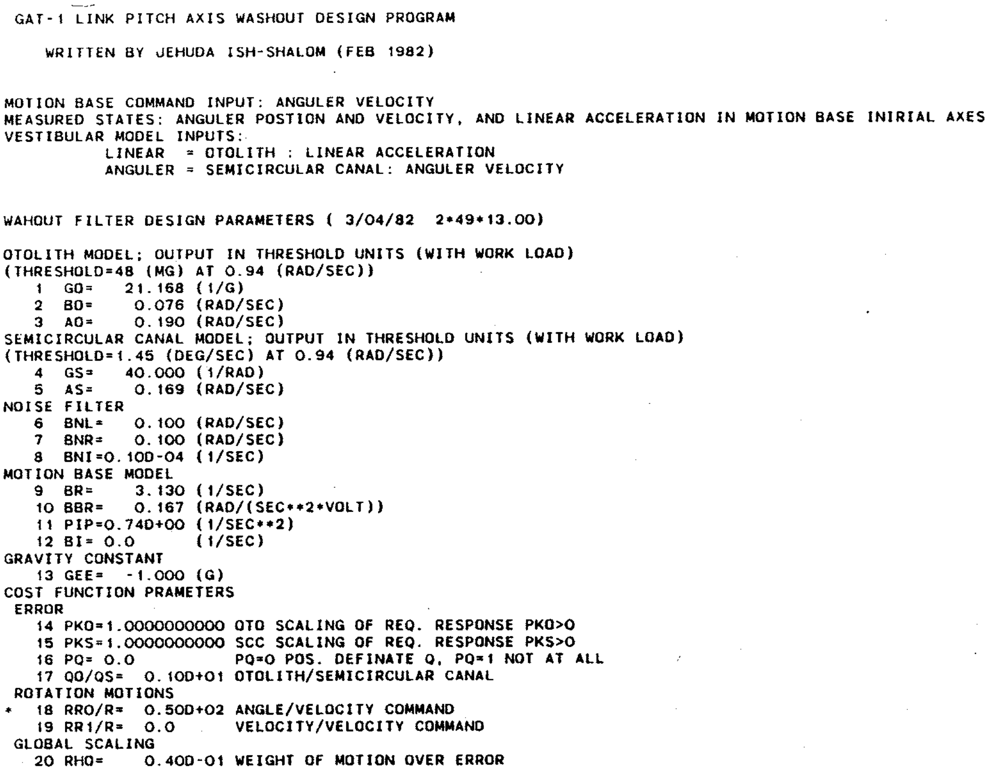
Resistor values used to implement the design.

```
SIMULATION CONDITIONS
 1  XZL=    0.200 |G S|
 2  SLL=    0.0   |G S/SEC|
 3  PLMAX= 0.0   |G S|
 4  XZR=    0.0   |RAD|
 5  SLR=    0.0   |RAD/SEC|
 6  RMAX=   0.0   |RAD|
 7  DT=     0.500 [SEC]
 8  T=     25.000 [SEC]
```

a^{a}_{xa}: 0.2 g step input for 25 sec's

Note:

The output of the Optimal Pitch Washout here is compressed. The page numbers apply to the real page numbers and order.



```
 GAT-1 LINK PITCH AXIS WASHOUT DESIGN PROGRAM

    WRITTEN BY JEHUDA ISH-SHALOM (FEB 1982)


MOTION BASE COMMAND INPUT: ANGULER VELOCITY
MEASURED STATES: ANGULER POSTION AND VELOCITY, AND LINEAR ACCELERATION IN MOTION BASE INIRIAL AXES
VESTIBULAR MODEL INPUTS:
          LINEAR  = OTOLITH : LINEAR ACCELERATION
          ANGULER = SEMICIRCULAR CANAL: ANGULER VELOCITY


WAHOUT FILTER DESIGN PARAMETERS ( 3/04/82  2*49*13.00)

OTOLITH MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=48 (MG) AT 0.94 (RAD/SEC))
   1  GO=   21.168 (1/G)
   2  BO=    0.076 (RAD/SEC)
   3  AO=    0.190 (RAD/SEC)
SEMICIRCULAR CANAL MODEL; OUTPUT IN THRESHOLD UNITS (WITH WORK LOAD)
(THRESHOLD=1.45 (DEG/SEC) AT 0.94 (RAD/SEC))
   4  GS=   40.000 (1/RAD)
   5  AS=    0.169 (RAD/SEC)
NOISE FILTER
   6  BNL=   0.100 (RAD/SEC)
   7  BNR=   0.100 (RAD/SEC)
   8  BNI=0.10D-04 (1/SEC)
MOTION BASE MODEL
   9  BR=    3.130 (1/SEC)
   10 BBR=   0.167 (RAD/(SEC**2*VOLT))
   11 PIP=0.74D+00 (1/SEC**2)
   12 BI= 0.0      (1/SEC)
GRAVITY CONSTANT
   13 GEE=  -1.000 (G)
COST FUNCTION PRAMETERS
 ERROR
   14 PKO=1.0000000000 OTO SCALING OF REQ. RESPONSE PKO>0
   15 PKS=1.0000000000 SCC SCALING OF REQ. RESPONSE PKS>0
   16 PQ= 0.0          PQ=0 POS. DEFINATE Q, PQ=1 NOT AT ALL
   17 QO/QS=  0.10D+01 OTOLITH/SEMICIRCULAR CANAL
 ROTATION MOTIONS
*  18 RRO/R=  0.50D+02 ANGLE/VELOCITY COMMAND
   19 RR1/R=  0.0      VELOCITY/VELOCITY COMMAND
 GLOBAL SCALING
   20 RHO=    0.40D-01 WEIGHT OF MOTION OVER ERROR
```

```
        OPEN LOOP EIGENVALUES

notice the motion
base is open loop unstable--EIGENVALUES---
                 REAL PART      IMAG PART    NAT FREQ(HZ)      ZETA        FREQ(HZ)

motion     1    -3.351D+00      0.0           5.333D-01      1.000000     0.0
base
model      2     2.208D-01      0.0           3.515D-02     -1.000000     0.0
oto        3  { -1.900D-01      0.0           3.024D-02      1.000000     0.0
model      4  { -1.900D-01      0.0           3.024D-02      1.000000     0.0
scc        5  { -1.695D-01      0.0           2.698D-02      1.000000     0.0
models     6  { -1.695D-01      0.0           2.698D-02      1.000000     0.0
airplane   7  { -1.000D-01      0.0           1.592D-02      1.000000     0.0
motion     8  { -1.000D-01      0.0           1.592D-02      1.000000     0.0
models     9  { -1.000D-05      0.0           1.592D-06      1.000000     0.0

                                 CLOSED-LOOP EIGENVALUES
                 REAL PART      IMAG PART    NAT FREQ(HZ)      ZETA        FREQ(HZ)

           1    -2.828D+01      0.0           4.501D+00      1.000000     0.0
main pole
of interest 2 ⇒ -5.648D-01      0.0           8.989D-02      1.000000     0.0
same as    3  { -1.900D-01      0.0           3.024D-02      1.000000     0.0
open loop  4  { -1.695D-01      0.0           2.698D-02      1.000000     0.0
of actual pilot
           5    -1.661D-01      0.0           2.644D-02      1.000000     0.0
           6    -1.021D-01      0.0           1.625D-02      1.000000     0.0
same as    7  { -1.000D-01      0.0           1.592D-02      1.000000     0.0
open loop
airplane   8  { -1.000D-01      0.0           1.592D-02      1.000000     0.0
motion     9  { -1.000D-05      0.0           1.592D-06      1.000000     0.0
models
        OPTIMAL CLOSED LOOP MATRIX ACL


                 COL    1           2          3          4          5          6          7          8          9
        ROW
```

```
 1    -1.9000D-01  0.0          0.0          0.0          0.0          0.0          -1.1400D-01 -1.1400D-01  0.0
 2     0.0        -1.6949D-01   0.0          0.0          0.0          0.0           0.0         0.0        -1.6949D-01
 3     0.0         0.0         -1.9000D-01   0.0         -1.1400D-01   0.0           0.0         0.0         0.0
 4     0.0         0.0          0.0         -1.6949D-01   0.0         -1.6949D-01    0.0         0.0         0.0
 5     0.0         0.0          0.0          0.0          0.0          1.0000D+00    0.0         0.0         0.0
 6     9.4505D+00  4.2784D+00  -9.4505D+00  -4.2784D+00  -1.4082D+01  -2.8754D+01    9.2169D+00  9.8724D+00  1.3746D+01
 7     0.0         0.0          0.0          0.0          0.0          0.0          -1.0000D-01  0.0         0.0
 8     0.0         0.0          0.0          0.0          0.0          0.0           0.0        -1.0000D-05  1.0000D+00
 9     0.0         0.0          0.0          0.0          0.0          0.0           0.0         0.0        -1.0000D-01

OPTIMAL GAIN MATRIX G


        COL  1          2            3            4            5            6            7            8            9
ROW
 1       -5.6590D+01 -2.5619D+01  5.6590D+01  2.5619D+01  8.8754D+01  1.5344D+02 -5.5191D+01 -5.9116D+01 -8.2311D+01
```

Well, you cannot always stop the printing of garbage.

```
          FEEDBACK GAINS

     FA

-0.56589931194160+02    -0.25619242803170+02

     FS

 0.56589931194150+02     0.25619242803210+02

     FM

 0.88753822401340+02     0.15343898872580+03

     FN L

-0.55191264717920+02

     FN R

-0.59116035197380+02    -0.82310729458450+02


      FEEDFORWARD RESISTORS

R1=       25.630 KOHM

R2=       23.185 KOHM

R3=       18.119 KOHM

R4=       17.671 KOHM

R5=       70.841 KOHM

      FEEDBACK RESISTORS

R6=       34.143 KOHM

R7=       43.448 KOHM

R8=       65.937 KOHM

R9=      260.221 KOHM


        DC GAINS COL:ACC,ANGLE,D_ANGLE; ROW:COM,ANGLE,D_ANGLE
```

INPUTS

OUTPUTS a^{a}_{xa} [g] θ^{a} [rad] $\dot{\theta}^{a}$ [rad]

```
            COL     1            2            3

     ROW

       1       -1.8683D+00 -2.2136D+00 -4.9874D+00

       2        4.2164D-01  4.9956D-01  1.1255D+00

       3        0.0         0.0         0.0
```

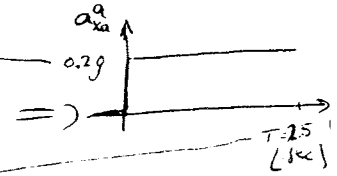
$\dot{\theta}^{cw}$ 1

θ^{s} 2

$\dot{\theta}^{s}$ 3

these are the important numbers

```
SIMULATION CONDITIONS
 1   XZL=    0.200  |G S|
 2   SLL=    0.0    |G S/SEC|
 3   PLMAX= 0.0    |G S|
 4   XZR=    0.0    |RAD|
 5   SLR=    0.0    |RAD/SEC|
 6   RMAX=   0.0    |RAD|
 7   DT=     0.500 [SEC]
 8   T=     25.000 [SEC]
```



θ^{s} $\dot{\theta}^{s}$

```
ORDER OF OUTPUTS Y(I)
Y(1)=EOTO,     Y(2)=ESCC,     Y(3)=TETAM,  Y(4)=D_TETAM
Y(5)=YAOTO,    Y(6)=YSOTO,    Y(7)=YASCC,  Y(8)=YSSCC
```

$u = \dot{\theta}^{cw}$

```
            COL     1
     ROW

       1          0.0

       2          0.0

       3          0.0

       4          0.0

       5          0.0

       6          0.0

       7          2.0000D-01

       8          0.0

       9          0.0
```

initial state

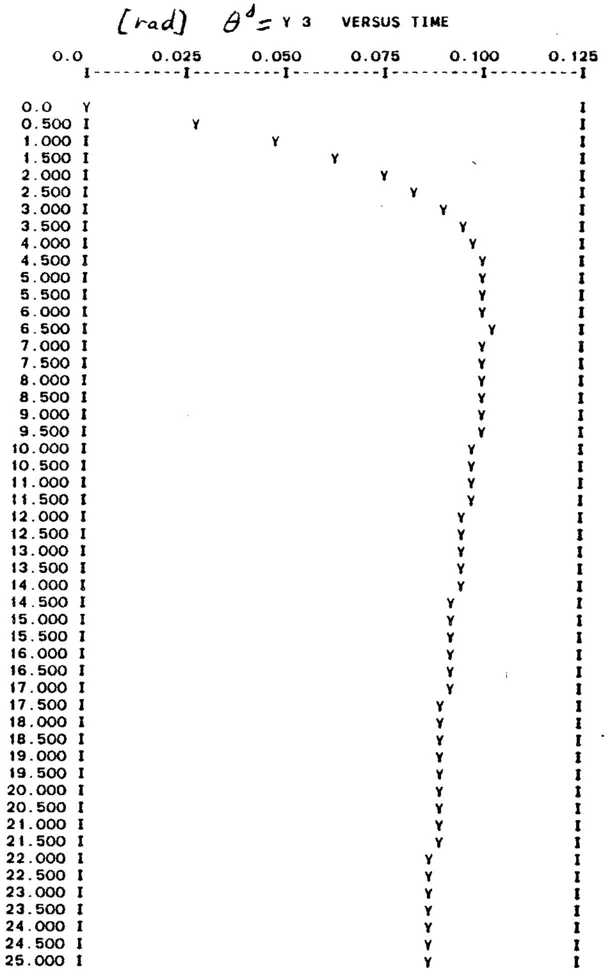
```
DEGREE OF PADE APPROXIMANT          =     7
TOLERANCE ACHIEVED                  =     0.21854268716410880-1
SPECIFIED CONVERGENCE TOLERANCE     =     0.1000000000000000D-1
```

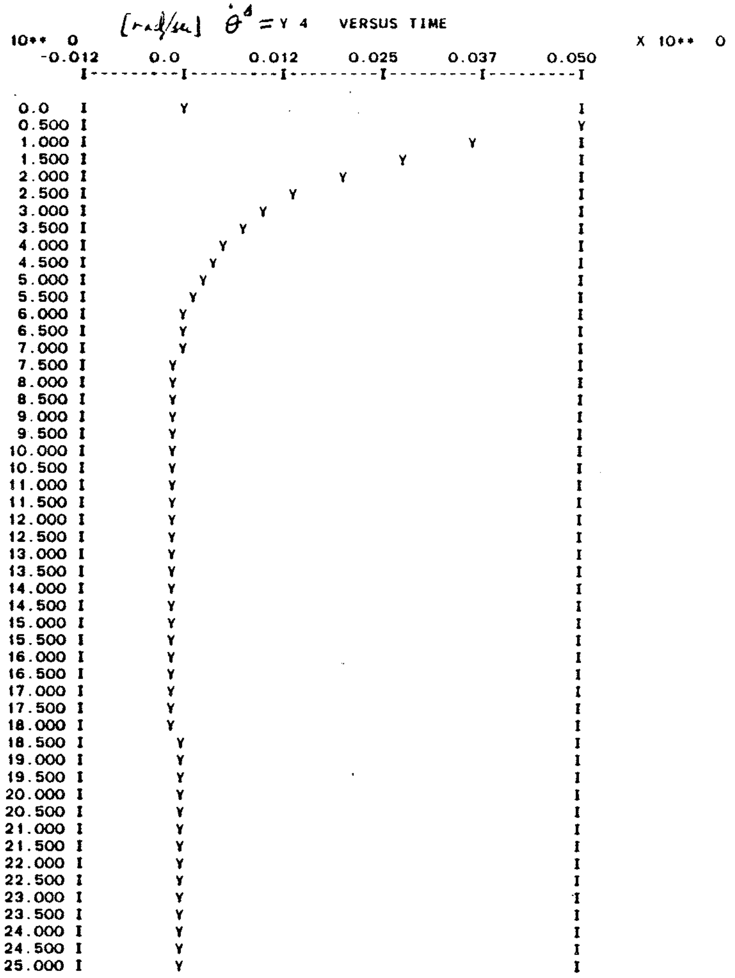
(th] ℓ_{oto} = Y 1 VERSUS TIME

```
     0.800     1.600     2.400     3.200     4.000     4.800   X 10**  0
       I---------I---------I---------I---------I---------I
   0.0 I                                          Y      I
 0.500 I                                Y                I
 1.000 I                        Y                        I
 1.500 I                   Y                             I
 2.000 I              Y                                  I
 2.500 I           Y                                     I
 3.000 I         Y                                       I
 3.500 I       Y                                         I
 4.000 I     Y                                           I
 4.500 I    Y                                            I
 5.000 I    Y                                            I
 5.500 I   Y                                             I
 6.000 I   Y                                             I
 6.500 I  Y                                              I
 7.000 I  Y                                              I
 7.500 I  Y                                              I
 8.000 I  Y                                              I
 8.500 I  Y                                              I
 9.000 I  Y                                              I
 9.500 I  Y                                              I
10.000 I  Y                                              I
10.500 I  Y                                              I
11.000 I  Y                                              I
11.500 I  Y                                              I
12.000 I  Y                                              I
12.500 I  Y                                              I
13.000 I  Y                                              I
13.500 I  Y                                              I
14.000 I  Y                                              I
14.500 I  Y                                              I
15.000 I  Y                                              I
15.500 I  Y                                              I
16.000 I  Y                                              I
16.500 I  Y                                              I
17.000 I  Y                                              I
17.500 I  Y                                              I
18.000 I  Y                                              I
18.500 I  Y                                              I
19.000 I  Y                                              I
19.500 I  Y                                              I
20.000 I Y                                               I
20.500 I Y                                               I
21.000 I Y                                               I
21.500 I Y                                               I
22.000 I Y                                               I
22.500 I Y                                               I
23.000 I Y                                               I
23.500 I Y                                               I
24.000 I Y                                               I
24.500 I Y                                               I
25.000 I Y                                               I
```

(th] e_{acc} = Y 2 VERSUS TIME

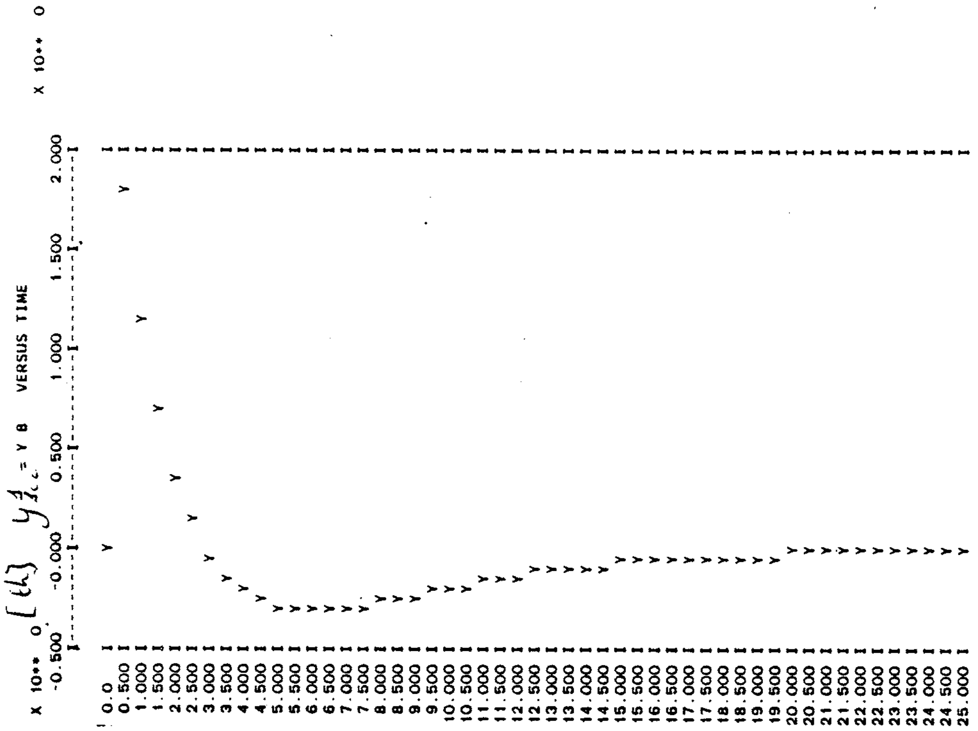
```
    -2.000    -1.500    -1.000    -0.500      0.0      0.500   X 10**  0
       I---------I---------I---------I---------I---------I
   0.0 I                                       Y         I
 0.500 I   Y                                             I
 1.000 I                Y                                I
 1.500 I                         Y                       I
 2.000 I                                Y                I
 2.500 I                                    Y            I
 3.000 I                                        Y        I
 3.500 I                                          Y      I
 4.000 I                                           Y     I
 4.500 I                                            Y    I
 5.000 I                                             Y   I
 5.500 I                                             Y   I
 6.000 I                                             Y   I
 6.500 I                                             Y   I
 7.000 I                                             Y   I
 7.500 I                                             Y   I
 8.000 I                                            Y    I
 8.500 I                                            Y    I
 9.000 I                                            Y    I
 9.500 I                                           Y     I
10.000 I                                           Y     I
10.500 I                                           Y     I
11.000 I                                          Y      I
11.500 I                                          Y      I
12.000 I                                          Y      I
12.500 I                                         Y       I
13.000 I                                         Y       I
13.500 I                                         Y       I
14.000 I                                         Y       I
14.500 I                                         Y       I
15.000 I                                        Y        I
15.500 I                                        Y        I
16.000 I                                        Y        I
16.500 I                                        Y        I
17.000 I                                        Y        I
17.500 I                                        Y        I
18.000 I                                        Y        I
18.500 I                                        Y        I
19.000 I                                        Y        I
19.500 I                                        Y        I
20.000 I                                       Y         I
20.500 I                                       Y         I
21.000 I                                       Y         I
21.500 I                                       Y         I
22.000 I                                       Y         I
22.500 I                                       Y         I
23.000 I                                       Y         I
23.500 I                                       Y         I
24.000 I                                       Y         I
24.500 I                                       Y         I
25.000 I                                       Y         I
```


[rad] θ^d = Y 3 VERSUS TIME
X 10** 0
0.0
0.025
0.050
0.075
0.100
0.125
0.0
0.500
1.000
1.500
2.000
2.500
3.000
3.500
4.000
4.500
5.000
5.500
6.000
6.500
7.000
7.500
8.000
8.500
9.000
9.500
10.000
10.500
11.000
11.500
12.000
12.500
13.000
13.500
14.000
14.500
15.000
15.500
16.000
16.500
17.000
17.500
18.000
18.500
19.000
19.500
20.000
20.500
21.000
21.500
22.000
22.500
23.000
23.500
24.000
24.500
25.000

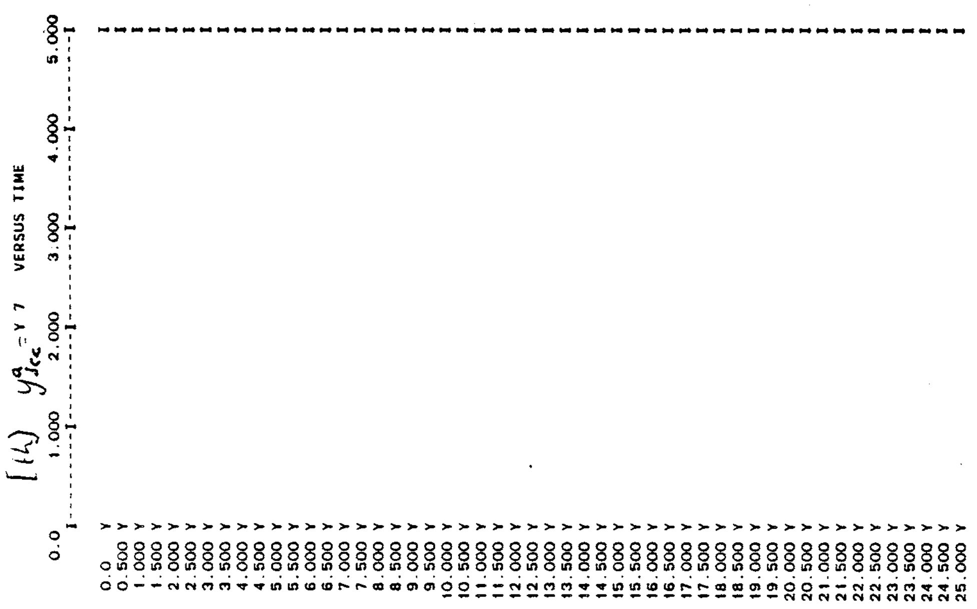

[rad/sec] $\dot{\theta}^d$ = Y 4 VERSUS TIME
X 10** 0
-0.012
0.0
0.012
0.025
0.037
0.050
0.0
0.500
1.000
1.500
2.000
2.500
3.000
3.500
4.000
4.500
5.000
5.500
6.000
6.500
7.000
7.500
8.000
8.500
9.000
9.500
10.000
10.500
11.000
11.500
12.000
12.500
13.000
13.500
14.000
14.500
15.000
15.500
16.000
16.500
17.000
17.500
18.000
18.500
19.000
19.500
20.000
20.500
21.000
21.500
22.000
22.500
23.000
23.500
24.000
24.500
25.000

```
          [th]  y^a_ob = Y 5   VERSUS TIME                       X 10**  0
      1.600     2.400     3.200     4.000     4.800     5.600
        I---------I---------I---------I---------I---------I
   0.0  I                                Y                I
 0.500  I                             Y                   I
 1.000  I                          Y                      I
 1.500  I                        Y                        I
 2.000  I                      Y                          I
 2.500  I                    Y                            I
 3.000  I                  Y                              I
 3.500  I                Y                                I
 4.000  I               Y                                 I
 4.500  I              Y                                  I
 5.000  I            Y                                    I
 5.500  I           Y                                     I
 6.000  I          Y                                      I
 6.500  I         Y                                       I
 7.000  I         Y                                       I
 7.500  I        Y                                        I
 8.000  I       Y                                         I
 8.500  I      Y                                          I
 9.000  I      Y                                          I
 9.500  I     Y                                           I
10.000  I     Y                                           I
10.500  I    Y                                            I
11.000  I    Y                                            I
11.500  I    Y                                            I
12.000  I   Y                                             I
12.500  I   Y                                             I
13.000  I   Y                                             I
13.500  I   Y                                             I
14.000  I  Y                                              I
14.500  I  Y                                              I
15.000  I  Y                                              I
15.500  I  Y                                              I
16.000  I  Y                                              I
16.500  I  Y                                              I
17.000  I Y                                               I
17.500  I Y                                               I
18.000  I Y                                               I
18.500  I Y                                               I
19.000  I Y                                               I
19.500  I Y                                               I
20.000  I Y                                               I
20.500  I Y                                               I
21.000  I Y                                               I
21.500  I Y                                               I
22.000  I Y                                               I
22.500  I Y                                               I
23.000  I Y                                               I
23.500  I Y                                               I
24.000  I Y                                               I
24.500  IY                                                I
25.000  IY                                                I
```

```
          [th]  y^d_cte = Y 6   VERSUS TIME                       X 10**  0
       0.0       0.400     0.800     1.200     1.600     2.000
        I---------I---------I---------I---------I---------I
   0.0  Y                                                 I
 0.500  I             Y                                   I
 1.000  I                       Y                         I
 1.500  I                              Y                  I
 2.000  I                                  Y              I
 2.500  I                                     Y           I
 3.000  I                                      Y          I
 3.500  I                                      Y          I
 4.000  I                                      Y          I
 4.500  I                                      Y          I
 5.000  I                                     Y           I
 5.500  I                                   Y             I
 6.000  I                                  Y              I
 6.500  I                                 Y               I
 7.000  I                                Y                I
 7.500  I                               Y                 I
 8.000  I                             Y                   I
 8.500  I                            Y                    I
 9.000  I                           Y                     I
 9.500  I                          Y                      I
10.000  I                         Y                       I
10.500  I                         Y                       I
11.000  I                        Y                        I
11.500  I                       Y                         I
12.000  I                       Y                         I
12.500  I                      Y                          I
13.000  I                      Y                          I
13.500  I                     Y                           I
14.000  I                     Y                           I
14.500  I                    Y                            I
15.000  I                    Y                            I
15.500  I                    Y                            I
16.000  I                   Y                             I
16.500  I                   Y                             I
17.000  I                   Y                             I
17.500  I                   Y                             I
18.000  I                  Y                              I
18.500  I                  Y                              I
19.000  I                  Y                              I
19.500  I                  Y                              I
20.000  I                  Y                              I
20.500  I                  Y                              I
21.000  I                  Y                              I
21.500  I                 Y                               I
22.000  I                 Y                               I
22.500  I                 Y                               I
23.000  I                 Y                               I
23.500  I                 Y                               I
24.000  I                 Y                               I
24.500  I                 Y                               I
25.000  I                 Y                               I
```



Γ(t) y^A_{Icc} = Y 8 VERSUS TIME
X 10** 0
-0.500
-0.000
0.500
1.000
1.500
2.000
X 10** 0
0.0
0.500
1.000
1.500
2.000
2.500
3.000
3.500
4.000
4.500
5.000
5.500
6.000
6.500
7.000
7.500
8.000
8.500
9.000
9.500
10.000
10.500
11.000
11.500
12.000
12.500
13.000
13.500
14.000
14.500
15.000
15.500
16.000
16.500
17.000
17.500
18.000
18.500
19.000
19.500
20.000
20.500
21.000
21.500
22.000
22.500
23.000
23.500
24.000
24.500
25.000



Γ(t) y^a_{Icc} = Y 7 VERSUS TIME
0.0
1.000
2.000
3.000
4.000
5.000
0.0
0.500
1.000
1.500
2.000
2.500
3.000
3.500
4.000
4.500
5.000
5.500
6.000
6.500
7.000
7.500
8.000
8.500
9.000
9.500
10.000
10.500
11.000
11.500
12.000
12.500
13.000
13.500
14.000
14.500
15.000
15.500
16.000
16.500
17.000
17.500
18.000
18.500
19.000
19.500
20.000
20.500
21.000
21.500
22.000
22.500
23.000
23.500
24.000
24.500
25.000

$[V_0(t)]$ $\dot{\theta}^{cw}$ = U 1 VERSUS TIME

```
                                                            X 10**  0
    -3.000    0.0       3.000     6.000     9.000    12.000
       I---------I---------I---------I---------I---------I

   0.0 I                                              U  I
 0.500 I           U                                     I
 1.000 I          U                                      I
 1.500 I         U                                       I
 2.000 I         U                                       I
 2.500 I        U                                        I
 3.000 I        U                                        I
 3.500 I        U                                        I
 4.000 I        U                                        I
 4.500 I        U                                        I
 5.000 I        U                                        I
 5.500 I        U                                        I
 6.000 I        U                                        I
 6.500 I       U                                         I
 7.000 I       U                                         I
 7.500 I       U                                         I
 8.000 I       U                                         I
 8.500 I       U                                         I
 9.000 I       U                                         I
 9.500 I       U                                         I
10.000 I       U                                         I
10.500 I       U                                         I
11.000 I       U                                         I
11.500 I        U                                        I
12.000 I        U                                        I
12.500 I        U                                        I
13.000 I        U                                        I
13.500 I        U                                        I
14.000 I        U                                        I
14.500 I        U                                        I
15.000 I        U                                        I
15.500 I        U                                        I
16.000 I        U                                        I
16.500 I        U                                        I
17.000 I        U                                        I
17.500 I        U                                        I
18.000 I        U                                        I
18.500 I        U                                        I
19.000 I        U                                        I
19.500 I        U                                        I
20.000 I        U                                        I
20.500 I        U                                        I
21.000 I        U                                        I
21.500 I        U                                        I
22.000 I        U                                        I
22.500 I        U                                        I
23.000 I        U                                        I
23.500 I        U                                        I
24.000 I        U                                        I
24.500 I        U                                        I
25.000 I        U                                        I
```

Chapter VIII

Conclusions and Further Research

The examples and pilot tests presented in this thesis are preliminary investigations into the feasibility of the optimal simulator design approach. The results so far are promising. The causal, linear, time-invariant "optimal" motion system derived here has parameters of the same order of magnitude as the conventional motion systems in use today. However unlike these systems, the "optimal" motion system can be "tuned" by a non-expert using our computer design method to satisfy a variety of additional conditions such as: different travel lengths of the simulator, different flight trajectories, and different emphasis on motion cues. Furthermore, it makes use of expected future airplane motions, accounts better for hard limits by use of PLQ (Pseudo Linear Quadratic) and takes into account axis transformations and head movements. It is simpler to implement and as a bonus gives the control system design for the motion-base itself.

It is recommended that this design method be transformed into an *optimal motion system design compiler* that is capable of transforming a simple minded, non-expert specification of the required motion system into a flight simulator motion-generation system. The design method that we have obtained can also be used for model-following or robot motion design.

In the first section, we describe the six elements developed in order to improve the initial causal, linear, time invariant OWS design. In the second section, we summarize the conclusion of the experiments and their bearing on the use of several of the six elements. In the third section, we describe suggestions for further research.

1. Elements in the Design of an Optimal Washout System

The main advantage of the causal, linear, time invariant OWS design is not obtaining better motion, but in finding a reasonable washout and explaining current washout designs in an ordered manner, based on knowledge about the vestibular system and not by use of engineering knowledge of a heuristic washout design based on long trial and error tests with pilots. Also, the heuristic approach does not provide a reliable measure of the simulator motion quality.

In order to further improve the washout design beyond that achieved by current washout systems, one needs to augment the causal, linear, time-invariant optimal washout system (OWS) design by six new elements. The first improvement is by use of a time varying OWS. The second is by augmenting the causal OWS by a noncausal deterministic washout as was demonstrated at the end of Chapter VII (Section 3.4, Figure VII.32). The third and more substantial improvement is using PLQ to design a nonlinear OWS. This is demonstrated in Chapter IV (Section 2.3) and in Chapter VII (Section 4, Figure 33). The fourth improvement is by use of a sign-sensitive cost in the optimization criteria for the vestibular outputs of the simulator and reference airplane pilot. This sign-sensitive cost was shown to be required based on sensory perception and vestibular physiology (Chapter IV, Section 3.1). The sign-sensitive cost was formulated by augmenting the usual cost with a correlation term (Chapter IV, Section 3.2). Using a sign-sensitive cost seems to be much harder to do in practice, since the washout design becomes very sensitive to small changes in the design parameters. In order to use this sign-sensitive cost, one is also required to use PLQ in order to better control the washout design.

The fifth improvement is an OWS implementation that accounts for the effect of the three axis systems involved in the motion design for a flight simulator, i.e. the inertial motion-base axis system, the simulator pilot axis system, and the reference airplane pilot axis system. In Chapter VI (Section 3) an example of proper simulation of the falsely called "Coriolis effect" is described using the OWS implementation. The sixth improvement is the extension of the above implementation to include the effect of head movement. This is done by consideration of another axis system, the head axis system (Chapter VI, Section 4). Further details of some of these elements are discussed in the next section.

It is unfortunate the experiment suggested in Appendix VII.A was not done, since the chance that the results of this thesis will be used for flight simulator motion design are considerably reduced. The main advantage of the use of optimal control is in the coupled design of all of the flight simulator's six degrees of freedom. The OWS testing was done on the GAT-1 using only pitch motion. Hence, this is a much less attractive design example in the eyes of potential users of the OWS design method.

2. Conclusions From the Experiments

Twenty pilots were tested using eleven different OWS designs which were implemented for the pitch and surge axes on a Link GAT-1 General Aviation flight simulator Trainer. These tests confirm the suggested design method using a causal, linear, time-invariant, "Gaussian based" OWS (L.Q.G. OWS Chapter III.4), but also point out some of the limitations of such a limited class of designs. The design is confirmed by the small range of design parameter $0.1 \leq Q_o/Q_s \leq 10$ which was experimentally found to cover the whole range that changed noticeably the OWS performance. Furthermore, the best value (based on pilot opinion and on flaps-down experiment) for this parameter seems to be between 0.32 and 1.0. This change in the ratio of otolith to semicircular canal sensitivity by a factor of $1.8\,(=\sqrt{3.2})$ confirms the ratio of threshold units chosen and the nominal design of equal weighting for the normalized vestibular linear and rotation components.

The first limitation of the above design is due to the zero mean and ergodic assumptions ("Gaussian based") used to describe the expected airplane motions. During take-off this is clearly not the case and not too surprising the take-off experiment shows the problem. An OWS with $Q_o/Q_s > 0.32$ gives the pilot a false "g-tilt" which causes experienced pilots to crash on take-off due to their reluctance to pull the airplane up, since they feel their nose is too high. This problem can be solved by a time varying OWS and by use of a deterministic washout i.e. representing the expected airplane motion more accurately using a nonzero mean process that has a time varying variance. In this case it was shown that it is sufficient to augment the OWS with a very simple deterministic washout ($\theta^s(t) = \theta^a(t)$ in the "steady state") to solve the problem (see example in chapter VI.2 and a demonstration short take-off and landing in Figure VII.32).

A second limitation of the OWS design above is its linearity. The linear OWS designer is required to compromise between hitting the motion limits (very bad) and giving the pilot more motion. In most cases it is impossible to make a reasonable compromise due to the large dynamic range of the airplane motions. Specifically in our simulation, pulling the brake after landing gives a very large deceleration which causes the simulator to hit it lower pitch limit. This problem can be solved by a nonlinear washout design using PLQ (Chapter IV Section 2.3 and Chapter VII Section 4). One should note that using a time-varying OWS would further help to solve this problem but this requires the designer to know in advance at what time or what "state" of the airplane would lead to a requirement to change the "gain" of the OWS.

The blind-test experiment shows that, unless directed, pilots do not notice major motion cues given during the simulation such as the acceleration cue during take-off (0.1 g given to pilot) and the braking after landing (0.15 g given to pilot). These linear acceleration cues are so large that all non-pilot subjects tested noticed it immediately. Nevertheless there were changes in the pilots' control which they were not aware of as documented in the washout detection and the flaps-down experiments. The flaps-down experiment also suggests a possible reason for many general aviation accidents that occur due to a stall during a landing approach (due to its similarity to this experiment). It was found that even very experienced pilots with more than ten thousand flight hours can easily be confused initially by the motion and make a wrong elevator control. However, they report making the right control. Thus we conclude that there are cases where the simulator motion has a significant influence on the pilot controls and as also seen from the take-off experiment (a crash during take-off). Furthermore pilot comments have to be treated with great caution. In order to verify the suggested reason for these stall caused accidents the national accident data bases would have to be searched.

It is not clear if the pilot reaction in the flaps-down experiment is to the incorrect rotation cue (pitch down) or to the correctly simulated deceleration cue. To resolve this question one needs to perform an experiment using a simulator with linear motion capability and/or a comparison test with a real airplane.

Pilots like the Link-like washout best due to its lower bandwidth, which reduce the chance of PIO (Pilot Induced Oscillations). The lower bandwidth is suspected to compensate for poor airplane aerodynamic and controls simulation in the GAT-1. Furthermore it is also easier to fly the Link-like washout since the motion given can be used by the pilot as an additional signal that corresponds to pitch attitude as given by the artificial horizon.

In the flaps-down experiment the time until the pilot detects the flaps-down, T_d, and the time of first elevator control, T_c, for each run should be equal, since after proper training they indicate the same thing. The pilot needs to respond to flaps-down as soon as he detects it. It was further found that the results for T_d are more consistent than the results for T_c. Thus one can assume that the flaps-down detection decision indicated by T_d is "more filtered" than that for T_c. It is surprising to find, however, that in general T_d is shorter than T_c.

3. Suggestions for Further Research

(i) Check the effect of accounting for non-vestibular inertial sensors.

(ii) Study the effect of a more detailed model for the vestibular system .

(iii) Study how to combine the control of the motion base with a g-seat.

(iv) Study the otolith-semicircular canal error trade-off using a two degree of freedom motion base with both linear and angular motion.

(v) Study the effects of head movement and their importance in high quality motion generation in flight simulators. Test the suggested washout implementation that takes into account head movements.

(vi) Check the stability of an OWS implementation that includes axis transformations (OWSI) and head movement (possibly by use of singular values).

(vii) Develop an adaptive washout as suggested in Chapter III.

(viii) Extend the design to include an optimization with "hard limits" for the deterministic washout.

(ix) Develop a washout system for a centrifuge based motion system.

(x) Prove stability for the multidimensional PLQ control approach. Further investigate the properties of a PLQ control system. Develop several more examples where PLQ can be used successfully for the control design of nonlinear plants, such as in robots.

(xi) Check the effects of using PLQ design to account for axis transformations.

(xii) Use PLQ to design a washout using a hexapod motion base.

(xiii) Design and test a full six degree of freedom motion base system using PLQ to account for the finite motion base size.

(xiv) Perform the suggested VMS experiments to test the minimum size motion base required as detailed in Appendix VII.A.

(xv) Do a comparison test of the pilot response to flaps-down and take-off experiments in the simulator to that obtained for pilots in the real airplanes. Do the results of the tests predict the reason for general aviation accidents that occur due to a stall during a landing approach? Check the relevant accident data bases to further test this suggestion.

(xvi) Add a new element to the OWS design method: a time scaling decomposition of a deterministic and stochastic subproblems as suggested for robot control (replace $\mathbf{u}^a(t)$ by $\mathbf{u}^a(r(t))$ where $r(t) - t$ can be modeled by a zero mean random process).