Rensselaer Polytechnic Institute
Troy, New York 12181

Final Report - Vol. III

Contract No. NAS8-21131

Covering period May 4, 1967-Nov. 3, 1968
NATIONAL AERONAUTICS AND
SPACE ADMINISTRATION

Optimal Control With Unavailable States

by John F. Cassidy Jr.

Submitted on behalf of

Rob Roy
Professor of Systems Engineering

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Page

ACKNOWLEDGMENTS

The author wishes to thank his colleagues, Dr. Ed Smith, Dr. James Rillings, and Mr. Chester Winsor, for the invaluable discussions and comments which helped to guide this work.

To Dr. Rob Roy, the author owes a debt of gratitude for his astute advice and firm guidance.

Also, Dr. Hans Hosenthien, Dr. Michael Borelli, and Mr. Stan Carroll of the Astrionics Laboratory of NASA's Marshall Spaceflight Center made many helpful suggestions.

The patience and skill of Miss Rosana Laviolette and Mrs. Joan Hayner, the typists and Mr. Gerard Maenhout who drew many of the figures is appreciated.

ABSTRACT

Despite the significant research effort that has been directed toward the modern control theory areas, relatively few applications have been made to practical problems. One explanation for this is that the implementation of most closed loop optimal control laws requires that all of the state variables be measured and fed back. In addition considerable computational effort is usually involved in obtaining the optimal solutions.

The Linear Specific Optimal Control Problem (SOC problem) that is formulated and solved in this document is an attempt to combine some of the practical features of the classical approaches with the analytic power of the modern theory. The formulation is based on the linear quadratic optimal control problem and has the following features.

1. Linear feedback control laws.
2. Unavailable state capbility.
3. Low computational effort.

A technique which allows the calculation of closed loop control laws which do not depend on all of the states is said to have an unavailable state capability. The above properties are obtained by specifying the structure of some of the weighting matrices of the cost index. The explicit values of these matrices are not known until the problem is solved; that is, part of the solution to a SOC problem involves the completion of the formulation.

This approach is justified from a mathematical point of view by the proof of the local existence and uniqueness of the SOC solutions and from an

engineering point of view by the successful application of the SOC technique to three general control problems, the unavailable state control problem, the model reference control problem, and the trajectory sensitivity control problem. In addition numerical methods are developed which allow these techniques to be applied with relatively low computational effort.

Of the three methods, the SOC Sensitivity approach appears to be the most promising. A significant feature of this problem is that the computational effort is relatively independent of the number of parameters considered and is of the same order as an unavailable state problem with no sensitivity considerations.

The SOC problem is the result of the application of the SOC concept, which involves the formulation of optimal control problems so that the optimal solutions have certain specified properties. The main emphasis of these formulations is on the properties of the solutions rather than the explicit values or interpretations of the cost index.

This theory is demonstrated by simple examples and the consideration of a significant engineering problem, the attitude control of the Saturn V launch vehicle. The aerodynamic instability and the flexible nature of the vehicle are factors which complicate this control problem. Critical parameters of the mathematical model of the booster are the bending frequencies, for a control system designed on the basis of a model with inaccurate bending frequencies may prove to be ineffective when applied to the actual booster. Wind gusts may cause the bending modes to be excited to such an extent that the structural integrity of the vehicle is violated. The application of the SOC Sensitivity technique resulted in a feedback control law which desensitized the rigid body responses of the vehicle to inaccurate knowledge of the bending frequencies. That is, the rigid body responses to a design test wind for bending frequencies from 80% to 100% of nominal were almost identical.

Chapter I

INTRODUCTION

1.1 Motivation

The control problem may be defined in loose terms as the manipulation of certain variables or inputs of a system to obtain a desired result or output. Almost all of man's activities may be considered as some type of a control problem. With the advent of technology, control problems on a simple scale became obvious. The water clock, windmills, and the steam engine governor were control problem solutions developed through the use of empirical methods.

Since World War II the art of control theory has come of age. Spurred on by the wartime demands, the pioneers at MIT's Lincoln Labs initiated the work which lead to a mathematical treatment of control theory. The design techniques of Bode, Evans, and Nichols, although based on mathematics, are of a cut and try nature. An initial solution for the problem is guessed, the system is analyzed by one or more of the techniques, and then another guess is made based on the results of the analysis. The effectiveness of this approach depends to a large extent on the nature of the problem and the experience of the user. Although these techniques have been used with great success, they are not very effective in attacking many of the large complex problems encountered today. Thus a more analytical approach to control systems design has been sought. Work by Wiener[1] and Newton, Gould, and Kaiser[2] were initial steps in this direction. Encouraged by their results, it was assumed that the power of the analytical approach would all but eliminate the art from control system design.

However, this has not been the case. In recent years much effort has been devoted to the analytical aspect of modern control theory with the study of the state space approach, stability theory, and optimization techniques. Unfortunately, relatively few applications of this theory have been made to problems of practical interest. These are three major difficulties preventing the widespread use of this theory. In the first place, it is often difficult to define the desired behavior or characteristics of the controlled system in precise mathematical terms. Secondly, once the problem has been formulated, it may be ill-posed from a mathematical point of view or the solutions may be difficult to calculate even with the aid of a high speed digital computer. Lastly, the solutions do not lend themselves to practical implementation.

This author feels that these difficulties do not arise from a basic limitation of the analytical approach but rather from an inappropriate formulation of the problem. It is the purpose of this work to formulate and solve an optimal control problem which will serve as a link between the theoretical and the practical. The Specific Optimal Control Problem or SOC problem presented in later sections attacks directly the last two difficulties indicated above and this theory may be used in a design procedure to reduce the first difficulty.

1.2 The SOC Concept

The SOC problem is an optimal control problem which is formulated so that its solutions have certain desirable properties. To place SOC in the proper perspective, the concept of the optimal control problem is reviewed.

The basic objective of an optimal control problem is to choose a control or set of input variables in some optimal fashion so that the output of a process or system meets certain specifications. The words process or system are used to indicate anything which involves a cause and effect relationship as shown in Fig. 1.1. In order to proceed in a precise manner the problem must be expressed in mathematical terms. The control is chosen to minimize (maximize) a mathematical function, the cost index, which in some sense reflects the desired system response or characteristics. The actual process or system is approximated by a mathematical abstraction or model which usually consists of a system of differential or difference equations which characterize the state of the system.[3] The cost index may be an integral with an integrand which is a function of the state and control.

$$\dot{\underline{x}} = \underline{f}(\underline{x}, \underline{u}, t) \; ; \quad \underline{x}(t_o) = \underline{c} \tag{1.2.1}$$

$$J = \int_{t_o}^{t_f} g(\underline{x}, \underline{u}, t) \, dt \tag{1.2.2}$$

Thus, the control, $\underline{u}$, is chosen to minimize the cost index, Eq. (1.2.2), subject to the constraint of the dynamics, Eq. (1.2.1). The necessary conditions which characterize an extremum of this problem consist of a system of differential equations which comprise a two point boundary value problem. In general, the determination of these necessary conditions and the solution of the two point boundary value problem are not trivial tasks. Moreover, it is often very difficult to translate the desired system response into the mathematical cost index function. Also, the control laws are usually of an open loop nature, that is they are not a function of the states, and they do not lend themselves to convenient implementation.

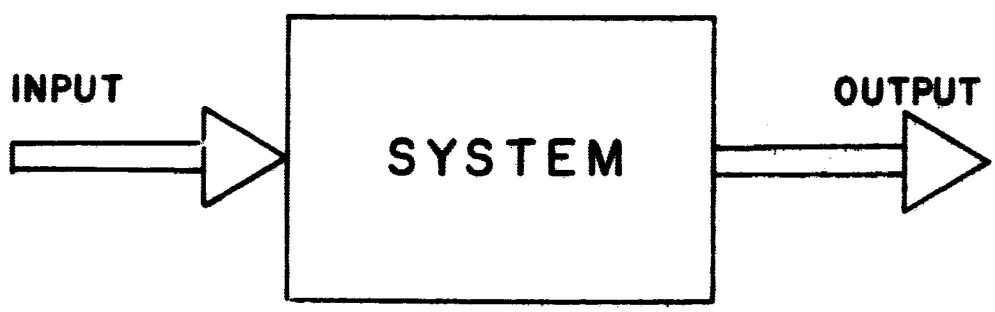


FIG. 1-1

The SOC approach attempts to combine the analytical power of optimal control theory with some of the practical aspects of the classical design techniques. To achieve this end, an optimal control problem is formulated which emphasizes certain properties of the solution. The explicit value of the cost index or its precise interpretation in terms of desired system characteristics is not of paramount importance. Rather, the optimal control formulation is used to provide a well defined structure which leads to control laws with the desirable properties. These ideas are summarized in the following definition of the SOC concept.

Definition 1 (D.1)- SOC Concept

The Specific Optimal Control Concept involves the formulation of optimal control problems so that the solutions have certain specified properties. The important consideration is not the explicit value of the cost index but rather that the minimization procedure serves as a well defined method to determine the control laws.

Thus by picking properties which allow the control laws to be of practical use, the SOC concept may generate practical analytical design procedures. The validity of the SOC approach is demonstrated by the success of the resulting techniques. Although, the SOC concept is applicable to the most general of systems, this work is concerned primarily with the study of linear systems and hereafter SOC will refer to the Linear Specific Optimal Control Problem.

1.3 Statement of the SOC Problem and Scope of the Work

The formulation of this SOC problem involves the specification of properties that the solution control laws will have and the formulation of an optimal control problem that leads to such solutions.

For reasons of sensitivity and implementation, closed loop control laws are usually specified. For linear systems, linear feedback control laws have proven to be adequate. However, care must be taken, for by closing the loop it is possible to generate stability problems. Also, the computational effort involved in calculating the control laws should not be excessive.

One of the tenents of modern control theory is that all of the states should be fed back in order to achieve optimal performance.[4] In most realistic situations it is difficult if not impossible to measure or estimate all of the states. Thus the ability to handle the unavailable state problem is of concern.

To summarize, the desired properties of the SOC solutions are listed below.

1. Linear feedback control law structure
2. Stability
3. Low computational effort
4. Unavailable state capabilities

Thus, the purpose of this work is to formulate and solve an optimal control problem with these properties. The proposed formulation, developed in later sections, is based on the linear quadratic optimal control problem. Properties of this formulation and its solutions are developed and discussed. This SOC theory is applied to three general control problems, design of controls with unavailable states, a model reference control problem, and a trajectory sensitivity control problem. Some of the properties of these techniques are discussed, examples presented, and their practical use is demonstrated by the solution of a non-trivial engineering problem, the design of a control system for the Saturn launch vehicle.

To place this work in a proper perspective a brief review of available theory and techniques is presented in the next section. In order to provide a basis for comparison the general control problems are defined below.

Definition 2 (D.2) - Unavailable State Problem

Given a model of the process or system to be controlled, a closed loop control system based on the available states is to be designed so that the controlled system meets certain specifications.

A significant problem with respect to the design of control systems for real systems concerns the relationship of the model to the actual process. Since the mathematical model is at best an approximation of the real situation. the modelling problem is in many cases a significant one. After the structure of the model is chosen, values of the parameters for this model must be obtained. For many practical problems it is very difficult to obtain accurate values for the parameters. In addition, component aging and other environmental changes lead to changes in the characteristics of the process and hence parameters of the model.

A control law designed on the basis of a nominal model may be inadequate when applied to the actual system. Thus it is important to be able to design control laws which compensate for these parameter variations. Model reference and trajectory sensitivity techniques have been used to attack this problem. In this work, SOC theory is used to develop model reference and trajectory sensitivity techniques with practical properties.

Definition 3 (D.3) - Model Reference Control Problem

In the model reference control scheme, the output of the actual system is compared with the output of a model which generates a nominal trajectory. A control system is designed, in this case with SOC techniques, to null the error between the actual and the nominal trajectories.

Definition 4 (D.4) - Sensitivity Control Problem

In the trajectory sensitivity approach, sensitivity variables are defined which are a measure of the sensitivity of the system trajectory to changes in system parameters. The sensitivity variables are placed in a cost index which is minimized by the choice of the control law. Thus, a tradeoff between system response and sensitivity may be obtained.

1.4 Historical Review

1.4.1 Unavailable State Problem

There are two basic approaches to the study of the problem of unavailable states. In the first, Kalman,[5] Luenberger,[6] and others have attacked the problem by estimating the unknown states. These estimates may then be used to formulate the control. Although the theory has been well developed, there are practical disadvantages involved in the use of this approach. The addition of the filter or state estimator to the system may unduly complicate the controller since satisfactory system performance may be obtained with controls based only on the available states. Furthermore, the use of the Kalman filter requires approximations for the statistics of the process which may not be meaningful in practical situations.

Thus, the second approach, that of calculating control laws which are a function of the available states has practical appeal. However, the theory of this approach is not as well developed as that of the first, although two basic methods have emerged. In their books Newton, Gould, and Kaiser[2] and Merriam[7] describe a straight forward parameter optimization approach. For a linear time invariant system, a linear feedback control structure depending on the available states is chosen. A set of design initial conditions is picked and an integral index with squared output and control terms is formulated. Parsevals Theorem is used to transform the integral into the frequency domain, the integration is carried out, and an expression for the index in terms of the feedback gains is obtained. This expression is minimized with respect to the gains by methods of ordinary calculus. This procedure suffers from a number of disadvantages since the gains are initial condition dependent and the method is restricted to time invariant single-input single-output systems. Also, the nonlinear functional dependence of the index expression on the gains becomes more and more complicated as the order of the system increases; for these higher order problems there is no systematic way to find this function.

In an attempt to remove the dependence of the solution upon the initial conditions, techniques employing max.-min. procedures have been developed.[8,9] A control structure is specified and a cost index is formulated as a function of the state and control. The cost index is maximized with respect to an initial condition set and then minimized with respect to the feedback gains. Although this technique is applicable to nonlinear systems, the problem of choosing an appropriate design initial condition set is not well defined and the computational effort involved in this max.-min. problem may be enormous for all but trivial examples. A recent contribution by Rekasius[10] employs a cost index which is a measure of the effectiveness of the chosen control structure to a control structure using all of the states. For linear systems, he has derived an analytical expression for the maximum of this expression with respect to all initial conditions. Thus the problem is reduced to the parameter optimization problem of picking the gains and accordingly suffers from similar disadvantages.

It is believed the SOC procedure described in this document is a new approach to the unavailable state problem. It is an application of the SOC concept of Definition 1 and is based on a linear optimal control problem with quadratic cost index. This problem was chosen as the basic structure because of the practical nature of its solutions. A brief description of the linear problem is presented so that the nature of SOC and its relationship to this theory is made clear. For a more complete exposition, the reader is referred to Kalman,[11,12] Schultz and Melsa,[4] and Athans and Falb.[13]

Anticipating that the SOC formulation will apply to the unavailable state problem. the linear quadratic optimal control problem will be referred to as the allstate problem. It is important to discuss the properties of the allstate problem since many of them will be extended to the SOC case. It is assumed that the process or system to be controlled is modeled by a system of linear differential equations.

$$\dot{\underline{x}} = A\,\underline{x} + B\,\underline{u}\ ; \quad \underline{x}(t_o) = \underline{c} \tag{1.4.1}$$

The integral cost index contains quadratic terms in state and control.

$$J = \frac{1}{2}\int_{t_o}^{t_f} (\underline{x}^T S\,\underline{x} + \underline{u}^T Q\,\underline{u})\,dt \tag{1.4.2}$$

Thus $\underline{u}$ is chosen to minimize Eq. (1.4.2) subject to Eq. (1.4.1). The necessary conditions which describe an extremum of the problem are given below and derived for the more general SOC problem in Chapter II.

Costate equation $\quad -\dot{\underline{p}} = A^T \underline{p} + S\,\underline{x}\ ; \quad \underline{p}(t_f) = \underline{0}$ (1.4.3)

Dynamics $\quad \dot{\underline{x}} = A\,\underline{x} + B\,\underline{u}\ ; \quad \underline{x}(t_o) = \underline{c}$ (1.4.4)

Control equation $\quad \underline{u} = -\,Q^{-1}\,B^T\,\underline{p}$ (1.4.5)

where $\underline{p}$ is the costate or multiplier vector.

These necessary conditions comprise a two point boundary value problem (TPBVP). It is well known that this TPBVP may be decoupled by use of the Ricatti transformation[11]

$$\underline{p} = P\,\underline{x} \tag{1.4.6}$$

where P is the Ricatti matrix. An equivalent set of necessary conditions may be written in terms of the Ricatti matrix.

Allstate Differential Ricatti Equation

$$-\dot{P} = A^T P + PA + S + PBQ^{-1}B^T P\ ; \quad P(t_f) = 0 \tag{1.4.7}$$

Dynamics

$$\dot{\underline{x}} = A\,\underline{x} + B\,\underline{u}\ ; \quad \underline{x}(t_o) = C \tag{1.4.8}$$

Control Law

$$\underline{u} = -\,K^T\,\underline{x} \tag{1.4.9}$$

Allstate Feedback Gains

$$K^T = Q^{-1}\,B^T\,P \tag{1.4.10}$$

Note that the computational effort involved in solving this problem is reduced since the TPBVP has been decoupled. The Ricatti equation may be integrated backwards in time from t_f to obtain P and K. Then integration of the dynamics in forward time generates the system trajectory. Other important features are the linear feedback control structure and the fact that the gains are independent of initial conditions. Furthermore, if the infinite time interval problem is considered, that is

$$J = \frac{1}{2} \int_{t_o}^{\infty} (\underline{x}^T S \underline{x} + \underline{u}^T Q \underline{u}) \, dt \qquad (1.4.11)$$

the Ricatti matrix and the feedback gains have constant values and are characterized by algebraic equations as opposed to differential equations.

Allstate Algebraic Ricatti Equation

$$A^T P + PA + S + PBQ^{-1}B^T P = 0 \qquad (1.4.12)$$

Allstate Feedback Gains

$$K^T = Q^{-1}B^T P \qquad (1.4.13)$$

Existence and uniqueness of solutions to the allstate problem[11] are guaranteed provided the control weighting is positive definite and the plant is completely controllable. A system (A, B) is said to be completely controllable if there exists some control $\underline{u} \in C'$ such that for any initial condition vector, the state of the system is brought to zero in some finite time. This condition is equivalent to requiring that at least NS of the NS • NC columns of $(B, AB, \ldots, A^{NS-1}B)$ be linearly independent.[14] The existence proof hinges on this restriction since it serves to provide a bound on the optimal solution to the Ricatti matrix.

Stability of the optimal closed loop system, $A - BK^T$, of the infinite time interval problem can be guaranteed by proper choice of weighting matrices and proven by a Lyapunov argument. Stability follows if (A, B) is completely controllable, the control weighting is positive definite, the state weighting is positive semi-definite, and (A, H) is completely observable. Since the state weighting, S, is positive semi-definite, it may be expressed in terms of the matrix H as[15]

$$S = H^T H \tag{1.4.14}$$

A system

$$\dot{\underline{x}} = A\,\underline{x} + B\,\underline{u}$$

$$\underline{y} = H\,\underline{x}$$

is said to be completely observable[14] if it is possible to reconstruct any set of initial conditions given $\underline{y}$ over a finite time interval. This condition is equivalent to requiring that there be NS linearly independent columns of

$$(H^T,\ A^T H^T, \ldots, A^{NS-1^T}\ H^T)$$

Thus many of the properties listed in Section 1.3 are inherent features of the allstate problem solutions. For a given design problem, the design objectives may not be modelled exactly in the quadratic index, however it has been shown[15] that the allstate solutions result in closed loop systems which have desirable properties in terms of the classical requirements of overshoot, damping, etc. Moreover if an initial solution of the problem leads to unsatisfactory system response, the weightings may be changed and the problem resolved.

The one property that is definitely missing is the unavailable state capability. However, it is clear that it is possible to stabilize certain systems by partial state feedback. Moreover, Kalman[12] has indicated that for any stable set of gains there exists a linear optimal control problem for which the given gains are the optimal control law. Thus it appears reasonable to expect that the allstate problem can be reformulated so that a specified control law structure is maintained in which only the available states are fed back. Chapter II is devoted to the formulation and solution of such a problem, the linear SOC problem. Although the SOC formulation and that of the allstate problem are similar in many respects, neither one is a subproblem of the other. The problems are different since different restrictions are made on the plants and weighting matrices. If the allowable weightings and plants are considered as sets in some abstract space, then neither set is a subset of the other although they may overlap.

1.4.2 Model Reference Control Problem

The basic objective in the model reference approach is to design a control system so that the error between the ideal output of the model and that of the actual system is nulled; two basic approaches have been used. In the first, termed model reference adaptive, on line adaptive changes in the feedback gains are made to reduce the error. Modern control theory has been applied to the design of such systems with some success. Osborn and Whitaker[17] formulated an integral cost index containing a quadratic term in the error between the system and model trajectories. An error measurement is obtained and the gradient of the index with respect to the gains is calculated on line. The gradient information is used to change the gains in order to minimize the index. Donalson and Leondes[18] employing a similar concept, added error derivative terms to the index. Dressler[19] introduced a related scheme which reduced the amount of on line computation. The most important consideration in these techniques is the stability of the adaptation procedure. A tradeoff between this stability and the rate of adaptation is obtained by the choice of the adaptation constants. There does not seem to be a well defined method for choosing these constants and an inappropriate choice often leads to instability.

In order to reduce the stability problem Parks[20] and Shackcloth[21] have taken an Lyapunov approach. A Lyapunov function with terms in the error, error derivative, and adaptation parameters is formulated and used to define the adaptation process. This approach insures that the adaptation procedure as well as the model reference system is stable. In order to implement this method it is necessary to be able to adapt all of the elements of the closed loop system matrix independently. For most systems this is not possible. From a practical point of view other disadvantages become apparent. The basic schemes involve on-line computation and measurement of all the states and in some cases state derivatives. The feasibility of such a complex control system for most realistic problems is in doubt.

The second approach to the design of model reference systems has been called model following. In this method optimal techniques are employed and the calculations are done off-line. Tyler[22] has proposed two methods. In one, the model is included in the cost index while in the other the model is incorporated into the system as a prefilter. The usual optimal control problems are present since all states must be known and the open loop terms of the control law are a function of the systems initial conditions and the input to the model. Recently, Asseo[23] has used a SOC-like concept to design a model following system which is independent of the model input.

The SOC model reference problem considered in Chapter V is of the model following type since the computations are done off-line. The SOC approach allows unavailable state capabilities and results in a control law which is independent of the nominal trajectory and hence the model input.

1.4.3 Trajectory Sensitivity Control Problem

The problem of sensitivity has always been of concern to control system designers. Bode,[24] in his pioneering work, made the basic definition of transfer function sensitivity. This measure of sensitivity is a ratio of the percent change in the transfer function to the percent change in the parameter. The reduction of sensitivity has long been advanced as a reason for using a feedback control law. Horowitz[25] made this reasoning precise with his definition of the return difference. In addition he indicated[26] that an adaptive control scheme with its inherent complex implementation might be replaced with a desensitizing feedback control law. Other frequency domain techniques such as pole zero and root locus sensitivities have been examined by Kuo[27] and Huang[28]. The basic disadvantages of these techniques involve their restriction to linear time invariant systems and the lack of information obtained about time domain sensitivity characteristics.

The development of the time domain approach has occurred relatively recently. Miller and Murray[29] made significant contributions in their study of the error involved in the numerical solution of differential equations. Dorato,[30] Rohrer and Sobral[31], and Pagurek[32] have applied optimal control techniques in their studies of the problem of cost index sensitivity. Holtzman and Horing[33] were concerned with the effect of parameter variations on terminal conditions of fixed endpoint optimal control problems.

The fundamental work which led to the Sensitivity Control Problem of Definition 4 was done by Tomovic,[34], Tuel,[35] and Dougherty[36]. Tomovic investigated various measures of sensitivity and proposed a parameter design procedure. Tuel conceived the idea of adding sensitivity variables to the cost index to be minimized by the choice of the control, and developed a design procedure for open loop controls. Dougherty extended these concepts to the closed loop case and formulated a design procedure based on control signal and parameter optimization techniques.

The optimal control approach leads to the computationally difficult two point boundary value problem, to the measurement of all the states, and to the dependency of the solution on the state initial conditions. In addition the augmented state vector formulation suffers from a dimensionality problem. For each parameter that is considered, the dimension of augmented state vector increases by the dimension of the original system state vector. For any system of any size with more than one parameter the dimension of this sensitivity problem becomes unwieldy. The SOC sensitivity problem is formulated in Chapter VI.

Nomenclature

Matrices

A	System matrix: NS by NS
B	Control coefficient matrix: NS by NC
H	Observability matrix: NS by NS
P	Ricatti Matrix: NS by NS
Q	Symmetric control weighting matrix: NC by NC
S	Symmetric state weighting matrix: NS by NS

Vectors

$\underline{c}$	State initial condition vector: NS
$\underline{p}$	Costate or multiplier vector: NS
$\underline{u}$	Control vector: NC
$\underline{x}$	State vector: NS

Scalars

J	Cost index
t	Time

Chapter II

THE SOC PROBLEM

2.1 Basic Equations

In this section the basic equations defining the SOC problem and its solution are derived. The SOC concept leads to the formulation of optimal control problems for which the solution control laws have certain specified properties. In this case, the property of importance is an unavailable state capability. For the allstate problem each of the feedback gains will in general be non-zero. The unavailable state capability is obtained by choosing some of the weighting matrices so that the gains corresponding to the unavailable states are zero. Thus, the crux of the SOC formulation involves the use of two classes of weighting matrices. The first class of weightings is chosen in the usual manner to obtain desirable system response and a tradeoff between state error and control effort and to insure the stability of the resulting closed loop system. The desired feedback structure is imposed by choosing the second class of matrices as a function of the unknown Ricatti matrix so that the unavailable state gains are forced to be zero. However, the necessary conditions are derived assuming that these weightings are known. By using these functional relations between the class two weightings and the Ricatti matrix, the formally derived necessary conditions reduce to a well defined set of equations similar to the allstate necessary conditions which do not depend on the weightings of class two. It is shown that the remaining weightings can be chosen to guarantee the existence and uniqueness of solutions to the reduced equations and hence existence and uniqueness of solutions to the formal SOC problem. The "cart before the horse" nature of this development is justified by the properties of the solutions and the effectiveness of the related techniques. If the reader is bothered by this pragmatic approach he may wish to view SOC as a Lyaponov stability design technique with a well defined procedure for generating the Lyaponov functions and the feedback control laws. However, SOC is much more than that as indicated in later sections.

The SOC control law is obtained from the minimization of an integral quadratic index, J, which contains bilinear terms between the state and control as well as the usual quadratic terms in state and control.

$$J = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^T S \underline{x} + \underline{x}^T \hat{S} \underline{x} + \underline{x}^T W \underline{u} + \underline{x}^T \hat{W} \underline{u} + \underline{u}^T Q \underline{u})\, dt \qquad (2.1.1)$$

The matrices marked with a caret, $\hat{S}$ and $\hat{W}$ belong to class two and are chosen to generate the specified SOC control structure. It is assumed that the dynamics of the systems to be controlled are modeled by a system of linear differential equations.

$$\dot{\underline{x}} = A \underline{x} + B \underline{u}; \quad \underline{x}(t_o) = \underline{c} \qquad (2.1.2)$$

Thus, $\underline{u}$ is chosen to minimize the cost index, Eq. (2.1.1), subject to the constraints of the dynamics, Eq. (2.1.2). The necessary conditions or Euler-Lagrange equations are given below and derived in Appendix A through the use of the calculus of variations.

<u>Euler-Lagrange Equations</u>

Costate equation

$$-\dot{\underline{p}} = (\frac{W}{2} + \frac{\hat{W}}{2})\, \underline{u}^o + (S + \hat{S})\, \underline{x}^o + A^T \underline{p} = \underline{0}; \quad \underline{p}(t_f) = \underline{0} \qquad (2.1.3)$$

Control Law

$$\underline{u}^o = Q^{-1}(B^T \underline{p} + (\frac{W^T + \hat{W}^T}{2}) \underline{x}^o) \tag{2.1.4}$$

Dynamics

$$\dot{\underline{x}}^o = A \underline{x}^o + B \underline{u}^o; \qquad \underline{x}(t_o) = \underline{c} \tag{2.1.5}$$

where the superscript zero indicates the optimal.

These equations comprise a two point boundary value problem which may be decoupled by the use of the Ricatti transformation.

$$\underline{p} = P \underline{x} \tag{2.1.6}$$

Equation (2.1.6) is used to eliminate $\underline{p}$ from Eq. (2.1.3) and (2.1.4) which results in an equivalent set of decoupled necessary conditions.

Unreduced Ricatti Equation

$$- \dot{P} = A^T P + PA + S + \hat{S} - (\frac{W+\hat{W}}{2} + PB) Q^{-1}(\frac{W^T+\hat{W}^T}{2} + B^T P) = 0; \qquad P(t_f) = 0 \tag{2.1.7}$$

Control Law

$$\underline{u} = - K^T \underline{x} \tag{2.1.8}$$

Feedback Gain Equation

$$K^T = Q^{-1}(\frac{W^T+\hat{W}^T}{2} + B^T P) \tag{2.1.9}$$

Dynamics

$$\dot{\underline{x}} = (A - BK^T) \underline{x} ; \qquad \underline{x}(t_o) = \underline{c} \tag{2.1.10}$$

The Ricatti matrix, P, and the feedback gain matrix, K, are found by the backward time integration of the Ricatti equation, Eq. (2.1.7); the trajectory is generated by the forward time integration of the dynamics. Note that these equations reduce to the allstate equations of Section 1.2 if the bilinear terms are zero.

From Eq. (2.1.9) it follows that $\hat{W}$ can be chosen as a function of W and P such that some of the feedback gains are identically zero. There is no loss of generality in requiring the gains to be zero since any other non-zero value may be obtained by redefining the system A matrix and then seeking zero gains. Thus if the last L states of the state vector are unavailable, define $\hat{W}$ as follows.

<u>Definition 5 (D.5) - $\hat{W}$</u>

$$\hat{W} = -2\, I_2 \left(PB + \frac{W}{2}\right) \tag{D.5}$$

where $I_2 = \begin{bmatrix} 0 & 0 \\ 0 & I_L \end{bmatrix}$ is a NS by NS matrix and I_L is the L by L identity matrix. For later use define

$$I_1 = \begin{bmatrix} I_{NS-L} & 0 \\ 0 & 0 \end{bmatrix}$$

which is a NS by NS matrix and I_{NS-L} is the NS-L by NS-L identity matrix and

$$I_1 + I_2 = I$$

the NS by NS identity matrix. Since $I_1 \, I_2 = 0$,

$$I_1 \, \hat{W} = 0$$

It is clear from (D.5) that the lower elements of W have no effect on the control law and hence on the closed loop system trajectories. Thus there is no loss in generality in assuming that they are chosen to be zero.

$$I_2 \, W = 0 \tag{2.1.11}$$

Now $\hat{S}$ is chosen to simplify the SOC necessary conditions by insuring that $\hat{W}$ will not appear in the reduced equations. Also $\hat{S}$ is required to be symmetric since only a symmetric portion of a matrix has any significance in a quadratic term.

Definition 6 (D.6) - $\hat{S}$

$$\hat{S} = \frac{1}{2}\left((W + \hat{W})\, K^T + K(W^T + \hat{W}^T)\right) \tag{D.6}$$

Using the definitions of $\hat{W}$ and $\hat{S}$ the optimal value of the cost index may be expressed as follows

$$J^\circ = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^{o^T} S\, \underline{x}^o + \underline{u}^{o^T} Q\, \underline{u}^o)\, dt \tag{2.1.12}$$

This does not imply that SOC is optimal with respect to a cost index of only quadratic terms but rather that the optimal index may be expressed as such. In fact, Kalman[12] has indicated that for a cost index of the form of Eq. (2.1.12) all of the states must be fed back.

D.5 and D.6 may be used to eliminate $\hat{W}$ and $\hat{S}$ from Eq. (2.1.7)-(2.1.10) to obtain the following:

Reduced Necessary Conditions

SOC Ricatti Equation

$$\dot{P} + A^TP + PA + S - EQ^{-1}E^T + I_2\, EQ^{-1}E^TI_2 + \frac{W}{2}\, Q^{-1}E^TI_1 + I_1EQ^{-1}\,\frac{W^T}{2} = 0\ ;$$

$$P(t_f) = 0 \tag{2.1.13}$$

where

$$E = \frac{W}{2} + PB$$

and

$$I_2\, W = 0$$

SOC Control Law

$$\underline{u} = - K^T \underline{x} \tag{2.1.14}$$

Feedback Gain Equation

$$K^T = Q^{-1}(B^T P + \frac{W^T}{2}) I_1 \tag{2.1.15}$$

Dynamics

$$\dot{\underline{x}} = (A - BK^T)\underline{x} ; \qquad \underline{x}(t_o) = \underline{c} \tag{2.1.16}$$

Note the similarity between the reduced SOC equations and the allstate equations. In fact, if W = 0 the only difference is that the quadratic terms in the Ricatti equation and the feedback gains corresponding to the unavailable states are missing.

It is convenient to rewrite the Ricatti equation in terms of the closed loop system matrix and the feedback gains. It is shown later that the two forms of the Ricatti equation are equivalent.

$$A_K = A - BK^T \tag{2.1.17}$$

$$K^T = Q^{-1}(B^T P + \frac{W^T}{2}) I_1 \tag{2.1.18}$$

$$\dot{P} + A_K^T P + PA_K + S + KQK^T = 0 ; \qquad P(t_f) = 0 \tag{2.1.19}$$

For comparison purposes the equivalent allstate equations are given below. Note that the structure of these Ricatti equations are identical, except that the SOC gains corresponding to the unavailable states are zero.

$$A_{\bar{K}} = A - B\bar{K}^T \tag{2.1.20}$$

$$\bar{K}^T = Q^{-1}(B^T P + \frac{W^T}{2}) \qquad (2.1.21)$$

$$\dot{P} + A_{\bar{K}}^T P + PA_{\bar{K}} + S + \bar{K}Q\bar{K}^T = 0 \; ; \qquad P(t_f) = 0 \qquad (2.1.22)$$

Steady State SOC Problem

If a system to be controlled is time invariant, and an infinite time interval problem with constant weightings is considered, that is $t_f \to \infty$, the solution to the Ricatti differential equation may approach a steady state value. Hence the feedback gains assume a steady state or constant value. In this case the differential equations describing the Ricatti matrix are replaced by nonlinear algebraic equations.

Steady state Ricatti equation

$$A^T P + PA + S - EQ^{-1}E^T + I_2 EQ^{-1}E^T I_2 + \frac{W}{2} Q^{-1}E^T I_1 + I_1 EQ^{-1} \frac{W^T}{2} = 0 \qquad (2.1.23)$$

where

$$E = PB + \frac{W}{2}$$

or

$$A_K^T P + PA_K + S + KQK^T = 0 \qquad (2.1.24)$$

where

$$A_K = A - BK^T$$

$$K^T = Q^{-1}(B^T P + \frac{W^T}{2}) \, I_1 \qquad (2.1.25)$$

In the following sections the properties which indicate that SOC may be a useful tool for the study of linear systems are described.

2.2 SOC Properties

In Section 2.1 the basic equations of the SOC problem were formally derived. In this section the significance and usefulness of the SOC problem is indicated by the examination of the properties of the SOC equations and solutions.

In order to guarantee that the SOC solutions will have certain properties it is necessary to make restrictions on the allowable systems and weighting matrices. The reasons for these restrictions will become clear as the properties are developed.

Restriction 1 (R.1) - Weighting Matrices

The control weighting, Q, must be a symmetric positive definite matrix.

The state weighting matrix, S, must be a symmetric, positive semi-definite, SOC observable matrix.

Definition 7 (D.7) - SOC Observability

Since S is positive semi-definite, it may be expressed[15] as

$$S = H^T H$$

where H is a NS by NS matrix. Now a system, A, and weighting matrix, S, are said to be SOC Observable if the matrix pair (A, H) is completely observable as defined by Kalman.[14]

Note that this definition differs from the Kalman allstate definition since the former involves a portion of the state weighting while the latter involves all of the state weighting. A further restriction on the allowable systems must be made, since it makes no sense to talk about the minimization of a cost index if there are no control laws (feedback gains) which result in a finite value of that index.

Definition 8 (D.8) - SOC Controllability

A system, A, is said to be SOC Controllable with respect to a specified feedback structure provided there exist finite values of feedback gains, $K \in C^1$, such that all initial condition responses are square integrable.

Let

$$\dot{\underline{x}} = (A - BK^T)\, \underline{x} = A_K\, \underline{x} \;; \qquad \underline{x}(t_o) = \underline{c}$$

and

$$\underline{x}(t) = \Phi_K(t,\, t_o)\, \underline{c}$$

where $\Phi_K(t, t_o)$ is the state transition matrix for the closed loop system A_K. Then for all c such that $\|c\| < \infty$

$$V = \int_{t_o}^{t_f} \underline{x}^T \underline{x}\, dt = \underline{c}^T \left\{ \int_{t_o}^{t_f} \Phi_K^T(\tau, t_o)\, \Phi_K(\tau, t_o)\, d\tau \right\} \underline{c} < \infty$$

For a linear time invariant system and the steady state problem, this condition is equivalent to the existence of a set of constant feedback gains such that the closed loop system in stable.

Existence and Uniqueness

The motivation for the SOC Controllability definition is provided by the following lemma which states a necessary condition for existence. The proof of this lemma follows directly from the definition.

Lemma 1:

A necessary condition for the existence of the solution to a SOC problem is that the plant and chosen feedback control structure be SOC Controllable.

A distinction must be made between existence and uniqueness properties of the SOC equations in reduced and unreduced forms. That is, given all the weightings of the formal SOC index the existence and uniqueness of the solutions to the necessary conditions may be demonstrated in exactly the same way as in the allstate case.

However, in order to use the SOC theory the question of the existence of the solutions to the reduced equations must be answered. The important point is that the choice of the class two matrices leads to a well defined set of equations (the reduced equations) in which these matrices do not appear. The existence of solutions to these equations is a justification of the SOC approach. If solutions exist to the reduced equations, the SOC procedure is shown to be valid from a mathematical point of view and only the interpretation of or motivation for the SOC problem from an engineering point of view is of concern.

The finite time interval and steady state problems lead to the study of systems of nonlinear differential and algebraic equations, respectively. These equations are very similar to the allstate equations. However, the approach used in the proof by Kalman[11] does not appear to be applicable in the SOC case. Demonstrating the existence of solutions to these problems is equivalent to proving the existence of solutions to the Ricatti equations. The fundamental point of Kalman's proof involves the derivation of a bound on the solution to the Ricatti equation. An attempt to follow this same path for the SOC Ricatti equation fails, since it leads to a bound that is a function of the Ricatti matrix. Despite significant effort along these lines, no general existence theorem has been developed. However, for some specific examples it is possible to say something positive about general existence. See the example at the end of this chapter.

It is fairly easy to prove local existence of a special nature with the aid of the Reverse SOC problem described below. This reverse problem provides an initial solution to the reduced SOC equations. A perturbation of the weighting matrices leads to a new set of equations which in some sense are close to the reverse problem equations. Thus the question of existence and uniqueness may be answered in terms of the solutions to equations containing parameters.

Since both differential and algebraic Ricatti equations are encountered, two types of existence proofs must be demonstrated. The results are stated in theorem form for preciseness and clarity and the proofs involve the application of certain well known theorems of analysis and differential equation theory.

Definition 9 (D.9) - Reverse SOC Problem

Given a set of feedback gains, determine if there exists a SOC index such that the gains are the SOC control law.

In order for the steady state reverse problem to have a solution, the allowable feedback gain must be stable, that is, the closed loop system is stable, while for the finite time interval problem any set of finite gains contained in C^1 will suffice.

Theorem 1:

For all SOC controllable systems with any set of allowable feedback gains, there exists a nonunique SOC problem with weighting matrices satisfying (R.1) and for which the given gains are the optimal control law.

Proof A: Steady State Problem

Choose any S and Q which satisfy (R.1) and such that $S + KQK^T$ is positive definite. For example S and Q might be the appropriate dimensioned identity matrices. Since the feedback gains are stable by assumption and $S + KQK^T$ is positive definite, there exists a unique positive definite solution, P, to the SOC Ricatti equation.[37]

$$A_K^T P + PA_K = - S - KQK^T \tag{2.2.1}$$

Note that $S + KQK^T$ is symmetric and since P^T also satisfies Eq. (2.2.1) which has a unique solution, the Ricatti matrix is symmetric. The feedback gain equation is used to find W,

$$\begin{aligned} W^T I_1 &= 2(QK^T - B^T P I_1) \\ W^T I_2 &= 0 \end{aligned} \tag{2.2.2}$$

while $\hat{W}$ and $\hat{S}$ are determined from their respective definitions. Thus the reverse SOC problem for which the given gains are optimal is specified. This problem is not unique since the choice of S and Q is not unique.

Proof B: Finite Time Problem

Again choose a S and Q which satisfy (R.1). The Ricatti matrix P is found by solving the SOC Ricatti differential equation where K, A, S, and Q are known.

$$\dot{P} + A_K^T P + PA_K + S + KQK^T = 0; \qquad P(t_f) = 0 \tag{2.2.3}$$

$$t_o \leq t \leq t_f$$

To show that a unique, positive definite solution to this differential equation exists, the following lemma will be useful.

Lemma 2:

The value of the optimal SOC index may be expressed in terms of the Ricatti matrix which is necessarily positive definite if (R.1) is satisfied.

$$J^o = \frac{1}{2} \underline{c}^T P(t_o) \underline{c} = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^{o^T} S \underline{x}^o + \underline{u}^{o^T} Q \underline{u}^o) dt \tag{2.2.4}$$

where P is the solution to the SOC Ricatti equation, Eq. (2.2.3), and $\underline{c}$ is a state initial condition vector.

Proof:

Equation (2.2.4) is derived by the manipulation of the SOC necessary conditions. Adjoin the dynamics to the cost index with the costate vector and integrate by parts.

$$J^o = \frac{1}{2} \int_{t_o}^{t_f} [\underline{x}^{o^T} S\, \underline{x}^o + \underline{u}^{o^T} Q\, \underline{u}^o + \underline{p}^T (A_K \underline{x}^o - \dot{\underline{x}}^o)]\, dt$$

or

$$J^o = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^{o^T} S\, \underline{x}^o + \underline{u}^{o^T} Q\, \underline{u}^o + \underline{p}^T A_K \underline{x}^o + \dot{\underline{p}}^T \underline{x}^o)\, dt - \frac{1}{2} \underline{x}^T \underline{p} \Big|_{t_o}^{t_f}$$

Using the Ricatti transformation, terminal conditions on $\underline{p}$, and the control law equations leads to

$$J^o = \frac{1}{2} \int_{t_o}^{t_f} [\, \underline{x}^{o^T} (S + KQK^T + PA_K + A_K^T P + \dot{P}) \underline{x}^o]\, dt + \frac{1}{2} \underline{c}^T P(t_o)\, \underline{c}$$

But the expression in the integral is the Ricatti equation, hence for any t_o and t_f

$$J^o = \frac{1}{2} \underline{c}^T P(t_o)\, \underline{c} \tag{2.2.5}$$

(R.1) requires S to be positive semi-definite and Q to be positive definite. The SOC observable requirement insures that $\underline{x}^T S\, \underline{x}$ will not be zero for any allowable trajectory.[37] Thus J^o is positive for any $\underline{c}$ and hence P is positive definite.

Now P can be expressed in terms of the state transition matrix for the closed loop system.

$$\dot{\underline{x}} = (A - BK^T)\,\underline{x}\ ; \qquad \underline{x}(t_o) = \underline{c}$$

$$\underline{x}(t) = \Phi_K(t, t_o)\,\underline{c}$$

then

$$J^o = \frac{1}{2}\,\underline{c}^T P \underline{c} = \frac{1}{2}\int_{t_o}^{t_f} [\,\underline{c}^T \Phi_K^{\ T}(\tau, t_o)(S + KQK^T)\,\Phi_K(\tau, t_o)\,\underline{c}\,]\,d\tau$$

or

$$P(t_o) = \int_{t_o}^{t_f} \Phi_K^{\ T}(\tau, t_o)(S + KQK^T)\,\Phi_K(\tau, t_o)\,d\tau \quad ; \quad t_o \leq \tau \leq t_f \tag{2.2.6}$$

Since Eq. (2.2.5) holds for all t_o, $t_o \leq t_f$, Eq. (2.2.6) defines $P(t)$.

$$P(t) = -\int_{t_f}^{t} \Phi_K^{\ T}(\tau, t)\,(S + KQK^T)\,\Phi_K(\tau, t)\,d\tau \tag{2.2.7}$$

Recall that $\Phi_K(t, t) = I$ and $\frac{d}{dt}\Phi_K = A_K\Phi_K$. Taking the time derivative of Eq. (2.2.7) leads to

$$\dot{P}(t) = -S - KQK^T - A_K^{\ T}P - PA_K$$

which is the Ricatti equation. Thus the existence and positive definiteness of P is established. Since Eq. (2.2.3) satisfies a Lipschitz condition, the uniqueness is demonstrated by the application of a standard theorem of differential equation theory.

As in the case of the steady state problem, W is chosen to satisfy the gain equation and $\hat{S}$ and $\hat{W}$ are found from their definitions.

It has been shown that the Reverse problem determines well behaved solutions to the SOC necessary conditions. Existence properties of these equations may be studied by considering the weighting matrices as parameters. For sets of weighting matrices which are suitably close to those of the Reverse problem, something may be said about the uniqueness and existence of the solutions to the corresponding SOC problems. To facilitate the discussion consider the following notation. A weighting vector $\underline{g}$ is formed from all the independent elements of S, Q, and W in column order. (Only the lower or upper triangular elements of the symmetric matrices are considered.) The weighting vector $\underline{g}$ may be pictured as a point in a finite dimensional Euclidean space, where the corresponding norm may be denoted by $\|g\|$. With this notation the concept of one set of weightings being close to another can be made precise.

Theorem 2:

Given a Reverse problem solution for a finite time interval problem, characterized by a weighting vector $\underline{g}_o$, solutions to the SOC problem exist and are unique for all weightings in some neighborhood $\mathcal{D}$ of $\underline{g}_o$.

Proof:

The existence of a solution to the SOC problem is equivalent to the existence of a solution to the SOC Ricatti equation over the time interval of interest. Since P is symmetric, this matrix differential equation can be written as a vector differential equation of dimension $NP = \frac{NS(NS + 1)}{2}$

$$"\dot{\underline{P}}" = - "(A^TP + PA + S - EQ^{-1}E^T + I_2EQ^{-1}E^TI_2 + \frac{W}{2} Q^{-1}E^TI_2 + I_1EQ^{-1} \frac{W^T}{2})"$$

$$"\underline{P}"(t_f) = \underline{0} \qquad (2.2.8)$$

where $E = PB + \frac{W}{2}$; $I_2 W = 0$ and "D" indicates a vector formed from the matrix D as follows.

$$"D"^T = (D_{1,1};\ D_{2,1};\ldots;D_{NS,1};\ D_{2,2};\ldots;D_{NS,NS})$$

or symbolically

$$\dot{\underline{"P"}} = \underline{F}(\underline{"P"}, \underline{g});\qquad \underline{"P"}(t_f) = \underline{0} \tag{2.2.9}$$

From Eq. (2.2.8) it is clear that partial derivatives of $\underline{F}$ with respect to the elements of P exist and are continuous and thus satisfy a Lipschitz condition in some neighborhood of $\underline{g}_o$. The existence and uniqueness of the solutions in some neighborhood, $\mathcal{N}$, of $\underline{g}_o$ is a standard result from the theory of differential equations. See Theorem 7.5 of Reference 38.

A similar theorem for the steady state problem may be demonstrated with the aid of the Implicit Function theorem. To clarify the discussion, consider each set of feedback gains as a point in some Euclidean space. This point is denoted by the feedback gain vector $\hat{\underline{k}}$ formed by the column ordering of the feedback gain matrix K.

Theorem 3:

Given a solution to a steady state Reverse SOC problem with gain vector $\hat{\underline{k}}_o$ and weighting vector $\underline{g}_o$, there exists a unique solution to the SOC problem for weightings in some neighborhood $\mathcal{N}$ of $\underline{g}_o$. Moreover, the stable feedback gains are continuous functions of the weighting vector.

Proof:

The proof will be carried out by the application of the Implicit Function theorem to a feedback gain vector function. Consider the steady state SOC equations

$$A_K^T P + PA_K + S + KQK^T = 0 \qquad (2.2.10)$$

$$K^T = Q^{-1}(B^T P + \frac{W^T}{2}) I_1 \qquad (2.2.11)$$

Since the eigenvalues of the closed loop system are continuous functions of the feedback gains, A_K is stable for all feedback gain vectors $\hat{\underline{k}}$ contained in a suitably small neighborhood of $\hat{\underline{k}}_o$, $\mathcal{K}$. Since A_K is stable, P may be found as a unique function of K, S, Q, and W. Using the equivalent vector notation introduced above, Eq. (2.2.10) may be rewritten as a linear system of NP equations.

$$E \text{"P"} = - \text{"}(S + KQK^T)\text{"}$$

$$\text{"P"} = - E^{-1} \text{"}(S + KQK^T)\text{"}$$

or symbolically

$$\text{"P"} = \underline{F}_1(\underline{k}, \underline{g})$$

where

$$E = \text{"}(I*A_K + A_K^T*I)\text{"}$$

and * represents the Kronecker product. The matrix E is simply the Kronecker matrix manipulated in the appropriate manner to form the coefficient matrix for the linear system. Both the matrix and vector forms represent the same system of scalar equations, with a particular form chosen by the context of the discussion.

To return to the proof, a vector gain function can be written as follows.

$$\underline{F}(\hat{\underline{k}}, \underline{g}) = {}^{\square}(Q^{-1} B^T P(\underline{k}, \underline{g}) - K^T)^{\square} = 0$$

The notation ${}^{\square}\underline{D}^{\square}$ indicates that a vector has been formed from the matrix D by column ordering.

$${}^{D}\underline{D}^{D\,T} = (D_{1,1};\ldots;D_{NS,1};\ D_{1,2};\ldots;D_{NS,NS})$$

Now the Implicit Function theorem may be applied to this equation[39], provided that

C1 : $\underline{F}(\hat{\underline{k}}, \underline{g}) = 0$ at $\hat{\underline{k}}_o$, $\underline{g}_o$

C2 : $\underline{F}(\hat{\underline{k}}, \underline{g}) \in C^1$

C3 : Jacobian at $(\hat{\underline{k}}_o, \underline{g}_o)$ is non-zero.

The Jacobian is the determinant of the partial derivative matrix of $\underline{F}$ with respect to $\hat{\underline{k}}$.

$$J = \det \left(\frac{\partial \underline{F}}{\partial \hat{\underline{k}}}\right)$$

where

$$\left(\frac{\partial \underline{F}}{\partial \hat{\underline{k}}}\right)_{ij} = \frac{\partial \underline{F}_i}{\partial \hat{\underline{k}}_j}$$

This theorem indicates that within some suitably small neighborhoods, $\mathcal{G}_o$ and $\mathcal{K}_o$ of $\underline{g}_o$ and $\hat{\underline{k}}_o$ respectively, there exists a unique continuous vector function $\underline{\theta}$ such that

$$\hat{\underline{k}} = \underline{\theta}(\underline{g})$$

$$F(\underline{\theta}(\underline{g}), \underline{g}) = 0 \qquad \underline{g} \in \mathcal{G}_o \qquad \underline{k} \in \mathcal{K}_o$$

Since only stable gains are of interest, the neighborhood of $\underline{g}$ is further restricted so that for any $\underline{g} \in \mathcal{G}$, $\underline{k} \in \mathcal{K}$ is a stable gain vector.

It is clear that conditions C1 and C2 are satisfied while C3 must be considered more closely.

Lemma 3:

For any stable set of gains $\hat{\underline{k}}_o$, it is possible to find a Reverse SOC problem characterized by $\underline{g}_o$ such that the Jacobian is non-zero $J(\hat{\underline{k}}_o, \underline{g}_o) \neq 0$.

Proof:

Note that the gain function equation can be written as

$$\underline{F}(\hat{\underline{k}}, \underline{g}) = {}^{\square}Q^{-1}B^TP^{\square} - \hat{\underline{k}}^T$$

Then

$$\frac{\partial \underline{F}}{\partial \hat{\underline{k}}} = {}^{\square}Q^{-1}\, B^T \frac{\partial P^{\square}}{\partial \hat{\underline{k}}} - I \qquad (2.2.12)$$

If $J(\hat{\underline{k}}_o, \underline{g}_o) = 0$, then at least one eigenvalue of $\frac{\partial F}{\partial \hat{\underline{k}}}$ must be zero. Thus, from Eq. (2.2.12) it follows that at least one eigenvalue of ${}^{\square}Q^{-1}B^T\frac{\partial P^{\square}}{\partial \underline{k}}$ is equal to 1. However, by a proper choice of $\underline{g}_o$, that is S and Q it is possible to insure that this is not the case.

Consider the $\frac{\partial P}{\partial \hat{\underline{k}}}$ term of the matrix in question. For convenience examine the equivalent matrix

$$\frac{\partial \text{"P"}}{\partial \hat{\underline{k}}}$$

Again this is a notational switch to allow for convenient manipulation. Since $\text{"}\underline{P}\text{"} = - E^{-1}\, \text{"}(S + KQK^T)\text{"}$

$$\frac{\partial \text{"P"}}{\partial \hat{\underline{k}}} = - \frac{\partial E^{-1}}{\partial \hat{\underline{k}}} \text{"}(S + KQK^T)\text{"} - E^{-1}\, \frac{\partial \text{"}(S + KQK^T)\text{"}}{\partial \hat{\underline{k}}}$$

If the Jacobian is zero it is possible to pick new values of S and Q to insure that the Jacobian is not zero. Thus C3 is satisfied and the proof is complete.

The local existence properties are sufficient to allow the practical use of the SOC theory as indicated in later chapters.

Stability

For the steady state SOC problem, the feedback control law consists of constant feedback gains. A linear system with such a control law is said to be stable if all the eigenvalues of the closed loop systems have negative real parts. In addition these feedback gains are said to be stable. It should be emphasized that an optimal control law is not necessarily a stable control law! It is possible to formulate an optimal control problem for an unstable plant for which the optimal control law and the cost index are identically zero. The steady state SOC problem has been structured so that the resultant closed loop system is necessarily stable.

Theorem 4:

Consider a SOC problem with a SOC controllable plant and weighting matrices which satisfy (R.1). For any constant feedback gain matrix, K, a necessary and sufficient condition that K be a set of stable SOC feedback gains is that there exist a Ricatti matrix, P, with the following properties.

C1 : $A_K = A - BK^T$

C2 : $K^T = Q^{-1}(B^TP + \frac{W^T}{2})\, I_1$

C3 : $A_K^T P + PA_K = - S - KQK^T$

C4 : P is positive definite and symmetric

Proof:

Necessity - By definition, if K is a matrix of stable SOC feedback gains the necessary conditions, Eqs. (2.1.23) and (2.1.25) are satisfied. Substitution of (2.1.16) and (2.1.25) into (2.1.23) leads to (2.1.24) and C3. The positive definiteness was demonstrated in Lemma 2 and the symmetry is easily shown. Since K is stable, there exists a unique solution P to C3. Since S and Q are symmetric P^T also satisfies C3, hence $P^T = P$.

Sufficiency - Let K be a constant matrix of feedback gains for a system, A. Let S and Q be matrices which satisfy (R.1) and let P and W be matrices such that C1, C2, C3 and C4 are satisfied. Constant values of $\hat{S}$ and $\hat{W}$ of the SOC index may be calculated using P, S, and W. Then it is clear that P is the solution to Eq. (2.1.7), the unreduced Ricatti equation.

The stability property is presented in the following lemma.

Lemma 4:

Given that the weighting matrices satisfy the hypothesis of the theorem and that C1, C2, C3 and C4 are satisfied A_K is asymptotically stable.

Proof:

The lemma is proved by a Lyapunov argument. Let $V = \underline{x}^T P \underline{x}$ be a positive definite Lyapunov function. Then

$$\dot{V} = - \underline{x}^T (S + KQK^T) \underline{x}$$

and asymtotic stability is guaranteed since V is negative over any possible trajectory.[4] Requiring S to be positive definite would be sufficient to insure the negative definiteness of V, but the SOC Observable restriction of (R.1) guarantees that $\underline{x}^T S \underline{x}$ will not be zero along any possible trajectory. This weaker requirement was introduced by Kalman[12] for the allstate problem.

The proof will be complete provided the unreduced steady state Ricatti equation has a unique positive definite solution. In that case the steady state solution to Eq. (2.1.7) and the solution to C3 must be identical. This uniqueness property can be shown by reformulating the SOC problem into an allstate problem and applying Kalman's allstate result.

An allstate control $\underline{u}$ is chosen to minimize

$$J = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^T \bar{S} \underline{x} + \underline{u}^T Q \underline{u}) \, dt \qquad (2.2.13)$$

subject to

$$\dot{\underline{x}} = \bar{A} \underline{x} + \bar{B} \underline{u} ; \qquad \underline{x}(t_o) = \underline{c} \qquad (2.2.14)$$

Define the following relationship between the SOC control $\underline{u}$, and the allstate control $\bar{\underline{u}}$.

$$\bar{\underline{u}} = \underline{u} + Q^{-1} \left(\frac{W^T + \hat{W}^T}{2}\right) \underline{x} \qquad (2.2.15)$$

Then

$$\bar{\underline{u}}^T Q \bar{\underline{u}} = \underline{u}^T Q \underline{u} + \frac{1}{2} \underline{x}^T(W + \hat{W}) \underline{u} + \frac{1}{2} \underline{u}^T(W^T + \hat{W}^T) \underline{x}$$

$$+ \frac{1}{4} \underline{x}^T(W + \hat{W}) Q^{-1}(W^T + \hat{W}^T) \underline{x}$$

or

$$\underline{u}^T Q \underline{u} + \underline{x}^T(W + \hat{W}) \underline{u} = \bar{\underline{u}}^T Q \bar{\underline{u}} - \frac{1}{4} \underline{x}^T(W + \hat{W}) Q^{-1}(W^T + \hat{W}^T) \underline{x} \qquad (2.2.16)$$

The SOC index is given by

$$J = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^T (S + \hat{S}) \underline{x} + \underline{x}^T(W + \hat{W}) \underline{u} + \underline{u}^T Q \underline{u}) \, dt \qquad (2.2.17)$$

Substituting Eq. (2.2.16) into (2.2.17) leads to

$$J = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^T(S + \hat{S} - \tfrac{1}{4}(W + \hat{W})\, Q^{-1}(W^T + \hat{W}^T))\, \underline{x} + \underline{\bar{u}}^T Q\, \underline{\bar{u}})\, dt \qquad (2.2.18)$$

To insure that both problems have the same trajectory, require that the dynamics be equal.

$$\dot{\underline{x}} = A\, \underline{x} + B\, \underline{u} = A\, \underline{x} + B(\underline{\bar{u}} - Q^{-1}\, (\frac{W^T + \hat{W}^T}{2})\, \underline{x})$$

or

$$\dot{\underline{x}} = \bar{A}\, \underline{x} + \bar{B}\, \underline{\bar{u}}$$

where

$$\bar{A} = A - B\, Q^{-1}\, (\frac{W^T + \hat{W}^T}{2})$$

$$\bar{B} = B$$

By requiring the indices to be equal, Eq. (2.2.13) and (2.2.18), the definition for S is obtained

$$\bar{S} = S + \hat{S} - \frac{(W + \hat{W})\, Q^{-1}(W^T + \hat{W}^T)}{4}$$

Thus the problems are equivalent and choosing $\underline{\bar{u}}$ to minimize Eq. (2.2.13) will give the same answer as choosing $\underline{u}$ to minimize Eq. (2.1.1). Kalman[11] has shown that there is a unique positive definite solution to the steady state allstate Ricatti equation. Since the Ricatti equations for the two problems discussed above are identical, this result also holds for the unreduced SOC equations. Kalman's proof depends on the structure of the equations and not on his restrictions on the state weighting of the allstate problem; it is possible that $\bar{S}$ may not satisfy the Kalman restrictions. Thus the proof of Theorem 4 is complete.

In addition to demonstrating the stability of the SOC control law this theorem has characterized the optimallity of a feedback control law in terms of the existence of a positive definite solution to a system of non-linear equations, the steady state SOC Ricatti equation.

2.3 Example

To clarify the formulation and indicate some of the properties of SOC, a simple second order damped oscillator example is presented. For a more practical example see Chapter VII which is a case study of the use of SOC to design a control system for a large flexible launch vehicle.

The state space representation of the example is given below and pictured in Fig. 2.1.

$$\ddot{y} + 2\zeta\omega\dot{y} + \omega^2 y = u$$

$$x_1 = \dot{y}$$

$$x_2 = y$$

$$\dot{\underline{x}} = A\,\underline{x} + B\,u; \qquad \underline{x}(t_o) = \underline{c}$$

where

$$A = \begin{bmatrix} -2\zeta\omega & -\omega^2 \\ 1 & 0 \end{bmatrix}; \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Assume that a rate feedback control law structure has been specified.

$$u = -k\,x_1 = -k\,\dot{y}$$

Now NS = 2, NC = 1, and L = 1. Let

$$Q = q; \quad S = \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix}; \quad P = \begin{bmatrix} P_1 & P_2 \\ P_2 & P_3 \end{bmatrix}; \quad W = \begin{bmatrix} W_1 \\ 0 \end{bmatrix}$$

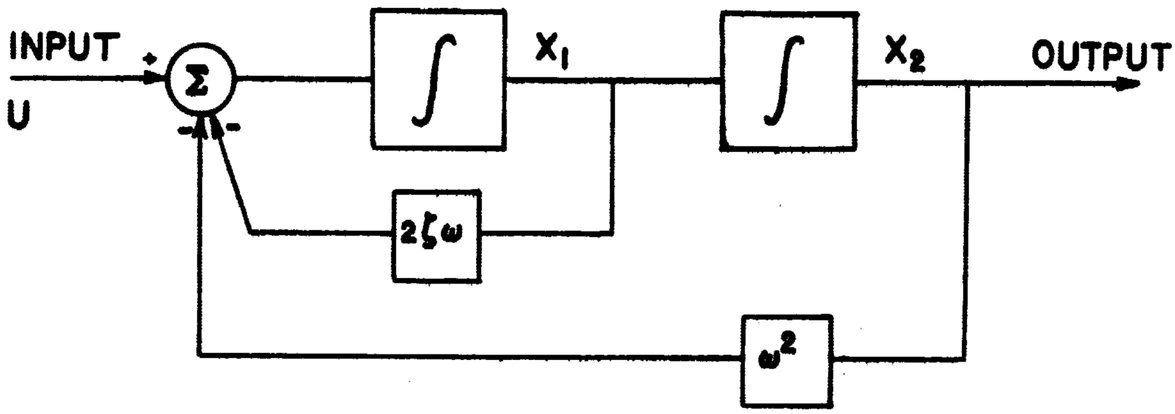


FIG. 2-1

The control u is chosen to minimize the following SOC index.

$$J = \frac{1}{2} \int_0^{\infty} (\underline{x}^T S \underline{x} + \underline{x}^T \hat{S} \underline{x} + \underline{x}^T (W + \hat{W}) u + q u^2) dt$$

The SOC steady state matrix Ricatti equation may be written as a system of three scalar equations.

$$A_K^T P + PA_K = - S - KQK^T \qquad (2.3.1)$$

or

$$2P_1(-2 \zeta \omega - k) + 2 P_2 = - S_1 - q k^2 \qquad (2.3.2)$$

$$-P_1 \omega^2 + P_2(-2 \zeta \omega - k) + P_3 = - S_2 \qquad (2.3.3)$$

$$-2\omega^2 P_2 = - S_3 \qquad (2.3.4)$$

The scalar gain is found from

$$K^T = Q^{-1}(B^T P + \frac{W^T}{2}) I_1 \qquad (2.3.5)$$

$$k = \left(\frac{P_1 + \frac{W_1}{2}}{q}\right) \qquad (2.3.6)$$

The elements of the Ricatti matrix and the feedback gains are found by the simultaneous solution of Eq. (2.3.2)-(2.3.4) and (2.3.6). Recall that the positive definite solution is sought.

$$P_1 = - 2 q \zeta \omega + \sqrt{4 q^2 \zeta^2 \omega^2 + S_1 q + \frac{W_1^2}{4} + \frac{S_3 q}{\omega^2}} \qquad (2.3.7)$$

$$P_2 = - \frac{S_3}{2\omega^2} \qquad (2.3.8)$$

$$P_3 = P_1 \omega^2 + P_2(2 \zeta \omega + k) - S_2 \qquad (2.3.9)$$

$$k = \frac{W_1}{2q} - 2 \zeta \omega + \frac{1}{q} \sqrt{4 q^2 \zeta^2 \omega^2 + S_1 q + \frac{W_1^2}{4} + \frac{S_3 q}{\omega^2}} \qquad (2.3.10)$$

The appropriate equations are used to find the matrices which complete the formulation of the SOC problems.

$$I_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\hat{W} = -2\, I_2(PB + \frac{W}{2}) = \begin{bmatrix} 0 \\ -2P_2 \end{bmatrix}$$

$$S = \frac{1}{2}((W + \hat{W})\, K^T + K(W + \hat{W})^T)$$

$$\hat{S} = \begin{bmatrix} \hat{S}_1 & \hat{S}_2 \\ \hat{S}_2 & \hat{S}_3 \end{bmatrix}$$

$$\hat{S}_1 = W_1\, k = \frac{W_1(2P_1 + W_1)}{2q}$$

$$\hat{S}_2 = -\frac{P_2(2P_1 + W_1)}{2q}$$

$$\hat{S}_3 = 0$$

To be more specific, consider some typical numbers. Let $\zeta = 0$ and $\omega = 1$ and choose

$$S = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} ; \quad W = \begin{bmatrix} 0 \\ 0 \end{bmatrix} ; \quad q = 1$$

From Eqs. (2.3.7)-(2.3.10)

$$P_1 = \sqrt{2}$$

$$P_2 = \frac{1}{2}$$

$$P_3 = \frac{3}{2}\sqrt{2}$$

$$k = \sqrt{2}$$

Thus

$$A_K = A - BK^T = \begin{bmatrix} -\sqrt{2} & -1 \\ 1 & 0 \end{bmatrix}$$

Note that the closed loop systems has a characteristic frequency of 1 radian/sec. and a damping ratio, ζ , of .707. The remaining SOC weighting matrices are given by

$$\hat{W} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

and

$$S = \begin{bmatrix} 0 & -\frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} & 0 \end{bmatrix}$$

Finally, the SOC index may be written as

$$J = \frac{1}{2} \int_{t_o}^{\infty} (x_1^2 - \sqrt{2}\, x_1 x_2 + x_2^2 - x_2 u + u^2)\, dt$$

From Eq. (2.3.4) it is seen that ω must be non-zero in order that the solutions to the SOC problem exist. What does this imply? An examination of the properties of the A matrix with $\omega = 0$ indicates that with a rate feedback control law, the system is not SOC controllable. The characteristic equation for the closed loop system is given below.

$$\det (SI - A_K) = 0$$

where

$$A_K = A - BK^T$$

Thus

$$\det \begin{bmatrix} S+k & 0 \\ -1 & S \end{bmatrix} = 0$$

or

$$S(S+k) = 0$$

The characteristic roots are

$$S_1 = 0$$

$$S_2 = - k$$

Clearly, there exists no value of k such that the closed loop system is asymtotically stable; thus this particular system and control structure is not SOC controllable.

For this example, SOC solutions exist for any set of matrices which satisfy (P.1). Note that any positive gain is sufficient for asymtotic stability of the closed loop system and any positive semi-definite S is SOC observable. From Eq. (2.3.10) any positive definite S with positive q leads to a positive value of gain for any W_1 and hence existence of SOC solutions.

In the chapters to follow, the computational aspects of the SOC problem are discussed and the SOC concept is applied to various general control problems of current interest.

Nomenclature

Matrices

A	System matrix: NS by NS
$\bar{A}$	Equivalent allstate system matrix: NS by NS
$A_{\bar{K}}$	Allstate closed loop system matrix: NS by NS
A_K	SOC closed loop system matrix: NS by NS
B	Control coefficient matrix: NS by NC
$\bar{B}$	Equivalent allstate control coefficient matrix: NS by NC
D	Notational matrix
E	Notational matrix
H	Observability matrix: NS by NS
I_1	Notational matrix
I_2	Notational matrix
I_L	Notational matrix
K	SOC feedback gain matrix: NS by NC
$\bar{K}$	Allstate feedback gain matrix: NS by NC
P	Ricatti matrix
Q	Symmetric control weighting matrix: NC by NC
S	Symmetric state weighting matrix: NS by NS
$\hat{S}$	Symmetric state weighting matrix, classtwo: NS by NS
$\bar{S}$	Equivalent allstate weighting matrix: NS by NS
W	Bilinear weighting matrix: NS by NC
$\hat{W}$	Bilinear weighting matrix, class two: NS by NC
$\bar{\Phi}_K$	Closed loop state transition matrix: NS by NS

Vectors

$\underline{c}$	State initial condition vector: NS
$\underline{g}$	Weighting vector
$\hat{\underline{k}}$	Feedback gain vector
$\underline{p}$	Costate or multiplier vector: NS
"$\underline{P}$"	Equivalent Ricatti vector: NP
$\underline{u}$	Control vector: NC
$\underline{x}$	State vector: NS

Scalars

J	Cost index
V	Notational scalar

Chapter III

COMPUTATIONAL CONSIDERATIONS

3.1 Introduction

If the theory of the preceding chapters is to be of any practical use, efficient computational procedures should be available. Even for modest problems, most of the modern control theory techniques tax even the amazing capabilities of state of the art digital computers. One of the goals of this work was to develop a design technique with reduced computational requirements. This technique should be programable on almost any digital facility and might be very useful as a time-share library routine. Hopefully, the procedure would have low execution times and would be easy to use. In this chapter, numerical methods are developed for the SOC problem with these properties.

SOC has a decided advantage over other optimal schemes, since the structure of the necessary condition equations leads to reduced computational effort. There are four main considerations.

Point 1: The two point boundary value problem has been eliminated.

Comment: The Ricatti matrix has been used to decouple the two point boundary value problem of the necessary conditions. That problem has been replaced with a system of simultaneous nonlinear differentialor algebraic equations.

Point 2: The structure of the necessary conditions is independent of the size and complexity of the system.

Comment: The necessary conditions of an equivalent parameter optimization problem are a system of nonlinear equations which must be solved to obtain the optimal feedback gains. The structure of the equations becomes more and more complicated as the size and complexity of the system increases. Moreover there is no systematic way to formulate these equations. In contrast, the well defined SOC necessary conditions have a quadratic structure which is independent of the size of the problem.

Point 3: The SOC feedback gains are independent of the state initial conditions.

Comment: This fact is clear from the structure of the SOC necessary conditions. The feedback gains are a function of the Ricatti matrix which is independent of the state as a result of the decoupling of the two point boundary value problem. Thus, the feedback gains comprise a control law which is optimal for all initial conditions. Since most other schemes generate control laws which depend on the initial conditions, a suitable choice of design initial conditions must be made. In some cases, attempts have been made to develop a systematic procedure for picking a design initial condition vector. These procedures usually involve a large amount of computational effort.

Point 4: There exist efficient numerical methods for the solution of the SOC equations.

Comment: For almost all problems with NS larger than two, it is impossible to obtain an analytical solution to the Ricatti equation. There are two basic numerical approaches. The finite time interval and steady state problems may be solved by numerical integration of the Ricatti differential equation or the steady state problem may be solved by the direct solution of the steady state Ricatti equation.

3.2 Solution by Numerical Integration

Since the two point boundary value problem was decoupled by the Ricatti transformation, the finite time interval problem may be solved by straightforward numerical integration. The Ricatti equation is integrated backwards in time from t_f to obtain the Ricatti matrix which is used to calculate the feedback gains. Then the dynamics are integrated in the forward time direction to simulate the system trajectory.

The integration approach may be used to calculate the solution to the steady state problem, although not in a straightforward manner. The SOC index contains weighting matrices, $\hat{W}$ and $\hat{S}$, which are functions of the unknown steady state Ricatti matrix. Thus integration of the unreduced Ricatti equation is impossible. However the reduced Ricatti differential equation may be used. Note that this equation is not equivalent to the unreduced SOC Ricatti equation. This is clear since $\hat{W}$ of the unreduced equation is a function of the steady state Ricatti matrix while $\hat{W}$ corresponding to the reduced equation is a function of the time varying Ricatti matrix. However if a steady state solution of the reduced equation exists, this matrix will also be a solution to the steady state unreduced SOC Ricatti equation. The general conditions for existence of the solution to the reduced differential equation have not been established, although numerical evidence suggests that the solution of most steady state SOC Ricatti equations may be obtained by the solution of the corresponding reduced Ricatti equation. This may be a moot point since the next section describes the direct solution of the steady state equation by iterative means. This approach is usually more effective than numerical integration from accuracy and execution time considerations.

3.3 Iterative Solution of the Steady State Equation

The direct solution of the allstate Ricatti equation has been proposed by various authors.[40,41,42,43] For the most part these methods can be extended to the SOC problem. The concepts of some of these methods are described briefly and an extension of one of the more promising is derived. In addition, a new method applicable to the allstate as well as the SOC equations is proposed.

MacFarlane[40] and Bass[41] have developed procedures which require calculation of eigenvalues. To determine these eigenvalues is not a trivial task especially for large systems. Blackburn[42] introduced a procedure based on the Newton Raphson method. See Appendix B for a brief description of the Newton Raphson (N.R.) concept. The Blackburn algorithm involves the direct application of the N.R. approach to the algebraic Ricatti equation. In a similar way this approach can be applied to the reduced steady state SOC equations.

$$A^TP + PA + S - EQ^{-1}E^T + I_2EQ^{-1}E^TI_2 + \frac{W}{2} Q^{-1}E^TI_1 + I_1EQ^{-1} \frac{W^T}{2} = 0 \qquad (3.3.1)$$

where

$$E = PB + \frac{W}{2}$$

$$I_2 W = 0$$

The major drawback of this algorithm is that an initial guess for the Ricatti matrix must correspond to a set of stable gains. That is, if P^o is the initial guess then $(A - BQ^{-1}B^TP^o)$ must be stable. In most cases it is a difficult task to find a suitable value of P^o.

Recently Kleinman[43] introduced an algorithm which is also a Newton Raphson method. However, the structure of this algorithm is different from that of the usual N.R. approach and it possesses regional rather than local convergence properties. Moreover, only a set of stable gains is required to initialize this method. With a little effort, the Kleinman method may be extended to the SOC problem. However, the Kleinman-SOC algorithm must be started with a P^o corresponding to stable gains. This algorithm is to be preferred over the Blackburn algorithm since the implementation of the former is somewhat simpler and for the allstate case it does not require the knowledge of a stable P^o.

The basic concept of the Kleinman algorithm involves the simplification of the Newton Raphson algorithm by recognizing certain properties of the Ricatti equation. Consider the allstate Ricatti equation

$$F(P) = A^T P + PA + S - PBQ^{-1} B^T P = 0$$

or in terms of the closed loop system matrix

$$F(P) = A_K^{\ T} P + PA_K + S + KQK^T = 0 \qquad (3.3.2)$$

and the recursive relation defining the standard Newton Raphson method in function space is,

$$P^{i+1} = P^i - \left(\frac{dF}{dP}\right)^{-1}\Bigg|_{P=P^i} F(P^i) \qquad (3.3.3)$$

where the $\left(\frac{dF}{dP}\right)^{-1}$ indicates the inverse of the differential matrix, $\frac{dF}{dP}$. That is, if

$$dF = \frac{dF}{dP}\, dP$$

then

$$dP = \left(\frac{dF}{dP}\right)^{-1} dF$$

To derive this matrix, take the total differential of $F(P)$.

$$dF = A^T\, dP + dPA - dPBQ^{-1} B^T P - PBQ^{-1} B^T\, dP$$

Since for the allstate problem $K^T = Q^{-1} B^T P$ and $A_K = A - BK^T$ this equation may be rewritten

$$dF = A_K^T \, dP + dP \, A_K$$

or

$$\frac{dF}{dP} = (A_K^T * I + I * A_K) \qquad (3.3.4)$$

where $*$ indicates the Kronecker product. Thus

$$\left(\frac{dF}{dP}\right)^{-1} = (A_K^T * I + I * A_K)^{-1}$$

and the inverse exists if A_K is stable. Equation (3.3.3) may be rewritten as

$$P^{i+1} = P^i - (A_{K^i}^T * I + I * A_{K^i})^{-1}(A_{K^i}^T P^i + P^i A_{K^i} + S + K^i Q K^{i^T})$$

By definition

$$(A_{K^i}^T * I + I * A_{K^i})^{-1} (A_{K^i}^T P^i + P^i A_{K^i}) = P^i$$

Thus the Kleinman recursive equation is obtained

$$P^{i+1} = - \left(\frac{dF}{dP}\right)^{-1}\Bigg|_{P=P^i} (S + K^i Q K^{i^T})$$

or

$$A_{K^i}^T P^{i+1} + P^{i+1} A_{K^i} = - S - K^i Q K^{i^T}$$

Using this same concept, a similar algorithm can be formulated for the SOC Ricatti equation. However, in this case, P^i is not eliminated from the recursive relation. Thus a P^o corresponding to stable gains is required to start the Kleinman-SOC procedure. Write the SOC Ricatti equation in terms of P and let

$$I_2 W = 0$$

$$F(P) = PA + A^T P + S - I_1 (PB + \frac{W}{2}) Q^{-1} (B^T P + \frac{W^T}{2}) I_1$$

$$- I_1 (PB + \frac{W}{2}) Q^{-1} B^T P I_2$$

$$- I_2 PBQ^{-1} (B^T P + \frac{W^T}{2}) I_1$$

$$+ \frac{W}{2} Q^{-1} (B^T P + \frac{W^T}{2}) I_1 + I_1 (PB + \frac{W}{2}) Q^{-1} \frac{W^T}{2} = 0$$

Taking the total differential

$$dF = dPA + A^T dP - I_1 (dPB) Q^{-1} (B^T P + \frac{W^T}{2}) I_1 - I_1 (PB + \frac{W}{2}) Q^{-1} (B^T dP) I_1$$

$$- I_1 (dPB) Q^{-1} B^T P I_2 - I_1 (PB + \frac{W}{2}) Q^{-1} (B^T dP) I_2$$

$$- I_2 (dPB) Q^{-1} (B^T P + \frac{W^T}{2}) I_1 - I_2 PBQ^{-1} (B^T dP) I_1$$

$$+ \frac{W}{2} Q^{-1} (B^T dP) I_1 + I_1 (dPB) Q^{-1} \frac{W^T}{2} = 0$$

This equation can be written in terms of the closed loop system.

$$dF = dP A_K + A_K^T dP - I_1 dPB Q^{-1} (B^T P I_2 - \frac{W^T}{2})$$

$$- (I_2 PB - \frac{W}{2}) Q^{-1} B^T dP I_1$$

or

$$dF = H dP$$

where

$$H = I * A_K + A_K^T * I - I_1 I * B Q^{-1} (B^T P I_2 - \frac{W^T}{2})$$

$$- (I_2 PB - \frac{W}{2}) Q^{-1} B^T * I I_1$$

The Newton Raphson recursive relation may be written as

$$P^{i+1} = P^i - H^{-1} F(P^i)$$

Anticipating that the desired structure is

$$P^{i+1} = - H^{-1} G(P^i)$$

rewrite $F(P_i)$ as follows

$$P^i A_{Ki} + A_{Ki}^T P^i + S + K^i Q K^{i^T} = 0$$

or adding zero

$$F(P^i) = P^i A_{Ki} + A_{Ki}^T P^i + S + K^i Q K^{i^T} - D^i + D^- - D^{i^T} + D^{i^T}$$

where

$$D^i = I_1 P^i BQ^{-1}(B^T P^i I_2 - \frac{W^T}{2})$$

From the definition of H

$$P^{i+1} = P^i - H^{-1} F(P^i) = P^i - P^i - H^{-1}(S + K^i Q K^{i^T} + D_i + D_i^T)$$

Thus

$$P^{i+1} = - H^{-1} G(P^i)$$

where

$$G(P^i) = S + K^i QK^{i^T} + D^i + D^{i^T}$$

If $I_2 = 0$, which is true for the allstate problem, this algorithm reduces to Kleinman's algorithm. To summarize, the Kleinman-SOC algorithm is an application of the Newton Raphson concept to the solution of the SOC Ricatti equation. An initial guess for the Ricatti matrix corresponding to a set of stable gains is required. For convenience the method is implemented in terms of the equivalent vector from, "P", and require a single solution of a system of NP linear equations for each iteration.

$$NP = \frac{NS(NS + 1)}{2}$$

A New Algorithm

The proposed algorithm is unique in that the Ricatti equation is not solved directly. Instead a feedback gain equation is solved for the gains with the Ricatti matrix acting as a constraint relating the feedback gains and the Ricatti matrix. The steady state SOC equations are given below.

$$A_K = A - BK^T \tag{3.3.5}$$

$$A_K^T P + PA_K + S + KQK^T = 0 \tag{3.3.6}$$

$$Q^{-1}(B^T P + \frac{W^T}{2})\ I_1 - K^T = 0 \tag{3.3.7}$$

The SOC Ricatti equation is used to find P as a function of K which leads to a gain equation in terms of K. It will be convenient to formulate the matrix equations in terms of a vector equation. Recall that the notation $\ulcorner \underline{D} \urcorner$ indicates a vector formed by the column ordering of the matrix D.

$$\underline{F}(\ulcorner K \urcorner) = \ulcorner Q^{-1}(B^T P(K) - \frac{W^T}{2})\ I_1 \urcorner - \ulcorner \underline{K} \urcorner = 0 \tag{3.3.8}$$

This notation is slightly redundant since the gain functions corresponding to unavailable state gains are identically zero. This equation is solved by Newton Raphson iteration. With this approach the reduction in the number of equations to be solved may be significant. The direct solution of the Ricatti equation requires that $\frac{NS(NS+1)}{2}$ nonlinear equations be solved while the proposed algorithm requires the solution of (NS-L)NC equations. For example, if NS = 7, L = 5, and NC = 1 there are 28 unknown Ricatti elements and only

2 unknown gains. An additional advantage of this new scheme is that only a stable set of gains is required to start the method.

The recursive relation defining the algorithm is given by

$$^{\square}\underline{K}^{\square\, i+1} = {}^{\square}\underline{K}^{\square\, i} - \left\{ \nabla_{\square K \square} F({}^{\square}K^{\square\, i}) \right\}^{-1} \underline{F}({}^{\square}K^{\square\, i}) \tag{3.3.9}$$

where $\nabla_{\square K \square} F$ represents the Jacobian matrix of first derivatives such that

$$\nabla_{\square K \square} F \;\; d\,{}^{\square}\underline{K}^{\square} = d\underline{F}$$

The central concept of the algorithm concerns finding P as a function of K and calculating the Jacobian. Manipulations may be carried out more conveniently in terms of the equivalent vector equations. Recall that "P" represents a NP element vector found from P as follows:

$$"P"^T = (P_{1,1}; \dots P_{NS,1}; P_{2,2}; \dots ; P_{NS,NS})$$

The Ricatti equation may be rewritten in vector form,

$$E \text{ "P"} = - \text{"}(S + KQK^T)\text{"}$$

where

$$E = \text{"}(A_K^{\;T} * I + I * A_K)\text{"}$$

and

$$\text{"P"}(K) = - E^{-1} \text{"}(S + KQK^T)\text{"}$$

The inverse of E exists as long as A_K does not have two eigenvalues, λ_i, λ_j such that $\lambda_i + \lambda_j = 0$.[44] Now

$$\frac{\partial \text{"P"}}{\partial\, {}^{\square}K_j^{\square}} = - \frac{\partial E^{-1}}{\partial\, {}^{\square}K_j^{\square}} \text{"}(S + KQK^T)\text{"} - E^{-1} \frac{\partial \text{"}(S+KQK^T)\text{"}}{\partial\, {}^{\square}K_j^{\square}}$$

where $\left(\frac{\partial \text{"P"}}{\partial K_j}\right)_{\ell}$ is the partial derivative of the ℓ th element of "$\underline{P}$" with respect to the j^{th} element of $^{\square}K^{\square}$. Since

$$\frac{\partial E^{-1}}{\partial\, ^{\square}K_j^{\square}} = - E^{-1} \frac{\partial E}{\partial\, ^{\square}K_j^{\square}} E^{-1}$$

$$\frac{\partial \text{"P"}}{\partial\, ^{\square}K_j^{\square}} = E^{-1} \frac{\partial E}{\partial\, ^{\square}K_j^{\square}} E^{-1} \text{"}(S + KQK^T)\text{"} - E^{-1} \frac{\partial}{\partial\, ^{\square}K_j^{\square}} (\text{"}KQK^T\text{"}) \qquad (3.3.10)$$

Thus

$$\nabla_{^{\square}K^{\square}} F = \left(\frac{\partial F}{\partial\, ^{\square}K_1^{\square}}, \ \ldots \ , \frac{\partial F}{\partial\, ^{\square}K^{\square}_{NC(NS-L)}}\right)$$

where

$$\frac{\partial F}{\partial\, ^{\square}K_j^{\square}} = {^{\square}}\left(Q^{-1} B^T \frac{\partial P}{\partial\, ^{\square}K_j^{\square}} I_1\right)^{\square} - \frac{\partial\, ^{\square}K^{\square}}{\partial\, ^{\square}K_j^{\square}}$$

The term $\frac{\partial P}{\partial\, ^{\square}K_j^{\square}}$ is calculated in vector form $\frac{\partial \text{"P"}}{\partial\, ^{\square}K_j^{\square}}$ and then manipulated into the matrix form.

At each iteration two basic tasks must be performed. In the first

$$\frac{\partial \text{"P"}}{\partial\, ^{\square}K_j^{\square}} \qquad 1 \leq j \leq NC(NS-L)$$

is calculated by solving $NC(NS-L)$ systems of NP linear equations, all with the same left hand side. This is significant since after the initial solution, there is very little effort involved in solving additional systems with the same coefficient matrix. Note that Eq. (3.3.10) can be rewritten as

$$\frac{\partial \text{"P"}}{\partial\, ^{\square}K_j^{\square}} = -E^{-1} \left(\frac{\partial E}{\partial\, ^{\square}K_j^{\square}} \text{"P"} + \frac{\partial}{\partial\, ^{\square}K_j^{\square}} \text{"}(KQK^T)\text{"}\right)$$

To calculate this matrix, first solve the following system of NP linear equations for "P".

$$E \text{ "P"} = - \text{"}(S + KQK)\text{"}.$$

Since $\frac{\partial E}{\partial \underline{K}_j}$ is independent of $\underline{K}$ it may be computed once and stored.

Secondly, the vectors, $\frac{\partial \text{"P"}}{\partial \underline{K}_j}$ are found by solving

$$E \frac{\partial \text{"P"}}{\partial \underline{K}_j} = \frac{\partial E}{\partial \underline{K}_j} \text{"P"} - \frac{\partial}{\partial \underline{K}_j}\left(\text{"}KQK^T\text{"}\right)$$

This involves the solution of NC(NS-L) additional linear systems all with the same coefficient matrix. With this data, the Jacobian of the gain function equation may be formulated.

The second phase of each iteration involves the computation of the gain perturbations by the solution of a system of NS - L linear equations

$$\nabla_{\underline{K}^i} F \cdot \Delta \underline{K} = - F(\underline{K}^i)$$

followed by the calculation of the new values of the gains

$$\underline{K}^{i+1} = \underline{K}^i + \Delta \underline{K}$$

Thus, to execute one iteration of this algorithm, NC(NS-L) + 1 systems of $\frac{NS(NS+1)}{2}$ order linear equations all with the same coefficient matrix and a linear system of NS - L equations must be solved. This new method has been called the SOCDES algorithm since it plays a role in the SOC design procedure described in the next chapter.

3.4 Comparison of Algorithms

To solve the finite time varying problem numerical integration must be used. It has been found that the simpler algorithms such as Fourth order Runge Kutta give more satisfactory results than some of the more sophisticated methods such as Hamming predictor corrector or the Bulirsch - Stoer technique. Care must be taken when using these methods since an improper choice of integration step size or other algorithm parameters may lead to excessive execution times or erroneous results.

For the steady state problem, it is usually advisable to follow the iterative path. If a suitable initial guess can be found, then the iterative techniques have faster execution times and a simple control over the accuracy of the results. Of these procedures the Kleinman SOC or SOCDES methods appear to be superior. The former requires an initial guess for the Ricatti matrix corresponding to stable gains while SOCDES needs only the stable gains. Since Kleinman SOC has a simpler structure, execution time per iteration is less than that of SOCDES. However, it has been found that SOCDES usually converges in a fewer number of iterations. Thus even if a suitable starting value for the Kleinman SOC method is known, it may be more efficient to use SOCDES especially for the many practical problems in which the number of feedback gains is small with respect to the number of Ricatti elements. For example, a third order SOC problem with one feedback gain was solved in 12 seconds by SOCDES, 18 seconds by Kleinman SOC and 150 seconds by Runge Kutta integration.

Nomenclature

Matrices

A	System matrix: NS by NS
A_K	Closed loop system matrix: NS by NS
B	Control coefficient matrix: NS by NC
E	Equivalent vector equation coefficient matrix: NP by NP
$\bar{E}$	Notational matrix
$\frac{dF}{dP}$	Differential matrix: NS by NS
I_1	Notational matrix
I_2	Notational matrix
K	Feedback gain matrix: NS by NC
P	Ricatti matrix: NS by NS
Q	Symmetric control weighting matrix: NC by NC
S	Symmetric state weighting matrix: NS by NS
W	Bilinear weighting matrix: NS by NC
$\nabla_{\ulcorner K \urcorner} F$	Jacobian matrix: NC.NS by NC.NS

Vectors

$\ulcorner \underline{K} \urcorner$	Feedback gain vector: NC.NS
"P"	Equivalent Ricatti vector: NP
$\frac{\partial \text{"P"}}{\partial \ulcorner K_j \urcorner}$	Partial derivative vector: NP

Chapter IV

THE UNAVAILABLE STATE SOC DESIGN PROCEDURE

4.1 Introduction

The basic SOC theory and computational considerations have been examined in previous chapters. It has been shown that the optimal control law of the SOC problem is linear feedback with only the available states fed back. In addition efficient numerical procedures are available for the calculation of these control laws. The theory and the numerical methods are tied together to form a design procedure which may be useful for the study of realistic unavailable state problems.

To apply these techniques to a problem, a state variable representation of the systems must be obtained. From a block diagram or differential equations describing the system a set of first order linear differential equations of the following form is determined.

$$\dot{\underline{x}} = A\,\underline{x} + B\,\underline{u}$$

where $\underline{x}$ is the state of the system and $\underline{u}$ the control or input vector. This model should be formulated so that the last L states of the state vector are the unavailable or unmeasurable variables. Note that in many cases an engineering decision is made as to which states are available. That is, there may exist sensors which can measure some of the unavailable states, but for economic or other reasons it may be decided to assume that these states are unavailable.

In addition, the control law structure and design specifications or goals must be determined. Some of the specifications might include closed loop stability, an inherent property of SOC, a maximum peak value of one or more of the states to a particular input, and a well damped initial condition response.

A SOC cost index is formulated and S and Q are chosen to model the design specification. This choice of S and Q is somewhat arbitrary since some of the specifications are not explicitly represented in the quadratic index. However, previous work has shown that the use of the quadratic index leads to systems which are satisfactory with respect to the classical specifications of overshoot, damping, etc. After the initial SOC problem has been solved, the response of the system is compared with the design requirements. In some cases, this initial design may be unsatisfactory. Then the weightings are changed in a logical manner so as to correct the unacceptable features of the current design. The SOC problem is solved and again the response is evaluated. This concept is different from the usual trial and error procedure for two reasons. First, the interpretation of SOC as an optimal control problem removes some of the art from the design process. At each step, the new weighting are chosen in a systematic manner rather than in an intuitive manner. For example if the peak or integral square values of the states are too large than the state weighting would be increased and or the control weighting decreased in order to reduce this state error. The choice of the perturbation in the weighting matrices is discussed in a more precise way in section 4.3. Second, the whole procedure may be programmed to run automatically on a digital computer. Thus in a short time a number of designs can be made and evaluated allowing the engineer to gain insight into the problem.

It is possible that after a careful evaluation of the system, through the application of SOC, no satisfactory design is found. This may indicate that the design specifications are inconsistent with respect to the system and the chosen control structure. Then the control structure, the system, or the design specifications may be changed and the design procedure repeated. This approach is not an elixir but it has been found to be a very useful tool for the study and design of linear control systems.

4.2 SOC Design Procedure

In this section an explicit systematic procedure for the design of control laws based on the concepts of section 2.1 is proposed. The central concept is to use the Reverse SOC problem to obtain an initial set of weighting matrices. These weightings are perturbed in a systematic manner to obtain a more satisfactory design. For each set of weightings the SOC equations are solved by numerical integration for finite time interval problems and by SOCDES for the steady state problem. A digital computer program SOCSES I based on this method has been developed. See Appendix C for the description, flow chart and listing of the program. The reduced running time and user effort compared with other optimal design control programs, indicate that SOCDES I may be a very useful design tool. In Chapter VII SOC is applied to the problem of controlling a large flexible launch vehicle.

In Fig. 4.1, a block diagram of the method is shown and it is described below.

Step 1: Determine System Specifications

Comment: Based on the problem to be solved a reasonable set of specifications must be determined. SOCDES I may be helpful in pointing out inconsistent requirements.

Step 2: Select a Control Configuration

Comment: As indicated above, the unavailable states must be specified. In addition, compensation in the form of a filter or network may be used. It may be considered as part of the system to be controlled and some of its parameters may be chosen by feeding back some of the filter states.

Step 3: Solve the Reverse SOC Problem

Comment: For the finite time interval problem any set of finite continuous feedback gains may be used in the solution of the Reverse problem. However, for the steady state problem a stable set of gains must be obtained. For the many physical systems which are stable, zero gains are sufficient. For those that are unstable it is usually not very difficult to generate a set of gains with stability as the only criterion. Even for the complex booster of Chapter VII, a calculation of the Routh array leads to a stable set of gains.

Note that the existence and uniqueness properties of Chapter II and the convergence properties of the iterative schemes of Chapter III are of a local nature. SOCDES I may be used to extend these properties to a region. For example if during the design procedure the Jacobian disappears or the equations become numerically difficult to solve, it is

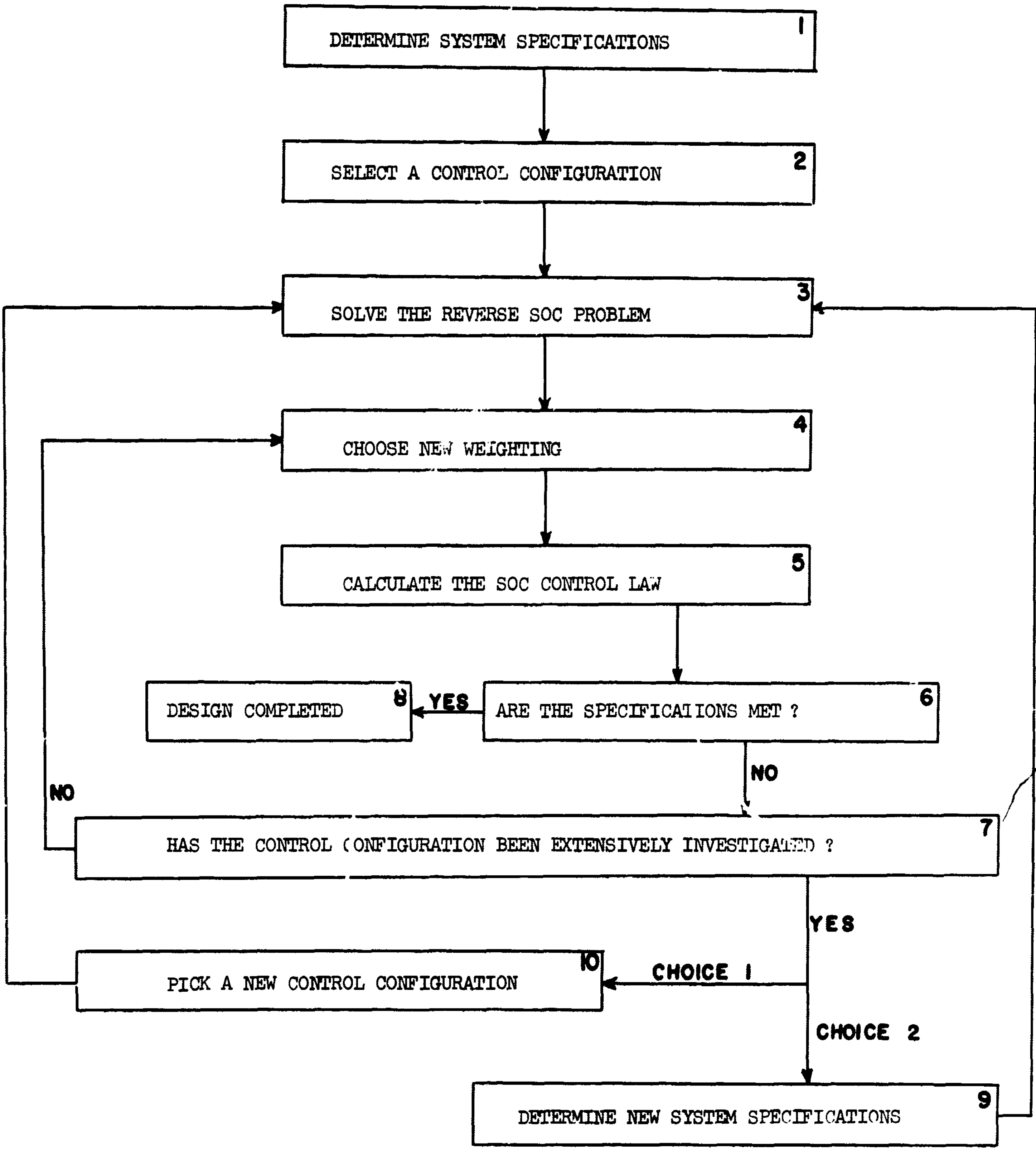

DETERMINE SYSTEM SPECIFICATIONS 1
SELECT A CONTROL CONFIGURATION 2
SOLVE THE REVERSE SOC PROBLEM 3
CHOOSE NEW WEIGHTING 4
CALCULATE THE SOC CONTROL LAW 5
ARE THE SPECIFICATIONS MET ? 6
YES
DESIGN COMPLETED 8
NO
HAS THE CONTROL CONFIGURATION BEEN EXTENSIVELY INVESTIGATED ? 7
NO
YES
CHOICE 1
PICK A NEW CONTROL CONFIGURATION 10
CHOICE 2
DETERMINE NEW SYSTEM SPECIFICATIONS 9

FIG. 4.1

possible to resolve the reverse problem and thus define a new neighborhood of existence and convergence which allows the design process to be continued.

Step 4: Choose New Weightings

Comment: See Section 4.3

Step 5: Calculate the SOC Control Law

Comment: The new SOC problem is solved by one of the numerical techniques of Chapter 3.

Step 6: Are the Specifications Met?

Comment: The current design is checked to see if the design specifications are met. This may include simulation of the closed loop system or other calculations such as finding the closed loop poles. If specifications are met, the design is complete; if not the design procedure is continued.

Step 7: Has the Control Configuration Been Extensively Investigated?

Comment: If the current control configuration has been carefully examined and no satisfactory design has been obtained then two choices are available. First, the analysis done so far may point out a new set of feasible specifications. Second, a new control structure may be chosen. This might include a new choice of available states or the use of a different compensator. Once a choice is made the design returns to step 3 and the cycle continues. Since the computational effort involved in implementing this procedure is low it may be feasible to examine various configurations and compare the results. In this way it may be possible to gain insight into the choice of a "best" controller configuration.

4.3 Systematic Choice of Perturbation Weighting Matrices

After each iteration in the SOC design procedure new weightings must be chosen to improve the design. A tradeoff between system error and control effort can be obtained by varying the relative magnitudes of S and Q. Intuitive reasoning indicates that by increasing the state weighting, S, the integral state error will decrease while increasing the control weighting, Q, will lead to reduced values of the integral square control effort. Since the control law is of a closed loop nature, the integral square values of control effort and state error are related. Assume that the state weighting is increased. In general, this will cause the magnitude of the feedback gains to increase and the state error to decrease. The control effort may increase or decrease corresponding to the relative magnitudes of these changes. These inituitive concepts have been substantiated by numerous examples. Moreover, it is possible to derive an expression which indicates the effect of perturbing the weighting matrices.

Given an expression which represents the properties of interest, say $\hat{J}_x = \int_{t_o}^{t_f} \underline{x}^T \underline{x} \; dt$, then determine the gradient of this expression with respect to the weightings. Again let $\underline{g}$ represent a weighting vector formed from the independent elements of S, W, and Q. Then the perturbation $d\,\hat{J}_x$ due to weighting changes is given by

$$d\,\hat{J}_x = \frac{\partial J_x}{\partial g}^T d\,\underline{g}$$

where $\frac{\partial \hat{J}_x}{\partial \underline{g}}$ is a vector such that $\left[\frac{\partial \hat{J}_x}{\partial \underline{g}}\right]_i = \frac{\partial \hat{J}_x}{\partial g_i}$

This approach is not restricted to integral square quantities and may be applied to any design characteristics which can be represented by a mathematical expression. Moreover, an indication of the consistency of the design requirements may be obtained. Form a vector composed of NN design specification expressions, $\hat{J}_i$, $1 \leq i \leq NN$.

$$\hat{\underline{J}} = \begin{bmatrix} \hat{J}_1 \\ \cdot \\ \cdot \\ \cdot \\ \hat{J}_{NN} \end{bmatrix}$$

Then calculate the gradient of this vector with respect to the weighting vector.

$$d\,\hat{\underline{J}} = \nabla_{\underline{g}}\,\hat{\underline{J}}^T\, d\underline{g} \qquad (4.3.1)$$

where

$$\nabla_{\underline{g}}\,\hat{J} = \left[\frac{\partial \hat{J}_1}{\partial \underline{g}}, \ldots, \frac{\partial \hat{J}_{NN}}{\partial \underline{g}}\right]$$

If for a particular design specification change, $d\hat{J}$, a solution to Eq. (4.3.1) exists, then the change is consistent and may be obtained with $d\underline{g}$ as the weighting change.

Consider $\hat{J}_x = \int_{t_o}^{\infty} \underline{x}^T \underline{x} \; dt$

and for simplicity assume a scalar control and the corresponding gain vector $\underline{k}$ of N elements. Using the chain rule

$$\frac{\partial \hat{J}_x}{\partial \underline{g}} = \nabla_g \underline{k} \; \frac{\partial \hat{J}_x}{\partial \underline{k}}$$

where $\nabla_g k = \left[\frac{\partial k_1}{\partial \underline{g}}, \; \dots , \; \frac{\partial k_N}{\partial \underline{g}} \right]$

The vectors, $\frac{\partial k_i}{\partial \underline{g}}$, can be calculated easily using the SOC necessary condition equations.

$$k^T = Q^{-1} (B^T P + \frac{W^T}{2}) I_1$$

$$A_k^T P + P A_k + S + \underline{k} Q \underline{k}^T = 0$$

or using the equivalent vector notation

$$"P" = - E^{-1} "(S + \underline{k} Q \underline{k}^T)"$$

Note that the weighting elements enter into these equations in a simple manner leading to easy calculations.

The calculation of $\frac{\partial J_x}{\partial \underline{k}}$ is not as trivial a matter, since a straightforward approach is not feasible. However, by interpreting J_x

as a cost index and using Lemma 2 of Chapter II it is possible to determine these terms. Let J_x be a function of the lower time integration limit and require that J_x have a quadratic representation.

$$J_x(t) = \underline{x}(t)^T D\underline{x}(t) = \int_t^{\infty} \underline{x}^T S\underline{x} \; dt$$

where S = I and D is a constant matrix to be determined. Differentiating with respect to t leads to

$$\frac{d J_x}{dt} = \dot{\underline{x}}^T Dx + x^T D\dot{x} = - x^T Sx$$

Since $\dot{x} = A_k \underline{x}$

$$\underline{x}^T A_k^T D\underline{x} + \underline{x}^T D A_k \underline{x} = - x^T Sx \qquad (4.3.2)$$

Since Eq. (4-3.2) is required to hold for all $\underline{x}$.

$$A_k^T D + D A_k = - S = - I \qquad (4.3.3)$$

If A_k is stable, Eq. (4.3.3) may be solved for D and expressed in the vector notation as

$$"D" = -E^{-1} "I"$$

Thus

$$\frac{\partial "D"}{\partial k_i} = - \frac{\partial E^{-1} "I"}{\partial k_i} = E^{-1} \frac{\partial E}{\partial k_i} E^{-1} "I"$$

For an initial condition vector $\underline{c}$

$$J_x = \underline{c}^T D \underline{c}$$

and

$$\frac{\partial J_x}{\partial k_i} = \frac{\partial(\underline{c}^T D \underline{c})}{\partial k_i} = \underline{c}^T \frac{\partial D}{\partial k_i} \underline{c}$$

where $\frac{\partial "D"}{\partial k_i}$ can be calculated and manipulated to obtain $\frac{\partial D}{\partial k_i}$.

Note that Eq.(4.3.4) appears in the SOCDES algorithm and thus it would be easy to calculate these gradients and implement the automatic choice of new weightings.

Using this approach the intuitive effect of varying the weightings may be verified for the second order example. The pertinent data for this example is given below.

$$A_k = \begin{bmatrix} -2\zeta\omega^{-k} & -\omega^2 \\ 1 & 0 \end{bmatrix}$$

$$b = \begin{bmatrix} 1 \\ 0 \end{bmatrix}; \quad S = \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix}; \quad W = \begin{bmatrix} W_1 \\ 0 \end{bmatrix}$$

Let $\zeta = 0$ and $\omega = 1$ then from Eq. (2.3.10)

$$k = \frac{W_1}{2q} + \frac{1}{q}\sqrt{S_1 q + \frac{W_1^2}{4} + S_3 q} \tag{4.3.5}$$

The parameter vector $\underline{g}$ has the following components

$$\underline{g} = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ W_1 \\ q \end{bmatrix}$$

and let

$$\hat{J}_x = \int_{t_o}^{\infty} \underline{x}^T \underline{x} \; dt$$

Using the chain rule

$$\frac{\partial \hat{J}_x}{\partial \underline{g}} = \frac{\partial \hat{J}_x}{\partial k} \frac{\partial k}{\partial \underline{g}}$$

where

$$\frac{\partial k^T}{\partial \underline{g}} = \left[\frac{\partial k}{\partial S_1} , \frac{\partial k}{\partial S_2} , \frac{\partial k}{\partial S_3} , \frac{\partial k}{\partial W_1} , \frac{\partial k}{\partial q} \right]$$

and from Eq. (l.3-5)

$$\frac{\partial k}{\partial S_1} = \frac{1}{2\sqrt{S_1 q + \frac{W_1^2}{4} + S_3 q}}$$

$$\frac{\partial k}{\partial S_2} = 0$$

$$\frac{\partial k}{\partial S_3} = \frac{\partial k}{\partial S_1}$$

$$\frac{\partial k}{\partial W_1} = \frac{1}{2q}\left[1 + \frac{W_1}{4\sqrt{S_1 q + \frac{W_1^2}{4} + S_3 q}}\right]$$

$$\frac{\partial k}{\partial q} = \frac{-W_1}{2q^2} - \frac{1}{q^2}\sqrt{S_1 q + \frac{W_1}{4} + S_3 q} + \frac{1}{2q}\,\frac{S_1 + S_3}{\sqrt{S_1 q + \frac{W_1^2}{4} + S_3 q}}$$

Using the definition of A_K and the fact that

$$E = \text{"}(A_K^T * I + I * A_K)\text{"}$$

it is easily shown that for this example

$$E = \begin{bmatrix} -2k & 2 & 0 \\ -1 & -k & 1 \\ 0 & -2 & 0 \end{bmatrix}$$

If this notation is bothersome, recall that this matrix may be obtained by writing the Ricatti matrix as a system of three scalar equations and simply identifying the coefficients as shown in Appendix E.

Now

$$E^{-1} = \begin{bmatrix} -\frac{1}{2k} & 0 & -\frac{1}{2k} \\ 0 & 0 & -\frac{1}{2} \\ -\frac{1}{2k} & 1 & -\frac{1}{2}\left(k + \frac{1}{k}\right) \end{bmatrix}$$

and

$$"I" = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

Thus

$$\frac{\partial "D"}{\partial k} = - \frac{\partial E^{-1}}{\partial k} "I" = \begin{bmatrix} -\frac{1}{2k^2} & 0 & -\frac{1}{2k^2} \\ 0 & 0 & 0 \\ -\frac{1}{2k^2} & 0 & \frac{1}{2}(1-\frac{1}{k^2}) \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

or if $D = \begin{bmatrix} D_1 & D_2 \\ D_2 & D_3 \end{bmatrix}$

Thus

$$\frac{\partial D_1}{\partial k} = - \frac{1}{k^2}$$

$$\frac{\partial D_2}{\partial k} = 0$$

$$\frac{\partial D_3}{\partial k} = - \frac{1}{k^2} + \frac{1}{2}$$

Thus

$$\frac{\partial J_x}{\partial k} = \frac{\partial (c^T D\, c)}{\partial k} = - \frac{{c_1}^2}{k^2} + \left(\frac{1}{2} - \frac{1}{k^2}\right) {c_2}^2$$

where $\underline{x}(t_o) = \underline{c} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$

Consider the specific values of the weightings used in the example of Chapter II which were

$$S_1 = S_3 = 1, \quad S_2 = 0, \quad q = 1, \quad W_1 = 0$$

which lead to the optimal SOC feedback gain,

$$k = \sqrt{2}$$

Assume that

$$c = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Then

$$\frac{\partial \hat{J}_x}{\partial S_1} = \frac{\partial \hat{J}_x}{\partial S_3} = \frac{\partial \hat{J}_x}{\partial k} \quad \frac{\partial k}{\partial S_3} = - \frac{1}{2} \cdot \frac{\sqrt{2}}{4} = - \frac{\sqrt{2}}{8}$$

$$\frac{\partial \hat{J}_x}{\partial q} = \frac{\partial \hat{J}_x}{\partial k} \quad \frac{\partial k}{\partial q} = - \frac{1}{2} \cdot - \frac{\sqrt{2}}{2} = \frac{\sqrt{2}}{4}$$

Hence the intuitive notions are verified since increases in the state weighting, S_1 or S_3 , causes the state error $\hat{J}_x$ to decrease, while increasing the control weighting, q , causes the state error to increase.

Calculations of this type may be made to verify or establish the effect of perturbing the weightings. It is possible to systematically vary the weightings based on this information to obtain changes in the design characteristics and hence proceed to a acceptable design in a logical manner. If desired this gradient procedure could be added to the SOCDES I program and the weighting perturbations could be calculated automatically.

Moreover, the SOCDES I program could be used as a computational method for solving other optimal problems. Suppose that it was desired to minimize

$$\hat{J}_x = \int_{t_o}^{\infty} \underline{x}^T \underline{x} \, dt$$

for some set of initial conditions. Then SOCDES I could be used as a gradient procedure to solve this optimal problem and possibly avoid some of the formulations and numerical difficulties of the parameter optimization approach.

4.4 Design Loci

One of the most useful of the classical techniques for the study of single input single output time invariant systems is the root locus. Essentially it is a graphical technique which plots the loci of the closed loop poles (system eigenvalues) as a function of the loop gain. This technique provides insight as well as explicit information about the behavior of the system. A similar procedure has been proposed for SOC through the use of the SOCDES I program. This technique involves determining the loci of the closed loop poles as a function of the weighting matrices. The SOCDES I program is used to solve the SOC problems for various values of the weightsings. For each step, the characteristic equation is solved and the poles obtained. Thus these poles as a function of the weightings are plotted.

Another locus which has been of use is the gain locus which involves the plotting of the feedback gains as a function of the weightings. From this locus, the gradient of the gains with respect to the weightings may be obtained and used to determine the effect of the weightings on the design criterion.

As an example of these loci again consider the second order example. Let

$$\gamma = 0, \quad \omega = 1, \quad S_1 = S_3 = 1, \quad q = 1, \; W_1 = 0,$$

then

$k = \sqrt{2}$ and the characteristic equation is given by

$$\det \begin{bmatrix} S + k & -1 \\ 1 & S \end{bmatrix} = S^2 + k\,S + 1 = 0$$

Since $k = \sqrt{2}$

$$S = -\frac{\sqrt{2}}{2} \pm j\,\frac{\sqrt{2}}{2}$$

Consider the loci of these roots and the gain as S_1 is varied which is plotted in Fig. 4.2a and 4.2b. As S_1 is increased the gain increases and the poles approach the real axis. As S_1 is decreased, k approaches one and the roots approach $-\frac{1}{2} \pm j\frac{\sqrt{3}}{2}$. In a similar manner consider the loci as a function of the control weighting, q. Then the trend is in the opposite direction since as q is increased the gain decreases and the poles approach the imaginary axis. By varying q it is possible to obtain all stable values of the feedback gain. See Fig. 4.3a and 4.3b.

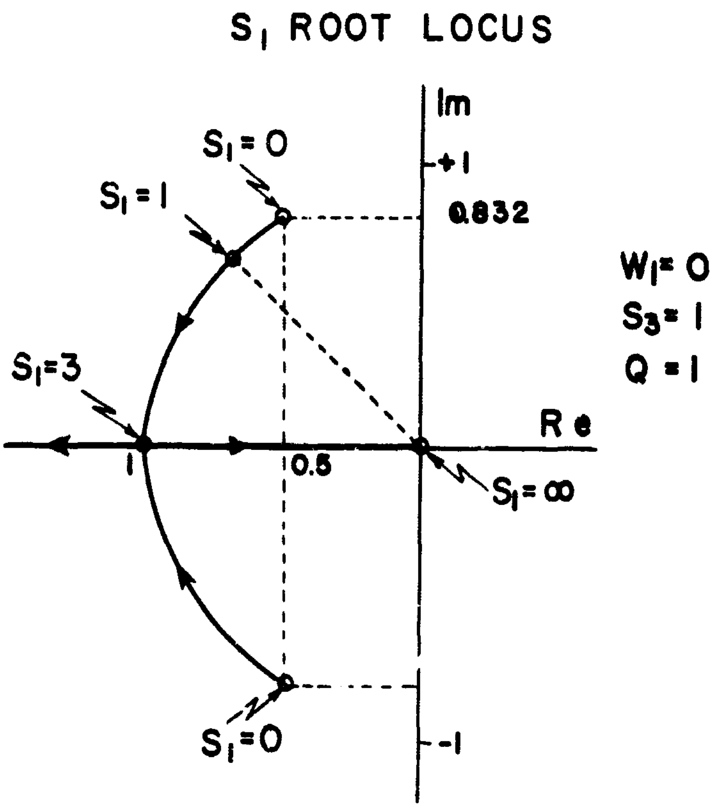


FIG. 4.2-a

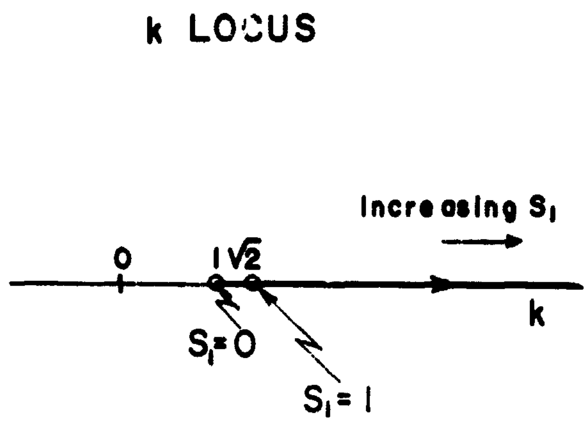


FIG. 4.2-b

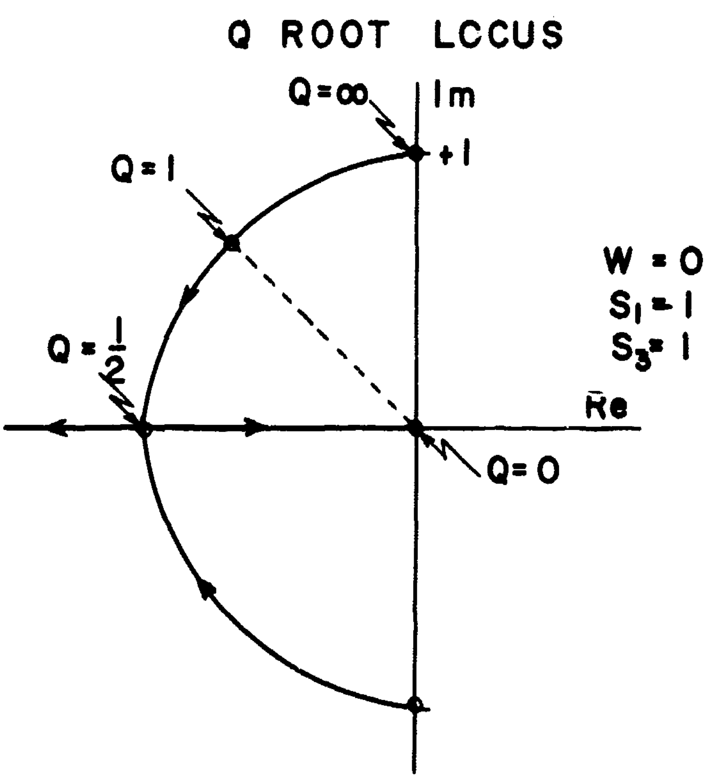


FIG. 4.3-a

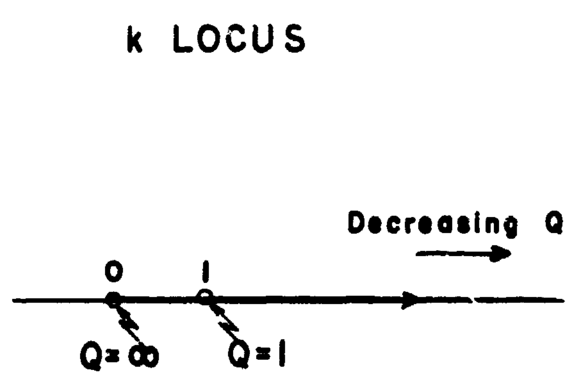


FIG. 4.3-b

Note that the S_1 and q root loci coincide where they both exist. Since various combinations of weightings may give the same gain it is possible to get identical root loci for different weighting sets.

Another useful tool is the graphical representation of the feedback gains corresponding to stable poles. Since the poles are a continuous function of the gains it is possible to plot the set of stable gains $\mathcal{K}$ in some region of a Euclidean space. Then the K locus may be plotted on the same graph. For a problem with two gains,

$$\underline{k} = \begin{pmatrix} k_1 \\ k_2 \end{pmatrix}$$

a typical plot is shown in Fig. 4.4

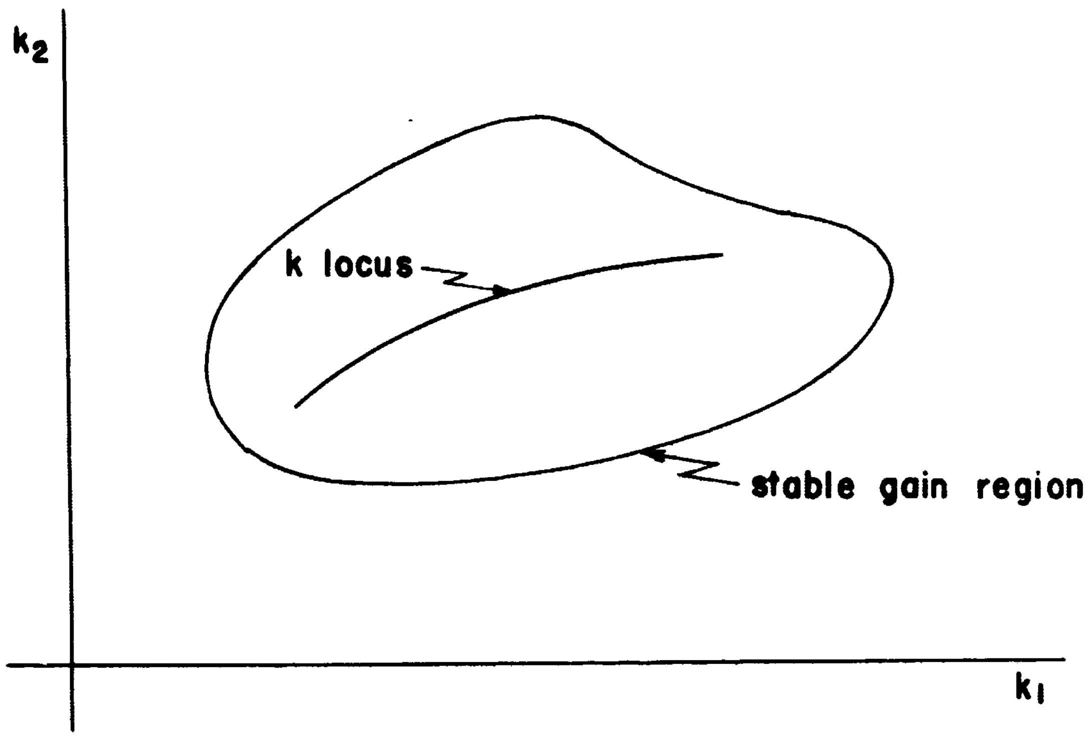


FIG. 4.4

Nomenclature

Matrices

A	System matrix: NS by NS
B	Control coefficient matrix: NS by NC
D	Notational matrix
E	Coefficient matrix for equivalent vector equation: NP by NP
K	Feedback gain matrix: NS by NC
P	Ricatti matrix: NS by NS
Q	Symmetric control weighting matrix: NC by NC
S	Symmetric state weighting matrix: NS by NS
$\frac{\partial D}{\partial k_i}$	Matrix of partial derivatives
$\nabla_{\underline{g}} \underline{\hat{J}}$	Jacobian matrix of $\underline{J}$ with respect to g
$\nabla_{\underline{g}} \underline{k}$	Jacobian matrix of $\underline{k}$ with respect to $\underline{g}$

Vectors

"D"	Vector equivalent of D
"I"	Vector equivalent of I
$\underline{g}$	Weighting vector
$\underline{\hat{J}}$	Vector of design criterion
$\underline{k}$	Vector of feedback gains
"P"	Vector equivalent of P
$\underline{x}$	State vector: NS
$\underline{u}$	Control vector: NC

Scalars

$\hat{J}_{\underline{x}}$	Design criterion expression

Chapter V

THE SOC MODEL REFERENCE PROBLEM

5.1 Introduction

In order to design a control system, a mathematical abstraction or model of the process to be controlled must be obtained. In any practical situation this model is only an approximate representation of the actual process. The effectiveness of the control system designs depends to a large extent on the accuracy of this model.

Once the model has been chosen and the nominal design completed, additional design factors must be considered. These factors include the effect of possible environmental changes, such as additive disturbances or plant parameter variations. Many of the current design techniques allow the consideration of additive disturbances; however, the plant parameter variations are not as easy to handle. These plant parameter variations may be of two types; there may be actual changes in the plant caused by component aging or the parameter estimates for the model may be inaccurate. For this study, the term variations does not refer to changes with time but rather to the fact that the constant parameters have unknown off-nominal values.

In recent years, the plant parameter problem has been attacked by sensitivity methods and by the model reference approach. The objective of these control schemes is to cause the trajectory of the system to remain close to the nominal in spite of plant parameter variations. The model reference scheme does this by attemping to null the error between the actual trajectory and the ideal nominal trajectory generated by a model. Note that the term "model" has been used in two different ways. The first usage referred to the mathematical description of a physical process while the second referred to a "black box" which may or may not have a physical realization and which generates the desired nominal system trajectory.

The scheme proposed in this section consists of two feedback loops. The inner loop is designed with the aid of conventional or optimal techniques on the basis of the assumed nominal process model in order to obtain satisfactory response to command inputs in the presence of additive disturbances. The outer feedback loop is designed with the SOC technique to compensate for inaccuracies in the process model parameters as well as any additive disturbances. An advantage of the model reference approach over that of trajectory sensitivity, is that the nominal model reference trajectory may be chosen independently of any sensitivity considerations, while the sensitivity approach involves a tradeoff between the nominal trajectory and sensitivity. The model reference approach pays for this advantage with increased controller complexity.

To be more specific, consider the regulator control problem of driving the output of a system to zero. The following development is easily extended to the more general case of a non-zero command input. In Fig. 5.1 the model reference scheme is pictured. The inner feedback gain matrix, K_o , is designed on the basis of the nominal process model. In the outer loop, the control is obtained by feeding back the difference between the actual system output and that of the output of the model.

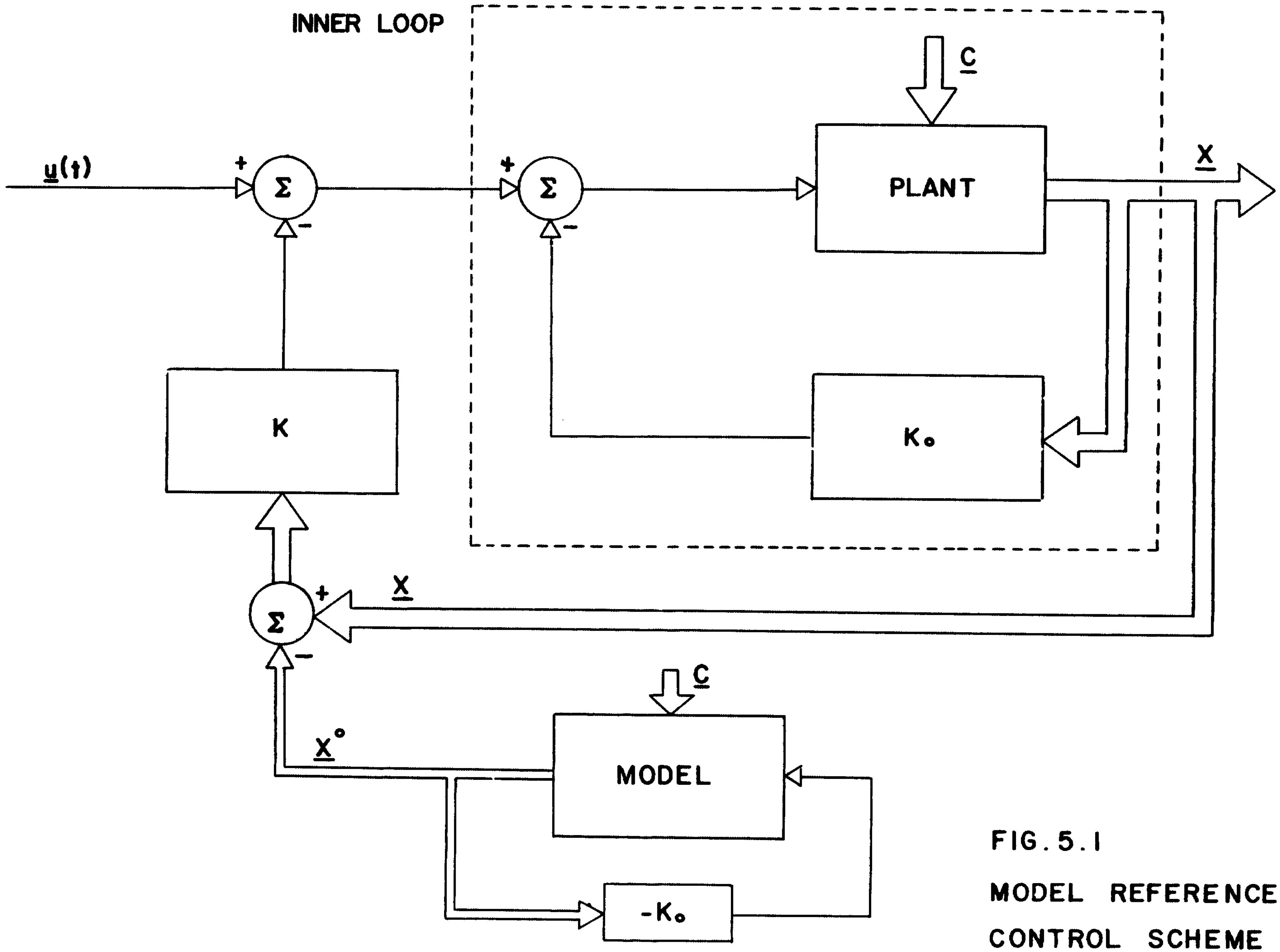


FIG. 5.1
MODEL REFERENCE
CONTROL SCHEME

The outer loop gain matrix, K , is found by the application of the SOC procedure. It is shown that this gain matrix depends on the process model and desired response characteristics but that it does not depend on the nominal trajectory. This property may be of practical significance. Consider the Saturn launch booster problem. Aside from the difficulties involved in generating an accurate process model, the actual flight of the vehicle is subject to severe additive wind disturbances. For a particular flight, the guidance command is a function of the mission requirements and the wind patterns, hence nominal trajectories vary from one flight to the next. Using this model reference scheme it is possible to precompute the feedback gains and hence the control law and then simply change the model input based on the nominal trajectory.

5.2 Formulation of the SOC Model Reference Problem

Previous sections have described the SOC theory and considered its application to control problems with unavailable states. The design of the outer control loop in order to keep the system trajectory close to the model reference trajectory in spite of parameter variations can be formulated as an unavailable state problem.

Assume that the inner control loop and the command input, which for the regulator problem is zero, are given. Consider the effect of parameter variations on the system trajectory. A perturbation model which describes this effect can be obtained by the linearization of the plant and the nominal feedback control about the nominal trajectory and parameter values. The parameter variations are considered as additional state variables and the SOC theory is used to determine a linear feedback control law that does not depend on the parameter states. Since the gains operate on the perturbed states, in the actual implementation they operate on the difference between the actual and nominal trajectories as shown in Fig. 5.1. Strictly speaking the analysis applies to small perturbations, although in many cases it has been found that the SOC model reference scheme gives satisfactory control for a wide range of parameter values.

Derivation of the Perturbation Model

Let the nominal process model with a inner loop control and a command input be described by a linear system of differential equations.

$$\dot{\underline{x}} = (A(\underline{q}^o) - B\,K_o^T)\,\underline{x}^o + B\,\underline{m}\ ; \quad \underline{x}(t_o) = \underline{c} \tag{5.2.1}$$

where q^o is a vector of NPA parameters and K_o is the matrix of inner loop gains. The superscript ^{o} indicates a nominal quantity.

By expanding Eq.(5.2.1) about the nominal parameter vector and trajectory, an expression for the differential equation system describing the off nominal trajectory can be obtained.

$$\dot{\underline{x}} = \dot{\underline{x}}^o + (A(\underline{q}^o) - B\,K_o^T)\,d\underline{x} + \sum_{\ell=1}^{NPA} \frac{\partial A}{\partial q_\ell}\,\underline{x}^o\,dq_\ell + B\,d\underline{m} + 0^2 \tag{5.2.2}$$

where $\frac{\partial A}{\partial q_\ell}$ denotes the matrix of partial derivatives evaluated at the nominal,

$$\left[\frac{\partial A}{\partial q_\ell}\right]_{ij} = \frac{\partial [A]_{ij}}{\partial q_\ell}$$

and 0^2 denotes second and higher order terms.

If the perturbations are suitably small, the higher ordered terms may be neglected and a linear model is obtained.

$$d\dot{x} = \dot{x}-\dot{x}^o = (A(q^o)-B\,K_o^T)\,dx + \sum_{\ell=1}^{NPA} \frac{\partial A}{\partial q_\ell}\,\underline{x}^o\,dq_\ell + B\,d\underline{m}$$

The SOC problem is formulated in terms of an augmented state vector.

$$\underline{y} = \begin{bmatrix} d\underline{x} \\ \hline d\underline{q} \end{bmatrix}$$

The dynamics which describe this state vector are obtained from the linear perturbations model and the fact that the parameter vector is assumed to be time invariant.

$$\dot{\underline{y}} = \bar{A}\,\underline{y} + \bar{B}\,\underline{u}\ ; \quad \underline{x}(t_o) = \underline{c} \qquad (5.2.3)$$

where

$$\bar{A} = \left[\begin{array}{c|cccc} A - BK_o^T & A_{q_1}\underline{x}^o & \cdots & A_{q_{NPA}} & \underline{x}^o \\ 0 & 0 & \cdots & 0 & \\ \vdots & \vdots & & \vdots & \\ 0 & 0 & \cdots & 0 & \end{array}\right]$$

$$A_{q_i} = \frac{\partial A}{\partial q_i}$$

$$\bar{B} = \begin{bmatrix} B \\ \hline 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\underline{\bar{c}} = \begin{bmatrix} 0 \\ --- \\ d\,\underline{q} \end{bmatrix}$$

$$\underline{u} = d\,\underline{m}$$

The upper elements of the initial condition vector are zero since it is assumed that the perturbations in the parameters do not effect the system initial conditions.

Formal Statement of SOC Problem

The SOC control $\underline{u}$ is structured so that the unavailable perturbation states and the parameter states are not fed back. This control, $\underline{u}$, is chosen to minimize J.

$$J = \frac{1}{2} \int_{t_o}^{t_f} (\underline{y}^T S\, \underline{y} + \underline{y}^T \hat{S}\, \underline{y} + \underline{y}^T W\, \underline{u} + \underline{y}^T \hat{W}\, \underline{u} + \underline{u}^T Q\, \underline{u})\, dt$$

subject to the dynamics of Eq. (5.2.3). Since $\underline{x}^o$ is a function of time, $\bar{A}$ is; and it would appear that this is a time varying SOC problem. Thus the SOC feedback gains would be time varying and would be characterized as follows.

$$\bar{K}^T(t) = Q^{-1} (\bar{B}^T P(t) + \frac{W^T}{2}) I_1$$

$$-\dot{P}(t) = \bar{A}_{\hat{K}}^T P + P \bar{A}_{\hat{K}} + S + \bar{K} Q \bar{K}^T \; ; \; P(t_f) = 0$$

where

$$\bar{K} = \begin{bmatrix} K \\ ---- \\ 0 \end{bmatrix}$$

$$\bar{A}_{\hat{K}} = \left[\begin{array}{c|ccc} A - B(K^T + K_o^T) & A_{q_1}\,\underline{x}^c & \cdots & A_{q_{NPA}}\,\underline{x}^o \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \end{array}\right]$$

However, a close examination of these equations indicates that the gains are independent of A_{q_i} and x^o ! This surprising result implies that insofar as the linearization model is accurate the model reference scheme compensates for any parameter variation around any nominal trajectory. Moreover, although the SOC problem has been formulated as a time varying problem, constant values of the model reference gains can be found by considering a time invariant process model and inner feedback gains and a SOC index terminal time of ∞ .

To demonstrate this result a matrix partitioning notation will be used. For convenience assume that there are two parameters and let

$$\bar{A}_{\hat{K}} = \left[\begin{array}{c|c|c} A - B(K^T + K_o^T) & A_{q_1}\,\underline{x}^o & A_{q_2}\,\underline{x}^o \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right]$$

$$S = \begin{bmatrix} S_1 & S_2^T & S_3^T \\ S_2 & S_4 & S_5^T \\ S_3 & S_5 & S_6 \end{bmatrix}$$

$$W = \begin{bmatrix} W_1 \\ \hline 0 \\ \hline 0 \\ \hline \end{bmatrix}$$

$$P = \begin{bmatrix} P_1 & P_2^T & P_3^T \\ P_2 & P_4 & P_5^T \\ P_3 & P_5 & P_6 \end{bmatrix} \qquad (5.2.4)$$

With this notation the matrix differential Ricatti equation can be decomposed and written in terms of six component equations.

$$K^T = Q^{-1} \; (B^T P_1 + \frac{W_1}{2}) \; I_* \qquad (5.2.5)$$

where $I_* = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ is NS by NS

and I is a NS-L by NS-L identity matrix. The last L states of the original state vector are assumed to be unavailable.

$$A_{\hat{K}} = A - B(K^T + K_o^T) \qquad (5.2.6)$$

$$-\dot{P}_1 = P_1 A_{\hat{K}} + A_{\hat{K}}^T P_1 + S_1 + K Q K^T \; ; \quad P_1(t_f) = 0 \qquad (5.2.6)$$

$$-\dot{P}_2^T = P_1 A_{q_1} \underline{x}^o + S_2^T + A_{\hat{K}}^T P_2^T \quad ; \quad P_2^T(t_f) = 0 \qquad (5.2.7)$$

$$-\dot{P}_3^T = P_1 A_{q_2} \underline{x}^o + S_3^T + A_{\hat{K}}^T P_3^T \quad ; \quad P_3^T(t_f) = 0 \qquad (5.2.8)$$

$$-\dot{P}_4 = P_2 A_{q_1} \underline{x}^o + \underline{x}^{o^T} A_{q_1}^T P_2^T + S_4 \quad ; \quad P_4(t_f) = 0 \qquad (5.2.9)$$

$$-\dot{P}_5^T = P_2 A_{q_2} \underline{x}^o + \underline{x}^{o^T} A_{q_1}^T P_3^T + S_5^T \quad ; \quad P_5(t_f) = 0 \qquad (5.2.10)$$

$$-\dot{P}_6 = P_3 A_{q_2} \underline{x}^o + \underline{x}^{o^T} A_{q_2}^T P_3^T + S_6 \quad ; \quad P_6(t_f) = 0 \qquad (5.2.11)$$

From Eq.(5.2.5) and (5.2.6) it is clear that the feedback gains depend only on P_1 which is independent of the other P partition blocks. Thus, the SOC gains are independent of A_{q_1}, A_{q_2} and $\underline{x}^o$. If the time invariant steady state problem is considered, the SOC model reference gains are determined from a algebraic matrix Ricatti equation.

$$P_1 A_{\hat{K}} + A_{\hat{K}}^T P_1 + S_1 + K Q K^T = 0 \qquad (5.2.12)$$

and

$$\dot{x} = (A-B(K_o^T + K^T))\, x \qquad (5.2.13)$$

The nominal composite closed loop system, $A_{\hat{K}}$, that is the system with feedback gains equal to the sum of the inner and outer loop gains is stable.

$$A_{\hat{K}} = A - B\,(K_o^T + K^T)$$

This can be shown by choosing the following Lyapunov function.

$$V = \underline{x}^T P_1 \underline{x}$$

where P_1 is positive definite and is obtained from the partitioned Ricatti matrix. From Eq.(5.2.12) and (5.2.13).

$$\dot{V} = - \underline{x}^T (S_1 + K Q K^T) \underline{x}$$

which is negative for any allowable system trajectory. With the application of the Lyapunov Stability Theorem, the result is obtained. If the parameter variations are suitably small so that the linearized dynamics are valid,

$$d\dot{\underline{x}} = (A(q_o) - B K_o^T) \, dx + \sum_{\lambda=1}^{NPA} \frac{\partial A}{dq_\lambda} \underline{x}^o \, dq_\lambda + B \, d\underline{m}$$

and $d\underline{m} = - K^T \, d\underline{x} = \underline{u}$

Or, in terms of the composite system matrix,

$$d\dot{\underline{x}} = A_K^\wedge \, d\underline{x} + \sum_{\ell=1}^{NPA} \frac{\partial A}{\partial q_\lambda} \underline{x}^o \, dq_\ell \quad ; \quad d\underline{x}\,(t_o) = \underline{0} \qquad (5.2.14)$$

Assume that the nominal trajectory is stable, then

$$\underline{x}^o \longrightarrow 0 \quad \text{as} \quad t \longrightarrow \infty$$

Since the composite system is stable, its state transition matrix $\phi_K^\wedge(t_1 t_o)$ approaches zero as t approaches infinity. If the last terms of Eq.(5.2.14) is considered as a forcing term, the trajectory despersion can be written as

$$d\underline{x}(t) = \int_{t_o}^{t} \sum_{\ell=1}^{NPA} \phi_K^\wedge (t, \tau) \frac{\partial A}{\partial q_\lambda} \underline{x}^o (\tau) \, dq_\lambda \, d\tau$$

and

$$\underline{x}(t) \approx \underline{x}^o(t) + d\underline{x}\,(t) \qquad (5.2.15)$$

Thus, the dispersion remains bounded since it can be shown that the integrand is bounded by some negative exponential.

From the block diagrams of Figure 5.1, it is seen that the differential equations describing the model reference system can be written in two ways

$$\dot{\underline{x}} = (A-B\,K_o^T)\,\underline{x} - B\,K^T(\underline{x} - \underline{x}^o)$$

or

$$\dot{\underline{x}} = (A-B\,(K_o^T + K^T)\,\underline{x} + B\,K^T\,\underline{x}^o$$

The solution for this second equation can be expressed as

$$\underline{x}(t) = \phi_{\hat{K}}(t, t_o)\,\underline{c} + \int_{t_o}^{t} \phi_{\hat{K}}(t, \tau)\,B\,K^T\,\underline{x}^o(\tau)\,d\tau \qquad (5.2.16)$$

From this viewpoint it is clear that the model reference system will remain stable as long as the parameter variations do not cause the composite system, $A_{\hat{K}}$, to become unstable. Note that Eq.(5.2.15) is an approximate relation derived from the linearized model which is used to calculate the outer loop, while Eq.(5.2.16) is an exact expression derived from the consideration of the model reference system block diagram.

A important feature of this model reference approach is the fact that the nominal response of the system, which is independent of the outer loop gains, may be designed to achieve the "best" system response without regard to parameter sensitivity considerations. Thus, the model reference gain, K^T, could be chosen so that the composite system is insensitive to parameter variations. If the parameters have nominal values, the "best" performance is obtained while if there are parameter variations, the response may deteriorate slightly but the entire system will remain stable.

Although most of the blocks of the Ricatti matrix do not effect the calculation of the feedback gains, they may provide useful information. Suppose that all of the blocks of the state weighting matrix, S except S_1 are chosen to be zero. Then the optimal index may be expressed as

$$J^o = \frac{1}{2} \int_{t_o}^{t_f} (\underline{dx}^T S_1 \underline{dx} + \underline{u}^T Q \underline{u}) \, dt$$

Using the definition of the control law and Lemma 2 of Chapter II it is possible to rewrite this equation as

$$J^o = \frac{1}{2} \begin{bmatrix} dx \\ \hline d_{\underline{q}} \end{bmatrix}^T P \begin{bmatrix} dx \\ \hline d_{\underline{q}} \end{bmatrix} = \frac{1}{2} \int_{t_o}^{\infty} (\underline{dx}^T (S_1 + K Q K^T) \underline{dx}) \, dt$$

The elements of P indicate the relative effect of the various parameters on the trajectory dispersion. The value of the cost index, which is an integral weighted square of the dispersion due to the parameter variations, $d\underline{q}$, can be found in terms of the Ricatti matrix elements. For example for the system and Ricatti matrix of (Eq. 5.2.4) the value of the index resulting from the variation dq_1 is given by

$$J_1^o = dq_1^2 \; P_4$$

Similarly, for a perturbation in q_2, dq_2

$$J_2^o = dq_2^2 \; P_6$$

With this information the designer has an indication of the relative effects of the various parameters. If J_1^o is large compared with

$J_2^{\,o}$, then it might be important to know the value of q_1 in a precise manner while q_2 might not have a significant effect on the system response.

5.3 Example

In order to illustrate the calculations and effectiveness of this model reference scheme, a second order damped oscillator example is considered. It is assumed that only the rate state is available. The model reference scheme is designed to compensate for lack of knowledge of the damping ratio, ζ . The differential equation defining the system is given below and the block diagram is shown in Fig. 2.1.

$$\ddot{x} + 2\zeta\omega\dot{x} + \omega^2 x = v(t)$$

Let $\zeta^o = 0$ and $\omega = 1$ and use as the nominal inner loop gain the SOC control law of the example of Chapter II.

$$k^o = \sqrt{2}$$

The formal model reference problem required the choice of the perturbation control $\underline{u}$ to minimize

$$J = \frac{1}{2}\int_{t_o}^{\infty} (\underline{y}^T S\, \underline{y}^T + \underline{y}^T \hat{S}\, \underline{y} + \underline{y}^T W u + \underline{y}^T \hat{W} u + u^2 Q)\, dt$$

subject to

$$\dot{\underline{y}} = \bar{A}\,\underline{y} + \bar{\underline{b}}\,u ; \ \underline{y}(t_o) = \bar{\underline{c}}$$

The augmented state vector is

$$\underline{y} = \begin{bmatrix} d\dot{x} \\ dx \\ d\zeta' \end{bmatrix}$$

and

$$\bar{A} = \begin{bmatrix} -2\zeta^{o}\omega - k_o & -\omega^2 & -2\omega\dot{x}^{o} \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\underline{\bar{b}} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\underline{\bar{c}} = \begin{bmatrix} 0 \\ 0 \\ d\zeta' \end{bmatrix}$$

$Q = q$, a scalar

The solution control law has the following structure.

$$u = -k\, d\dot{x}$$

Instead of calculating the formal problem, the reduced problem was solved for various weighting matrices. The equations which characterize this reduced problem are

$$K = Q^{-1} (B^T P + \frac{W^T}{2}) I_*$$

$$A_{\hat{K}}^T P_1 + P_1 A_{\hat{K}} + S_1 + qKK^T = 0 \qquad (5.3.1)$$

where

$$A_K^{\wedge} = \begin{bmatrix} -2 \zeta^{o} \omega - k - k^{o} & -\omega^2 \\ 1 & 0 \end{bmatrix}$$

$$P_1 = \begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix}$$

$$S_1 = \begin{bmatrix} s_1 & s_2 \\ s_2 & s_3 \end{bmatrix}$$

Eq.(5.3.1) can be written as an equivalent set of scalar equations and for convenience the values of ζ and ω have been substituted and it is assumed that $W = 0$.

$$k = \frac{p_1}{q}$$

$$2p_1 (-k_o - k) + 2p_2 = - s_1 - q k^2$$

$$-p_1 - k p_2 + p_3 = - s_2$$

$$-2p_2 = - s_3$$

or

$$k = - k_o + \frac{1}{q} \sqrt{q^2 k_o^2 + s_1 q + s_3 q}$$

These equations were solved for the following three sets of weighting matrices.

1) $S = \begin{bmatrix} 1.91 & 0 \\ 0 & 1.91 \end{bmatrix}$, $q = 1$, $W = 0$

$k = 1$

2) $S = \begin{bmatrix} 8.76 & 0 \\ 0 & 8.76 \end{bmatrix}$, $q = 1$, $W = 0$

$k = 3$

3) $S = \begin{bmatrix} 53.08 & 0 \\ 0 & 53.08 \end{bmatrix}$, $q = 1$, $W = 0$

$k = 9$

These solutions are compared by perturbing the parameter, γ, simulating the system and calculating

$$J_x = \int_{t_o}^{t} x(t)^2 \, dt$$

with t = 10 seconds. To provide a basis for comparison the system was simulated with only the inner loop control for the various values of parameters. The numerical integration was done with a fourth order Runge Kutta algorithm. Three off-nominal values of γ were examined and the resultsare logged in Table 5.1. Note that $\gamma = -1.664$ with the nominal gain alone corresponds to an unstable system as indicated by

J_x \ k		0	1.0	3.0	9.0
ζ	0	.354	.354	.354	.354
	-.414	.500	.448	.414	.380
	-1.164	1.960	.807	.562	.436
	-1.664	∞	1.501	.722	.480

$\zeta^\circ = 0$

$k^\circ = \sqrt{2}$

Table 5.1 $J_x(k,\zeta)$

the entry of ∞ in the table. As expected the value of J_x for an off nominal parameter decreases as the model reference gain increases. This corresponds to the tradeoff between state error and control effort. In Fig. 5.2, the simulation results for the nominal control and parameters are compared with an off-nominal parameter with inner loop control only, and the full model reference system. Note that the model reference scheme succeeds in keeping the trajectory close to the nominal in spite of the parameter variation. In Chapter VII this model reference scheme is applied to launch vehicle problem.

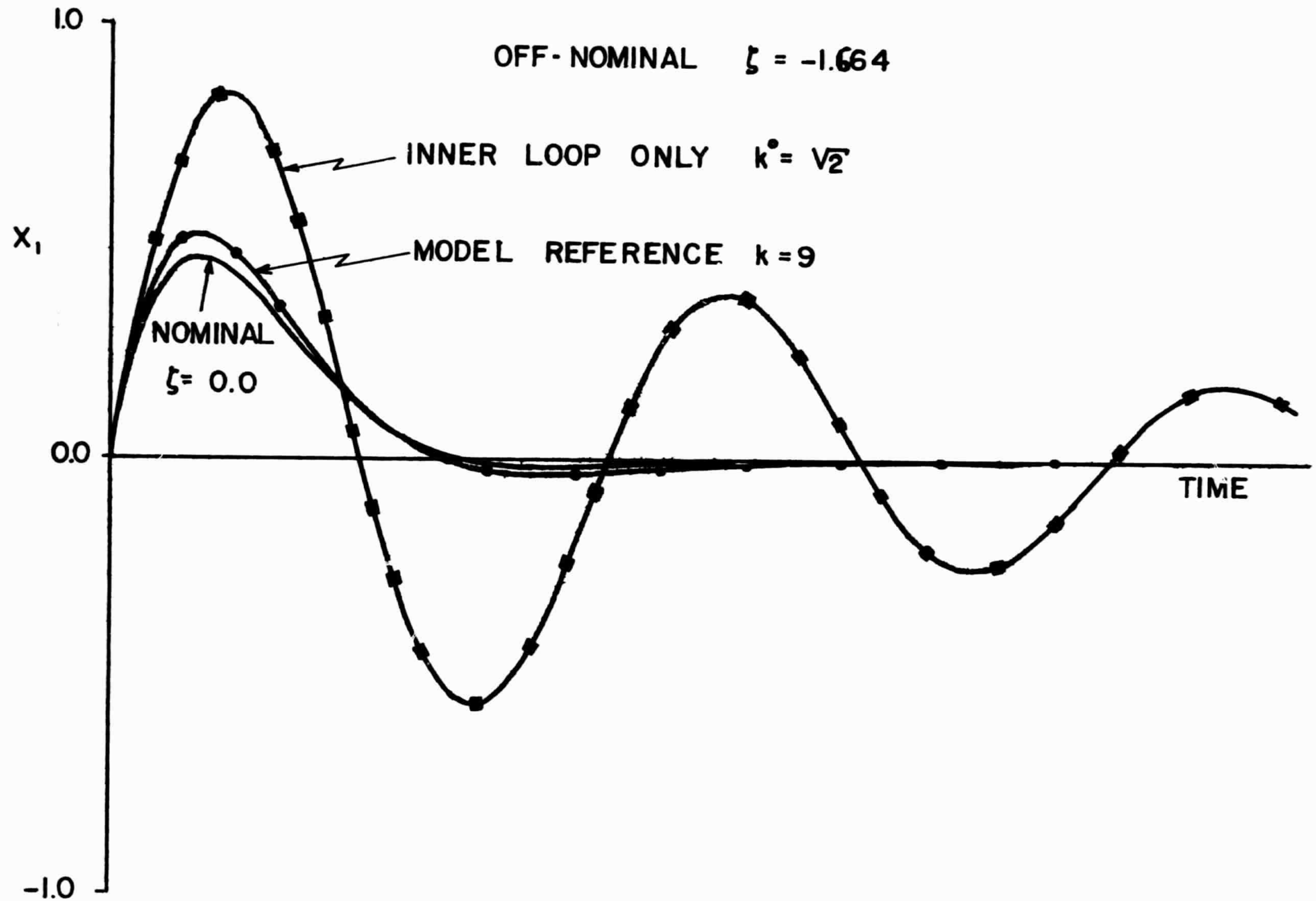


FIG. 5.2-a RATE RESPONSE x_1

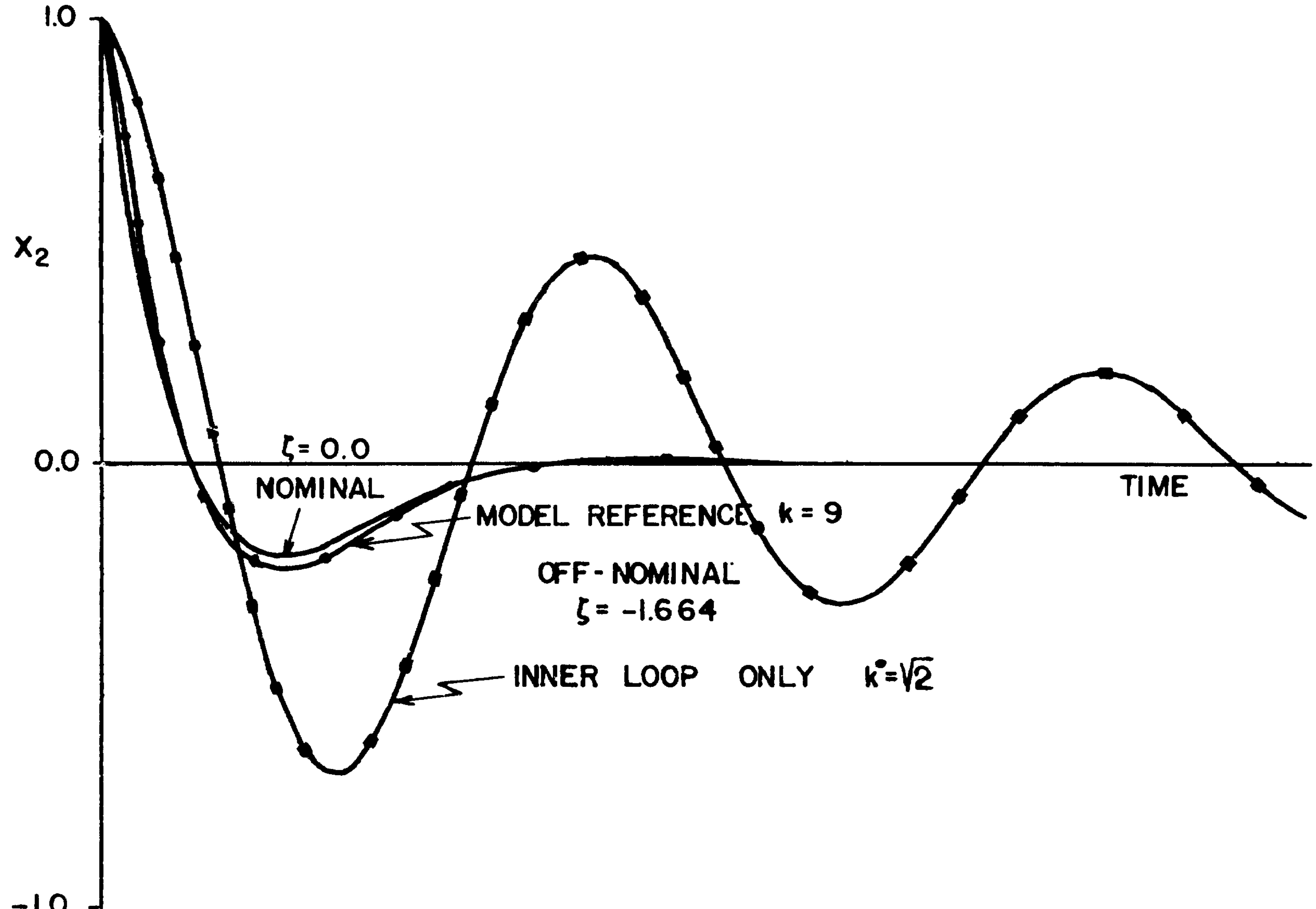


FIG. 5.2-b POSITION RESPONSE X_2

Nomenclature

Matrices

A	System matrix: NS by NS
$\bar{A}$	Perturbation system matrix: NPA + NS by NPA + NS
B	Control coefficient matrix: NS by NC
$\bar{B}$	Perturbation system control coefficient matrix: NS + NPA by NC
K	Model reference feedback gain matrix: NS by NC
K_o	Inner loop feedback gain matrix: NS by NC
$\hat{K}$	Composite feedback gain matrix: NS by NC
$\bar{K}$	Perturbation model feedback gain matrix: NS + NPA by NC
P	Ricatti matrix: NS by NS
Q	Symmetric control weighting matrix: NC by NC
S	Symmetric state weighting matrix: NS + NPA by NS + NPA
$\hat{S}$	Symmetric state weighting matrix, class two: NS + NPA by NS + NPA
S_1	Component matrix of S: NS by NS
W	Bilinear weighting matrix: NS + NPA by NC
$\hat{W}$	Bilinear weighting matrix: NS + NPA by NC
W_1	Component matrix of W: NS by NC
$\frac{A}{\partial q_\ell}$	Matrix of partial derivatives

Vectors

$\underline{c}$	Initial condition vector: NS
$\dot{\underline{c}}$	Perturbation model initial condition vector: NS + NPA
$\underline{m}$	System input vector: NC
$d\underline{m}$	Perturbation model control vector: NC
$\underline{q}$	Parameter vector: NPA
$d\underline{q}$	Perturbation parameter vector: NPA
$\underline{u}$	Perturbation model control vector: NC
$\underline{x}$	System state vector: NS
$\underline{y}$	Perturbation model state vector: NS + NPA

Chapter VI

THE SOC SENSITIVITY PROBLEM

6.1 Introduction

The concepts of optimal control have been applied to the problem of plant parameter sensitivity in order to calculate control schemes which are relatively insensitive. The basic concept is to define a variable which represents the sensitivity of the trajectory or cost index to changes in system parameters. These sensitivity variables are considered as additional state variables and are placed in the cost index to be minimized. Since most of the closed loop control laws of optimal control require knowledge of all of the state variables, the additional sensitivity states must be generated, adding to the complexity of the controller. It is clear that for a given feedback control structure, certain values of gains lead to less sensitive closed loop systems than others. Thus it appears feasible to formulate a SOC problem which determines a control law that does not feed back any sensitivity states, and yet allows a tradeoff between system error, sensitivity, and control effort. Using this approach feedback control laws may be designed with sensitivity considerations, rather than designing and then analyzing for sensitivity characteristics.

6.2 Problem Formulation

Previous work[34,35,36,45] has defined and developed the concept of trajectory sensitivity functions as outlined below. Assume that the state or trajectory of a system may be described by a system of first order linear differential equations, which are a function of a vector of constant parameters, $\underline{q}$.

$$\dot{\underline{x}}^o = A(\underline{q}^o)\, \underline{x}^o + B\, \underline{u}^o; \qquad \underline{x}(t_o) = \underline{c} \tag{6.2.1}$$

where the superscript o indicates the nominal. Consider the effect of a small change in the parameter on the system trajectory. The resulting off-nominal trajectory is described by the following system of differential equations.

$$\dot{\underline{x}} = A(\underline{q}^{o} + d\underline{q})\, \underline{x} + B\, \underline{u}; \qquad \underline{x}(t_o) = \underline{c} \; .$$

This trajectory may be represented by a Taylor series expansion about the nominal parameter.

$$\underline{x} = \underline{x}^{o} + \frac{\partial \underline{x}}{\partial \underline{q}}\, d\, \underline{q} + 0^{2}$$

where 0^2 represents second and higher order ter and $\frac{\partial \underline{x}}{\partial \underline{q}}$ is a matrix of partial derivatives.

$$\left[\frac{\partial \underline{x}}{\partial \underline{q}} \right]_{i,j} = \frac{\partial x_i}{\partial q_j}$$

Similarly the trajectory dispersion is given by

$$\Delta\, \underline{x} = \underline{x} - \underline{x}^{o} = \frac{\partial \underline{x}}{\partial \underline{q}}\, d\, \underline{q} + 0^{2} \qquad (6.2.2)$$

Assuming that the first order terms are sufficient to describe the trajectory dispersion, it is clear that for a given parameter perturbation the dispersion can be made small by limiting the magnitude of $\frac{\partial \underline{x}}{\partial \underline{q}}$. Thus, define the sensitivity matrix, Z, as follows.

$$Z = \frac{\partial \underline{x}}{\partial \underline{q}} \qquad \left[Z \right]_{i,j} = \frac{\partial x_i}{\partial q_j} \qquad \begin{matrix} 1 \le i \le NS \\ 1 \le j \le NPA \end{matrix} \qquad (6.2.3)$$

Let $\underline{z}_j$ denote the j^{th} sensitivity vector corresponding to the j^{th} parameter and the j^{th} column of Z. These sensitivity vectors are adjoined to the system state vector to form an augmented state vector.

$$\underline{z}_j = \begin{bmatrix} \frac{\partial x_1}{\partial q_j} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial x_{NS}}{\partial q_j} \end{bmatrix} ; \qquad \hat{\underline{x}} = \begin{bmatrix} \underline{x} \\ \underline{z}_1 \\ \cdot \\ \cdot \\ \cdot \\ \underline{z}_{NPA} \end{bmatrix} \tag{6.2.4}$$

The augmented state vector is to be placed in a SOC cost index; by appropriate choice of weighting matrices a tradeoff between system performance and sensitivity may be obtained. The formulation of the SOC problem requires that a differential equation describing the behavior of the state vector be known. Fortunately, such an equation may be easily derived. Since by assumption $\underline{q}$ is independent of time and the first order partial derivatives are continuous, the differential operators may be interchanged.

$$\dot{\underline{z}}_j = \frac{d\underline{z}_j}{dt} = \frac{d}{dt}\left(\frac{\partial \underline{x}}{\partial q_j}\right) = \frac{\partial}{\partial q_j}\left(\frac{d\underline{x}}{dt}\right) = \frac{\partial \dot{\underline{x}}}{\partial q_j} \tag{6.2.5}$$

Note that the partial derivatives are taken with respect to the nominal. Using Eq. (6.2.1) this expression becomes

$$\dot{\underline{z}}_j = \frac{\partial A}{\partial q_j}\underline{x} + \left(A + B\frac{\partial \underline{u}}{\partial \underline{x}}\right)\frac{\partial \underline{x}}{\partial q_j} ; \qquad \left.\frac{\partial \underline{x}}{\partial q_j} = \underline{0}\right|_{t=t_o}$$

or

$$\dot{\underline{z}}_j = \frac{\partial A}{\partial q_j}\underline{x} + \left(A + B\frac{\partial \underline{u}}{\partial \underline{x}}\right)\underline{z}_j ; \qquad \underline{z}_j(t_o) = \underline{0} \tag{6.2.6}$$

The initial conditions are zero, since the parameter variations have no effect on the systems initial conditions. If the control law is linear feedback,

$$\underline{u} = -K^T\underline{x}$$

then Eq. (6.2.6) reduces to

$$\dot{\underline{z}}_j = \frac{\partial A}{\partial q_j}\underline{x} + (A - BK^T)\,\underline{z}_j ; \qquad \underline{z}_j(t_o) = \underline{0} \tag{6.2.7}$$

The differential equation system describing the augmented state vector may be written in convenient state variable notation.

$$\hat{\underline{x}} = \begin{bmatrix} \underline{x} \\ \underline{z}_1 \\ \vdots \\ \underline{z}_{NPA} \end{bmatrix} \tag{6.2.8}$$

$$\dot{\hat{\underline{x}}} = \hat{A}\,\hat{\underline{x}} + \hat{B}\,\underline{u} ; \qquad \hat{\underline{x}}(t_o) = \begin{bmatrix} \underline{c} \\ \underline{0} \\ \vdots \\ \underline{0} \end{bmatrix} \tag{6.2.9}$$

where

$$\frac{\partial A}{\partial q_j} \triangleq A_{q_j}$$

$$\hat{A} = \begin{bmatrix} A & 0 & 0 & \cdot & \cdot & 0 \\ A_{q_1} & A-BK^T & 0 & & & \vdots \\ A_{q_2} & 0 & A-BK^T & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \ddots & \vdots \\ \vdots & \vdots & 0 & & \ddots & 0 \\ A_{q_{NPA}} & 0 & 0 & \cdot & 0 & A-BK^T \end{bmatrix}$$

$$\hat{B} = \begin{bmatrix} B \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Consider the problem of formulating a quadratic index in terms of the state and sensitivity variables and solving for the optimal control law. It is well known that the solution to the linear quadratic problem is a linear feedback controller. Note that the term, $\frac{\partial \underline{u}}{\partial \underline{x}}$, appears in the sensitivity differential equation. This term prevents the direct use of the linear approach since the necessary conditions defining the optimal solution are derived assuming that the $\hat{A}$ matrix is independent of the control and hence the feedback gains. Thus a straightforward application of the SOC concept is not possible since the gain matrix K^T appears in $\hat{A}$.

However, it is possible to reformulate the problem and remove this difficulty. Define a new control vector

$$\underline{\hat{u}} = \begin{bmatrix} \underline{u} \\ \underline{m}_1 \\ \vdots \\ \underline{m}_{NPA} \end{bmatrix} \tag{6.2.10}$$

Anticipating that the SOC control law is linear feedback, formulate the SOC sensitivity problem so that $\underline{u}$ and $\underline{m}_j$ have the following structure.

$$\underline{u} = - K^T \underline{x}$$

$$\underline{m}_j = - K^T \underline{z}_j \qquad 1 \leq j \leq NPA$$

where the NS by NC gain matrices in all the equations are required to be identical. This is a different application of SOC than was used in the unavailable state problem. In this case the gains are required to have equal but unknown values which will be determined by the solution of the SOC problem. In addition, the unavailable state property is used to insure that neither the unavailable states nor the sensitivity variables are fed back. Now, the dynamics may be rewritten.

$$\dot{\hat{\underline{x}}} = \bar{A}\,\hat{\underline{x}} + \bar{B}\,\hat{\underline{u}} \qquad (6.2.11)$$

where

$$\bar{A} = \begin{bmatrix} A & 0 & \cdot & \cdot & \cdot & 0 \\ A_{q_1} & A & 0 & \cdot & \cdot & 0 \\ \cdot & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ A_{q_{NPA}} & 0 & \cdot & \cdot & 0 & A \end{bmatrix}$$

$$\bar{B} = \begin{bmatrix} B & 0 & \cdot & \cdot & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \cdot & & \ddots & \ddots & \\ 0 & \cdot & \cdot & 0 & B \end{bmatrix}$$

The SOC sensitivity control law, $\underline{u}$, is chosen to minimize

$$J = \frac{1}{2} \int_{t_o}^{\infty} \left(\hat{\underline{x}}^T S\, \hat{\underline{x}} + \hat{\underline{x}}^T \hat{S}\, \hat{\underline{x}} + \hat{\underline{x}}^T W\, \hat{\underline{u}} + \hat{\underline{x}}^T \hat{W}\, \hat{\underline{u}} + \hat{\underline{u}}^T Q\, \hat{\underline{u}} \right) dt \qquad (6.2.12)$$

subject to the dynamics

$$\dot{\hat{\underline{x}}} = \bar{A}\,\hat{\underline{x}} + \bar{B}\,\hat{\underline{u}} \; ; \qquad \hat{\underline{x}}(t_o) = \begin{bmatrix} \underline{c} \\ \underline{0} \\ \vdots \\ \underline{0} \end{bmatrix} \qquad (6.2.13)$$

and the SOC structure constraints.

$$\underline{u} = - K^T \underline{x}$$

$$\underline{m}_j = - K^T \underline{z}_j$$

or

$$\hat{\underline{u}} = - \bar{K}^T \underline{x}$$

where

$$\bar{K}^T = \begin{bmatrix} K^T & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & K^T \end{bmatrix}$$

The selection of $\hat{S}$ and $\hat{W}$ and the derivation of the necessary conditions are described in Appendix D and summarized below.

<u>SOC Sensitivity Ricatti Equation</u>

$$A_{\bar{K}}^T P + PA_{\bar{K}} + S + \bar{K} Q \bar{K}^T = 0 \qquad (6.2.14)$$

where

$$A_{\bar{K}} = (A - B\bar{K}^T) : \quad (NPA + 1)NS \text{ by } (NPA + 1)NS$$

$$A_K = (A - BK^T) : \quad NS \text{ by } NS$$

and

$$A_{\bar{K}} = \begin{bmatrix} A & 0 & \cdots & \cdots & 0 \\ A_{q_1} & A_K & \ddots & & \vdots \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ A_{q_{NPA}} & 0 & \cdots & 0 & A_K \end{bmatrix}$$

Feedback Gain Equation

$$\tilde{K}^T = \begin{bmatrix} K^T & 0 & \cdot & \cdot & 0 \\ 0 & \cdot & & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & 0 \\ 0 & \cdot & \cdot & 0 & K^T \end{bmatrix}$$

where

$$K^T = Q^{-1}(B^T P + \frac{W^T}{2})\, I_{11}^1$$

and $I_{11}^1 = \begin{bmatrix} I_{NS-L} & 0 \\ 0 & 0 \end{bmatrix}$ is a (NPA + 1) NS by (NPA + 1) NS matrix with I_{NS-L} the NS-L by NS-L identity matrix.

6.3 Problem Simplification

The computational effort involved in solving the optimal trajectory sensitivity problem by other methods[35,36] may be very large. The use of SOC reduces the computational requirements. However, the dimension of the augmented state vector may become unwieldy. For each parameter of the parameter vector the dimension of the augmented vector is increased by NS. For NS system states and NPA parameters, $\hat{x}$ has (NPA + 1)NS elements and the Ricatti matrix has $\frac{(NPA + 1)NS\,((NPA + 1)NS + 1)}{2}$ elements. For example, with a system of 7 states and 1 parameter the augmented state vector has 14 states and the symmetric Ricatti matrix has 105 independent elements. To obtain the solution to this Ricatti equation would require the solution of 105 simultaneous nonlinear equations, which is not a trivial task. If two additional parameters are considered, the corresponding Ricatti equation would involve 406 elements. If the SOCDES iterative approach is used to solve these equations, two linear systems of dimension equal to the number of unknown Ricatti elements must be solved at each iteration. Accuracy and running time considerations would indicate that this approach is not feasible for most practical problems.

However, a careful examination of the SOC sensitivity equations indicates that this "curse" of dimensionality may be reduced significantly. It is shown below that the computational effort involved in solving the sensitivity problem is approximately equal to the effort involved in solving a SOC problem for the original system, regardless of the number of parameters. That is, systems of equations on the order of $\frac{NS(NS + 1)}{2}$ must be solved for any number of parameters.

To demonstrate this reduction, the matrices of the Ricatti equation are partitioned into blocks of NS by NS elements. For convenience, a parameter vector of two elements is considered, NPA = 2.

Thus,

$$A_{\bar{K}} = \begin{bmatrix} A_K & 0 & 0 \\ A_{q_1} & A_K & 0 \\ A_{q_2} & 0 & A_K \end{bmatrix} : \quad 3NS \text{ by } 3NS$$

where $A_K = A - BK^T$

$$P = \begin{bmatrix} P_1 & P_2^T & P_3^T \\ P_2 & P_4 & P_5^T \\ P_3 & P_5 & P_6 \end{bmatrix} : \quad 3NS \text{ by } 3NS$$

$$S = \begin{bmatrix} S_1 & S_2^T & S_3^T \\ S_2 & S_4 & S_5^T \\ S_3 & S_5 & S_6 \end{bmatrix} : \quad 3NS \text{ by } 3NS$$

$$Q = \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & Q_2 & 0 \\ 0 & 0 & Q_3 \end{bmatrix} : \ 3NC \ \text{by} \ 3NC$$

$$\bar{K}^T = \begin{bmatrix} K^T & 0 & 0 \\ 0 & K^T & 0 \\ 0 & 0 & K^T \end{bmatrix} : \ 3NC \ \text{by} \ 3NS$$

and

$$\bar{K} Q \bar{K}^T = \begin{bmatrix} K Q_1 K^T & 0 & 0 \\ 0 & K Q_2 K^T & 0 \\ 0 & 0 & K Q_3 K^T \end{bmatrix} . \ 3NS \ \text{by} \ 3NS$$

Using this notation the SOC sensitivity Ricatti equation may be written as a set of 6 NS by NS matrix equations.

$$\underline{P_1 A_K + A_K^T P_1} + P_2^T A_{q_1} + A_{q_1}^T P_2 + P_3 A_{q_2} + A_{q_2}^T P_3 + S_1 + K Q_1 K^T = 0 \qquad (6.3.1)$$

$$\underline{\underline{P_2 A_K + A_K^T P_2}} + P_4 A_{q_1} + P_5^T A_{q_2} + S_2 = 0 \qquad (6.3.2)$$

$$\underline{\underline{P_3 A_K + A_K^T P_3}} + P_5 A_{q_1} + P_6 A_{q_2} + S_3 = 0 \qquad (6.3.3)$$

$$\underline{P_4 A_K + A_K^T P_4} + S_4 + K Q_2 K^T = 0 \qquad (6.3.4)$$

$$\underline{\underline{P_5 A_K + A_K^T P_5}} + S_5 = 0 \qquad (6.3.5)$$

$$\underline{P_6 A_K + A_K^T P_6} + S_6 + K Q_3 K^T = 0 \qquad (6.3.6)$$

and

$$K^T = \left[Q_1^{-1} B^T P_1 + Q_1^{-1} \frac{W^T}{2} \right] I_{11}^1 \tag{6.3.7}$$

Since P and S are symmetric, the diagonal blocks of the partitioned representation will also be symmetric while in general the off-diagonal elements will not be. Note the recurring underlined portions of the above equations and consider general matrix equations of the same form.

<u>Type I</u>:

$$\underline{P_i A_K + A_K^T P_i} = D_i \tag{6.3.8}$$

where P_i is symmetric.

<u>Type II</u>:

$$\underline{\underline{P_j A_K + A_K^T P_j}} = D_j \tag{6.3.9}$$

where P_j is not symmetric.

In Eq. (6.3.1)-(6.3.7), P_1, P_4, and P_6 are symmetric while P_2, P_3, and P_5 are not. If the SOCDES approach is used to solve the SOC problem, a stable K matrix is known at each iteration and Eq. (6.3.1)-(6.3.7) must be solved for P_i and P_j.

Since A_K is stable, there exists a unique solution to equations of Type I which may be found by the solution of an equivalent set of $\frac{NS}{2}(NS + 1)$ linear equations. Denote this equivalent set by

$$\text{"}A_K\text{" "}P_i\text{"} = \text{"}D_i\text{"} \tag{6.3.10}$$

This equivalent system of equations is described in detail in Appendix E. The manipulations involved in this transformation do not seem to be well known as evidenced by a recent publication.[46] The Type II equations may be reformulated so as to reduce the solution effort. Consider Eq. (6.3.9) and its transpose.

$$P_j A_K + A_K^T P_j = D_j \tag{6.3.11}$$

$$P_j^T A_K + A_K^T P_j^T = D_j^T \tag{6.3.12}$$

Define symmetric and skew symmetric matrices as follows:

$$\bar{P}_j = \frac{P_j + P_j^T}{2} ; \quad \bar{P}_j = \bar{P}_j^T$$

$$\bar{\bar{P}}_j = \frac{P_j - P_j^T}{2} ; \quad \bar{\bar{P}}_j^T = - \bar{\bar{P}}_j$$

and

$$P_j = \bar{P}_j + \bar{\bar{P}}_j$$

By adding and subtracting Eqs. (6.3.11) and (6.3.12) equations for $\bar{P}_j$ and $\bar{\bar{P}}_j$ are derived

$$\bar{P}_j A_K + A_K^T \bar{P}_j = (D_j + D_j^T)/2 \tag{6.3.13}$$

$$\bar{\bar{P}}_j A_K + A_K^T \bar{\bar{P}}_j = (D_j - D_j^T)/2 \tag{6.3.14}$$

Note that Eq.(6.3.13) is of Type I; thus the equivalent linear system of $\frac{NS}{2}(NS + 1)$ equations can be written as

$$"A_K" \; "\bar{P}_j" = "(D_j + D_j^T)"/2 \tag{6.3.15}$$

Since $\bar{\bar{P}}_j$ is skew symmetric, only $\frac{NS}{2}(NS - 1)$ elements must be found, corresponding to the lower or upper off diagonal triangular elements. Thus Eq. (6.3.14) is not of Type I but is closely related. An equivalent linear system can be found for these unknowns.

$$'A_K' \; '\bar{\bar{P}}_j' = '(D_j - D_j^T)'/2 \tag{6.3.16}$$

'A_K' is generated in much the same fashion as "A_K" except that minus signs are involved since $\bar{\bar{P}}_j$ is skew symmetric. Thus, Eq. (6.3.1)-(6.3.7) can be written in terms of the equivalent linear systems.

$$"A_K"\ "P_1" = -\ "(P_2^T A_{q_1} + A_{q_1}^T P_2 + P_3^T A_{q_2} + A_{q_2}^T P_3 + S_1 + K Q_1 K^T)" \qquad (6.3.17)$$

$$"A_K"\ "\bar{P}_2" = -\ "(P_4 A_{q_1} + A_{q_1}^T P_4 + P_5^T A_{q_2} + A_{q_2}^T P_5 + S_2 + S_2^T)"/2 \qquad (6.3.18)$$

$$"A_K"\ "\bar{P}_3" = -\ "(P_5 A_{q_1} + A_{q_1}^T P_5^T + P_6 A_{q_2} + A_{q_2}^T P_6 + S_3 + S_3^T)"/2 \qquad (6.3.19)$$

$$"A_K"\ "P_4" = -\ "(S_4 + K Q_2 K^T)" \qquad (6.3.20)$$

$$"A_K"\ "\bar{P}_5" = -\ "(S_5 + S_5^T)"/2 \qquad (6.3.21)$$

$$"A_K"\ "P_6" = -\ "(S_6 + K Q_3 K^T)" \qquad (6.3.22)$$

$$'A_K'\ '\bar{\bar{P}}_2' = -\ '(P_4 A_{q_1} - A_{q_1}^T P_4 + P_5^T A_{q_2} - A_{q_2}^T P_5 + S_2 - S_2^T)'/2 \qquad (6.3.23)$$

$$'A_K'\ '\bar{\bar{P}}_3' = -\ '(P_5 A_{q_1} - A_{q_1}^T P_5^T + P_6 A_{q_2} - A_{q_2}^T P_6 + S_3 - S_3^T)'/2 \qquad (6.3.24)$$

$$'A_K'\ '\bar{\bar{P}}_5' = -\ '(S_5 - S_5^T)'/2 \qquad (6.3.25)$$

Equations (6.3.17)-(6.3.22) are six systems of $\frac{NS(NS + 1)}{2}$ equations with the same coefficient matrix, while Eqs. (6.3.23)-(6.3.25) are three systems of $\frac{NS(NS - 1)}{2}$ equations with the same coefficient matrix. This is significant since after an initial solution to a system of linear equations is obtained, the computational effort involved in obtaining solutions for different right hand side vectors is relatively very low.

Thus, using this approach, Eq. (6.3.20), (6.3.21), (6.3.22) and (6.3.25) may be solved for P_4, P_5, and P_6. Then Eq. (6.3.18), (6.3.19), (6.3.23) and (6.3.24) are solved for P_2 and P_3. Finally Eq. (6.3.17) is used to find P_1.

To summarize, instead of solving a system of $\frac{(NPA + 1)NS\,((NPA + 1)NS + 1)}{2}$ equations to determine the Ricatti matrix, a system of $\frac{NS(NS + 1)}{2}$ equations is solved $\frac{(NP + 1)\,(NP + 2)}{2}$ times and a system of $\frac{NS(NS - 1)}{2}$ equations is solved $\frac{NP(NP + 1)}{2}$ times. For example, if NS = 7 with two parameters (NPA = 2), the solution of a system of 231 equations is replaced by the solution of a 28 equation system 6 times and a 21 equation system 3 times. This is a substantial reduction in computational effort.

With this computational approach, the SOC sensitivity problem is no more difficult to solve than a SOC problem for the original system. Thus SOC has a distinct computational advantage over other trajectory sensitivity formulations. It now becomes feasible to apply the sensitivity techniques to practical problems.

6.4 Examples

A. First Order Example

Consider the first order system described by this differential equation.

$$\dot{x} = a\,x + b\,u$$

Assume that the value of a is not accurately known but that it lies somewhere near a nominal value of -1 and let b = 1. The sensitivity variable for this problem is defined as follows.

$$z = \frac{\partial x}{\partial a}$$

Use the SOC sensitivity procedure to calculate a feedback control so that the closed loop system is insensitive with respect to a.

$$\dot{x} = (a-k)\ x \ ; \qquad u = -\ kx$$

Choose $\underline{\hat{u}}$ to minimize J,

$$J = \frac{1}{2} \int_{0}^{\infty} (\underline{\hat{x}}^T\ S\ \underline{\hat{x}} + \underline{\hat{x}}\ \hat{S}\ \underline{\hat{x}} + \underline{\hat{x}}^T\ W\ \underline{\hat{u}} + \underline{\hat{x}}^T\ \hat{W}\ \underline{\hat{u}} + \underline{\hat{u}}^T\ Q\ \underline{\hat{u}})\ dt$$

subject to

$$\dot{x} = -\ x + u$$

$$\dot{z} = -\ z + x + m$$

or

$$\dot{\underline{\hat{x}}} = \begin{bmatrix} -1 & 0 \\ 1 & -1 \end{bmatrix} \underline{\hat{x}} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \underline{\hat{u}}$$

and

$$S = \begin{bmatrix} \hat{s}_{11} & \hat{s}_{12} \\ \hat{s}_{21} & \hat{s}_{22} \end{bmatrix} \ ; \qquad s_{12} = s_{21}$$

$$W = \begin{bmatrix} \hat{w}_{11} & \hat{w}_{12} \\ \hat{w}_{21} & \hat{w}_{22} \end{bmatrix}$$

$$\underline{\hat{x}} = \begin{bmatrix} x \\ z \end{bmatrix}$$

$$\underline{\hat{u}} = \begin{bmatrix} u \\ m \end{bmatrix}$$

Let

$$P = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \ ; \ p_{12} = p_{21}$$

$$S = \begin{bmatrix} s_{11} & 0 \\ 0 & s_{22} \end{bmatrix}$$

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and

$$W = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The Ricatti equation which defines the solution gain is given below.

$$\bar{A}_{\bar{K}}^{T} P + P \bar{A}_{\bar{K}} + S + \bar{K} Q \bar{K}^{T} = 0$$

$$\bar{K} = \begin{bmatrix} k & 0 \\ 0 & k \end{bmatrix}$$

and

$$\left.\begin{aligned} &-2 p_{11} - 2 p_{11} k + 2 p_{12} + s_{11} + k^2 = 0 \\ &-2 p_{12} (1 + k) + p_{22} = 0 \\ &-2 p_{22} (1 + k) + s_{22} + k^2 = 0 \end{aligned}\right\} \qquad (6.4.1)$$

and

$$k = p_{11}$$

To illustrate the equivalent vector notation, P can be found as a function of k and thus k as a function of s_{11} and s_{22}.

$$E \text{ "}P\text{" } = - \text{ "}(S + \bar{K}Q\bar{K}^{T})\text{"}$$

The coefficient matrix E is obtained from Eq. (6.4.1).

$$"P" = \begin{bmatrix} p_{11} \\ p_{12} \\ p_{22} \end{bmatrix}$$

$$"S + \bar{K}Q\bar{K}^T" = \begin{bmatrix} s_{11} + k^2 \\ 0 \\ s_{22} + k^2 \end{bmatrix}$$

Thus,

$$\begin{bmatrix} -2-2k & 2 & 0 \\ 0 & -2-2k & 1 \\ 0 & 0 & -2-2k \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} -s_{11} - k^2 \\ 0 \\ -s_{22} - k^2 \end{bmatrix} \qquad (6.4.2)$$

Then p_{11} as a function of k, s_{11}, and s_{22} may be determined.

$$p_{11} = \frac{1}{2}\left[\frac{s_{11} + k^2}{1 + k} + \frac{s_{22} + k^2}{2(1+k)^3}\right] \qquad (6.4.3)$$

Since $p_{11} = k$, Eq. (6.4.3) can be used to define an equation in s_{11}, s_{22}, and k.

$$k^4 + 4\,k^3 + (\tfrac{9}{2} - s_{11})\,k^2 + (2 - 2\,s_{11})\,k - s_{11} - \frac{s_{22}}{2} = 0 \qquad (6.4.4)$$

A positive solution to this equation is sought since the positive definite solution to the Ricatti equation is of interest ($p_{11} > 0$). It is expected that as the weighting on the sensitivity variable, s_{22}, is increased the corresponding closed loop system will become less sensitive to changes in a.

Let the initial set of weightings be chosen as follows.

$s_{11} = 1.0$

$s_{22} = .876$

Equation (6.4.4) is solved to obtain

$$k = 0.5$$

The sensitivity weighting is increased.

$$s_{11} = 1.0$$

$$s_{22} = 15.0$$

and Eq. (6.4.4) is solved to obtain,

$$k = 1.0$$

As this weighting is increased further, the feedback gain also increases. Clearly, this leads to a decrease in the system sensitivity to a. For a off-nominal value of a, a = 0, this is verified by the entries in Table 6.1. The optimal trajectory is described by

$$\dot{x}^o = - (a^o + k)\, x^o = - (1 + k)\, x^o \; ; \quad x(0) = 1$$

while the off-nominal trajectory is described by

$$\dot{x} = - (k)\, x \; ; \quad x(0) = 1$$

This table also indicates integral square values of the sensitivity variable, z, and trajectory dispersion $\Delta x = x - x^o$. Note the integral square values of these variables decrease as the sensitivity weighting and feedback gain increase.

Although the actual value of the cost index may not be of any use, it is interesting to look at the specific nature of the formal index. To do this explicit values of $\hat{S}$ and $\hat{W}$ may be found from their respective definitions. From Eq. (6.4.2), with k = 1, $s_{11} = 1.0$, and $s_{22} = 15.0$, the Ricatti matrix is

$$P = \begin{bmatrix} 1 & 1 \\ 1 & 4 \end{bmatrix}$$

S_{22}	k	$\int_0^\infty x_e^2\,dt$	$\int_0^\infty z^2\,dt$	$\int_0^\infty \Delta x^2\,dt$
0.876	0.5	.333	.074	.3333
15.0	1.0	.250	.031	.0834
28.7 10^3	10.0	.045	.0003	.0003

Table 6.1
FIRST ORDER SENSITIVITY EXAMPLE

S	k	$\int_0^5 \underline{x}^T \underline{x}\,dt$	$\int_0^5 \underline{z}^T \underline{z}\,dt$	$\int_0^5 \Delta\underline{x}^T \Delta\underline{x}\,dt$
0.005	1.73	5.78 10^{-1}	3.51 10^{-1}	2.61 10^{-1}
0.1	1.75	5.71 10^{-1}	2.39 10^{-1}	2.57 10^{-1}
1.0	1.95	5.12 10^{-1}	2.46 10^{-1}	1.65 10^{-1}
10.0	2.68	3.49 10^{-1}	9.46 10^{-2}	4.66 10^{-2}
100.0	4.34	2.17 10^{-1}	2.25 10^{-2}	8.22 10^{-3}
$1\cdot10^4$	8.40	1.00 10^{-1}	3.07 10^{-3}	9.30 10^{-4}
$1\cdot10^6$	26.6	2.65 10^{-2}	1.44 10^{-4}	3.63 10^{-5}

Table 6.2
SECOND ORDER SENSITIVITY EXAMPLE

The structure of $\hat{W}$ is somewhat simplified since all of the states, in this case one, are fed back. From the definition of $\hat{W}$, given in Appendix D, and noting that $W = 0$ and $I_{11}^{2} = 0$,

$$\hat{W}_{11} = -2\,[[\,I_{11}^{2}(P\bar{B} + \tfrac{W}{2})]]_{11} = 0$$

$$\hat{W}_{12} = -2\,[[P\bar{B}]]_{12} = -2\left[\left[\begin{pmatrix}1 & 1\\ 1 & 4\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\right]\right] = \left[\begin{bmatrix}-2 & -2\\ -2 & -8\end{bmatrix}\right] = -2$$

$$\hat{W}_{12} = -2$$

See Appendix D for an explanation of this notation. Similarly

$$\hat{W}_{21} = -2\,[[P\bar{B}]]_{21} = -2$$

and

$$\hat{W}_{22} = 2\left\{[[I_{11}^{1}\,P\bar{B}]]_{11} - [[P\bar{B}]]_{22}\right\}$$

$$P\bar{B} = \begin{bmatrix}1 & 1\\ 1 & 4\end{bmatrix}; \quad I_{11}^{1}\,P\bar{B} = \begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}1 & 1\\ 1 & 4\end{bmatrix} = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$$

Thus

$$\hat{W}_{22} = 2(1 - 4) = -6$$

and

$$\hat{W} = \begin{bmatrix}0 & -2\\ -2 & -6\end{bmatrix}$$

As a check

$$\bar{K}^T = (B^T P + \frac{\hat{W}}{2}) = \begin{bmatrix} 1 & 1 \\ 1 & 4 \end{bmatrix} + \begin{bmatrix} 0 & -1 \\ -1 & -3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Finally $\hat{S}$ is given by its definition.

$$\hat{S} = \frac{1}{2} (\hat{W}\bar{K}^T + \bar{K}\hat{W}^T) = \begin{bmatrix} 0 & -2 \\ -2 & -6 \end{bmatrix}$$

Using these values of $\hat{S}$ and $\hat{W}$ this SOC problem may be stated as choosing u and m to minimize J.

$$J = \frac{1}{2} \int_0^{\infty} (x^2 + 9 z^2 - 4 xz - 2 xm - 2 zu - 6 mz + u^2 + m^2) \, dt$$

subject to

$$\dot{x} = - x + u$$

$$\dot{z} = - z + x + m$$

with the solution

$$u = - x$$

$$m = - z \, .$$

B. Second Order Example

To compare this method with other techniques, consider the second order damped oscillator example. Once again the differential equation describing this system is given by

$$\ddot{y} + 2 \zeta \omega \dot{y} + \omega^2 y = u$$

Assume that the damping parameter ζ is susceptible to variations. The state equations are

$$\underline{x} = \begin{bmatrix} \dot{y} \\ y \end{bmatrix}$$

$$\dot{\underline{x}} = A\,\underline{x} + \underline{b}\,u$$

$$A = \begin{bmatrix} -2\,\zeta\,\omega & -\omega^2 \\ 1 & 0 \end{bmatrix}$$

$$b = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and

$$A_q = \begin{bmatrix} -2\omega & 0 \\ 0 & 0 \end{bmatrix}$$

Only the rate signal will be fed back. Thus

$$u = -\,k\,x_1 = -\,k\,\dot{y}$$

and NS = 2, L = 1, NC = 1. For illustrative purposes, use the reduced formulation described by the following equations where each of the partition blocks is of the proper dimension to allow consistent multiplication.

$$P = \begin{bmatrix} P_1 & P_2^T \\ P_2 & P_3 \end{bmatrix}$$

$$S = \begin{bmatrix} S_1 & S_2^T \\ S_2 & S_3 \end{bmatrix}$$

$$W = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$A_K = A - BK^T = \begin{bmatrix} -k-2\zeta\omega & -\omega^2 \\ 1 & 0 \end{bmatrix}$$

$$K = \begin{bmatrix} k \\ 0 \end{bmatrix}$$

$$Q = \begin{bmatrix} q & 0 \\ 0 & q \end{bmatrix}$$

The reduced matrix equations describing the optimal solutions are

$$P_1 A_K + A_K^T P_1 + P_2^T A_q + A_q^T P_2 + S_1 + K Q K^T = 0 \tag{6.4.5}$$

$$P_2 A_K + A_K^T P_2 + P_3 A_q + S_2 = 0 \tag{6.4.6}$$

$$P_3 A_K + A_K^T P_3 + S_3 + K Q K^T = 0 \tag{6.4.7}$$

Let

$$P_1 = \begin{bmatrix} p_1^1 & p_1^2 \\ p_1^2 & p_1^3 \end{bmatrix} : \quad S_1 = \begin{bmatrix} s_1^1 & s_1^2 \\ s_1^2 & s_1^3 \end{bmatrix}$$

$$P_2 = \begin{bmatrix} p_2^1 & p_2^3 \\ p_2^2 & p_2^4 \end{bmatrix} : \quad S_2 = \begin{bmatrix} s_2^1 & s_2^3 \\ s_2^2 & s_2^4 \end{bmatrix}$$

$$P_3 = \begin{bmatrix} p_3^1 & p_3^2 \\ p_3^2 & p_3^3 \end{bmatrix} : \quad S_3 = \begin{bmatrix} s_3^1 & s_3^2 \\ s_3^2 & s_3^3 \end{bmatrix}$$

where the elements of these matrices are scalars. Assume that $\mathcal{J}^{o} = 0$ and $\omega = 1$; then Eq. (6.4.7) can be written,

$$\left.\begin{aligned} & -2\,p_3^{\,1}\,k + 2\,p_3^{\,2} + s_3^{\,1} + q\,k^2 = 0 \\ & -p_3^{\,1} - p_3^{\,2}k + p_3^{\,3} + s_3^{\,2} = 0 \\ & -2\,p_3^{\,2} + s_3^{\,3} = 0 \end{aligned}\right\} \qquad (6.4.8)$$

Equation (6.4.6) becomes,

$$\left.\begin{aligned} & -k\,p_2^{\,1} + p_2^{\,3} - k\,p_2^{\,1} + p_2^{\,2} - 2\,p_3^{\,1} + s_2^{\,1} = 0 \\ & -p_2^{\,1} - k\,p_2^{\,3} + p_2^{\,4} + s_2^{\,3} = 0 \\ & -k\,p_2^{\,2} + p_2^{\,4} - p_2^{\,1} - 2\,p_3^{\,2} + s_2^{\,2} = 0 \\ & -p_2^{\,2} - p_2^{\,3} + s_2^{\,4} = 0 \end{aligned}\right\} \qquad (6.4.9)$$

and Eq. (6.3.1) is equivalent to

$$\left.\begin{aligned} & -2\,p_1^{\,1}\,k + 2\,p_1^{\,2} - 4\,p_2^{\,1} + s_1^{\,1} + q\,k^2 = 0 \\ & -p_1^{\,1} - p_1^{\,2}\,k + p_1^{\,3} - 2\,p_2^{\,2} + s_1^{\,2} = 0 \\ & -2\,p_1^{\,2} + s_1^{\,3} = 0 \end{aligned}\right\} \qquad (6.4.10)$$

The gain equation is

$$K^T = \frac{b^T\,P_1\,I_{11}^{\,1}}{q}$$

or

$$k = \frac{p_1^{\,1}}{q} \qquad (6.4.11)$$

For the following set of weightings

$$S_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; \qquad S_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$S_3 = \begin{bmatrix} .005 & 0 \\ 0 & .005 \end{bmatrix}; \qquad q = 1$$

Equation (6.4.8)-(6.4.11) can be solved to obtain

$$k = 1.73$$

As the sensitivity weighting

$$S_3 = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$$

is increased, the resulting closed loop system becomes less sensitive to $\mathcal{J}$ as shown in Table 6.2. The results entered in this table were obtained for a off-nominal $\mathcal{J}$ of -1.0. Again for this simple problem the insensitive nature is obtained by an increase in the magnitude of the feedback gain so that for the same value of parameter variation, the relative effect is diminished. Figure 6.1 indicates the nominal and off-nominal time domain response to an initial condition of $\dot{y}(0) = 1$ for $k = 1.73$ and $k = 8.4$ while Fig. 6.2 and 6.3 compare the sensitivity variables and trajectory dispersions.

A basic difference between the model reference and sensitivity techniques is pointed out in the responses of Fig. 6.1. In the sensitivity approach the feedback gains and hence the nominal trajectory are chosen to be insensitive to parameter variations. In the model reference technique the nominal performance of the system is independent of any sensitivity considerations. This may be an advantage since reduced sensitivity may correspond to degraded nominal performance. The "price" paid for this model reference feature is the increased complexity of the model reference controller.

As an indication of the feasibility of the SOC sensitivity approach, it was compared with the method described by Dougherty.[36] Both methods were used to solve the same second order problem which is similar to the problem discussed above except that both position and rate information is fed-back. Dougherty's

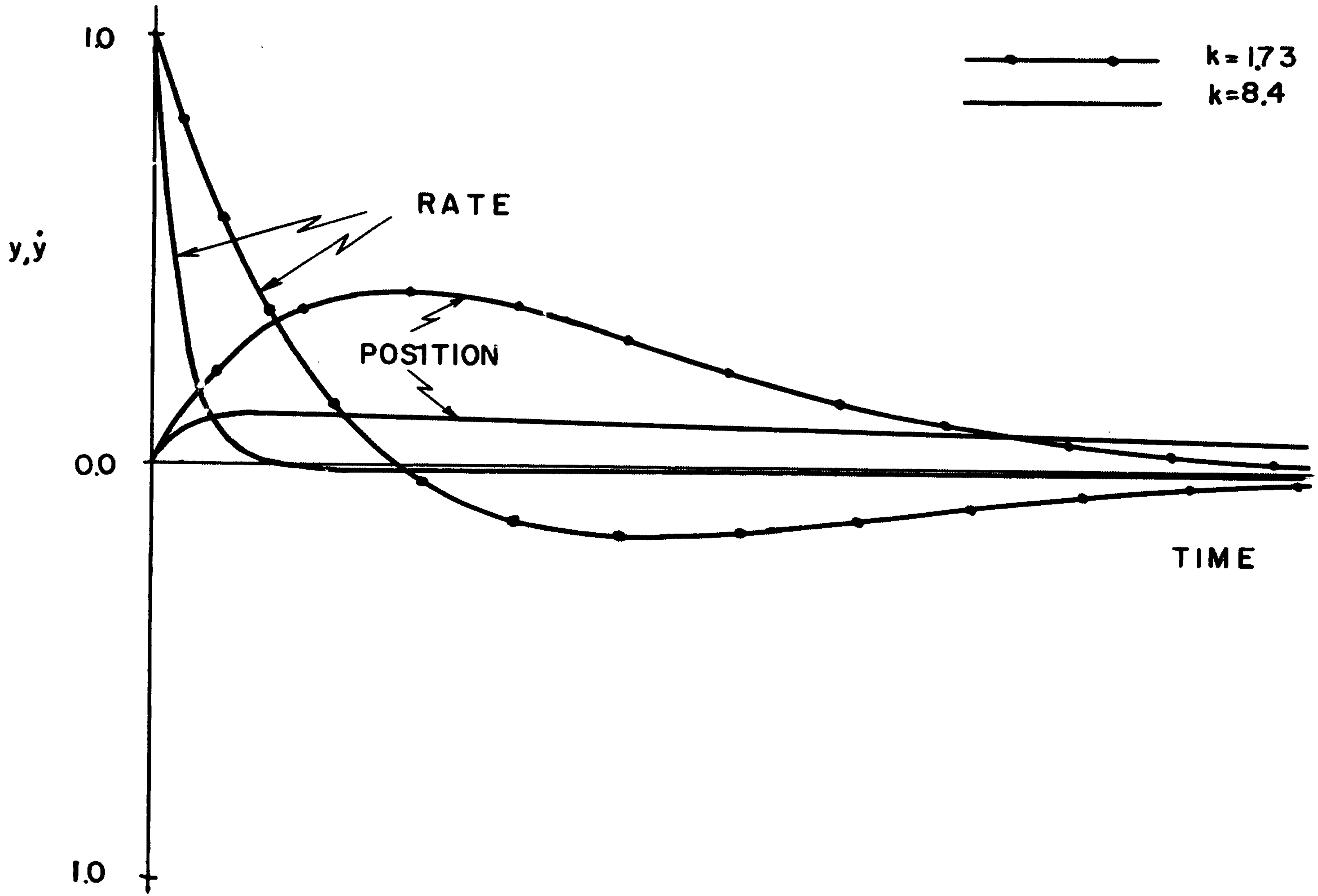


FIG. 6.1 INITIAL CONDITION RESPONSE

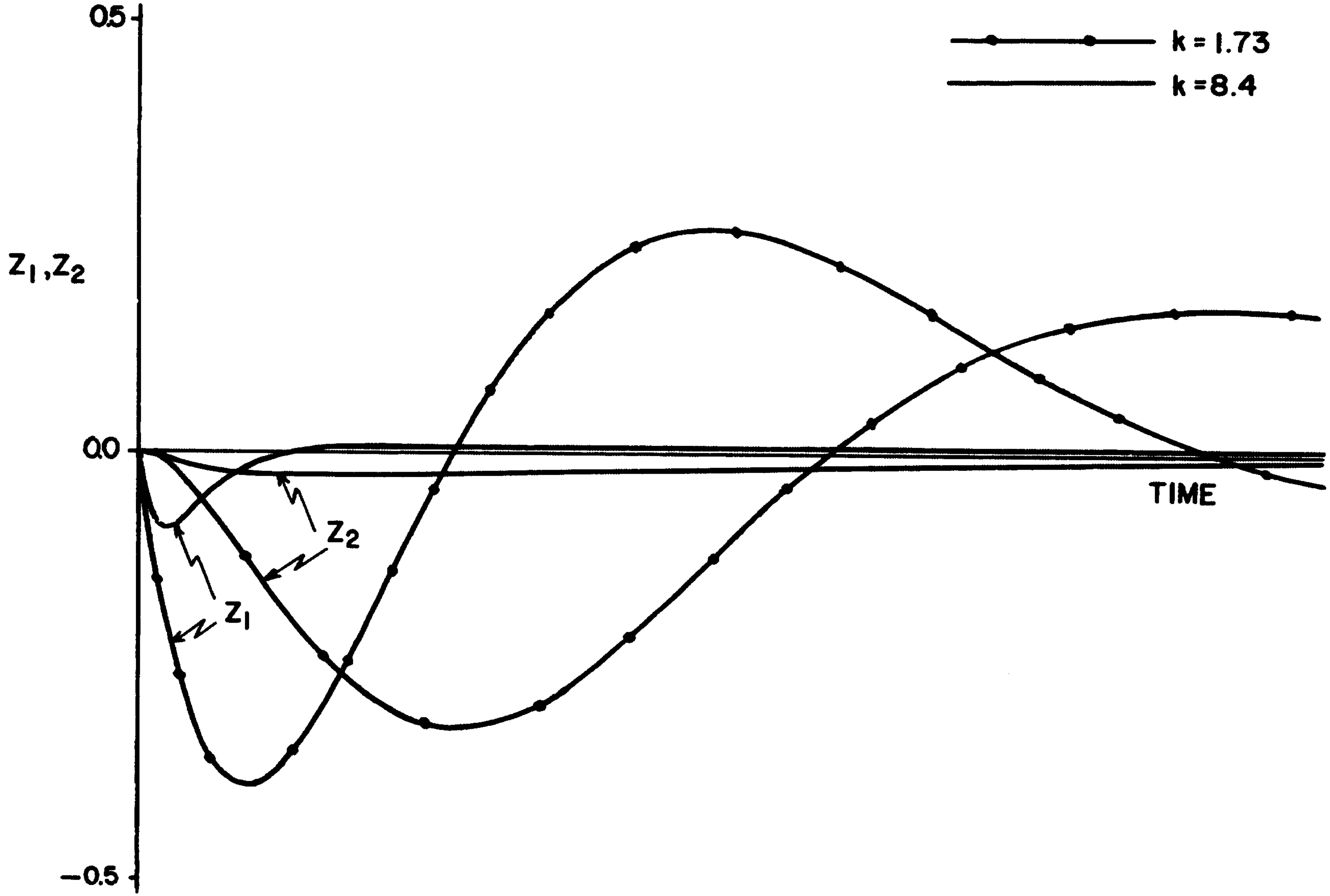


FIG. 6.2 SENSITIVITY VARIABLES

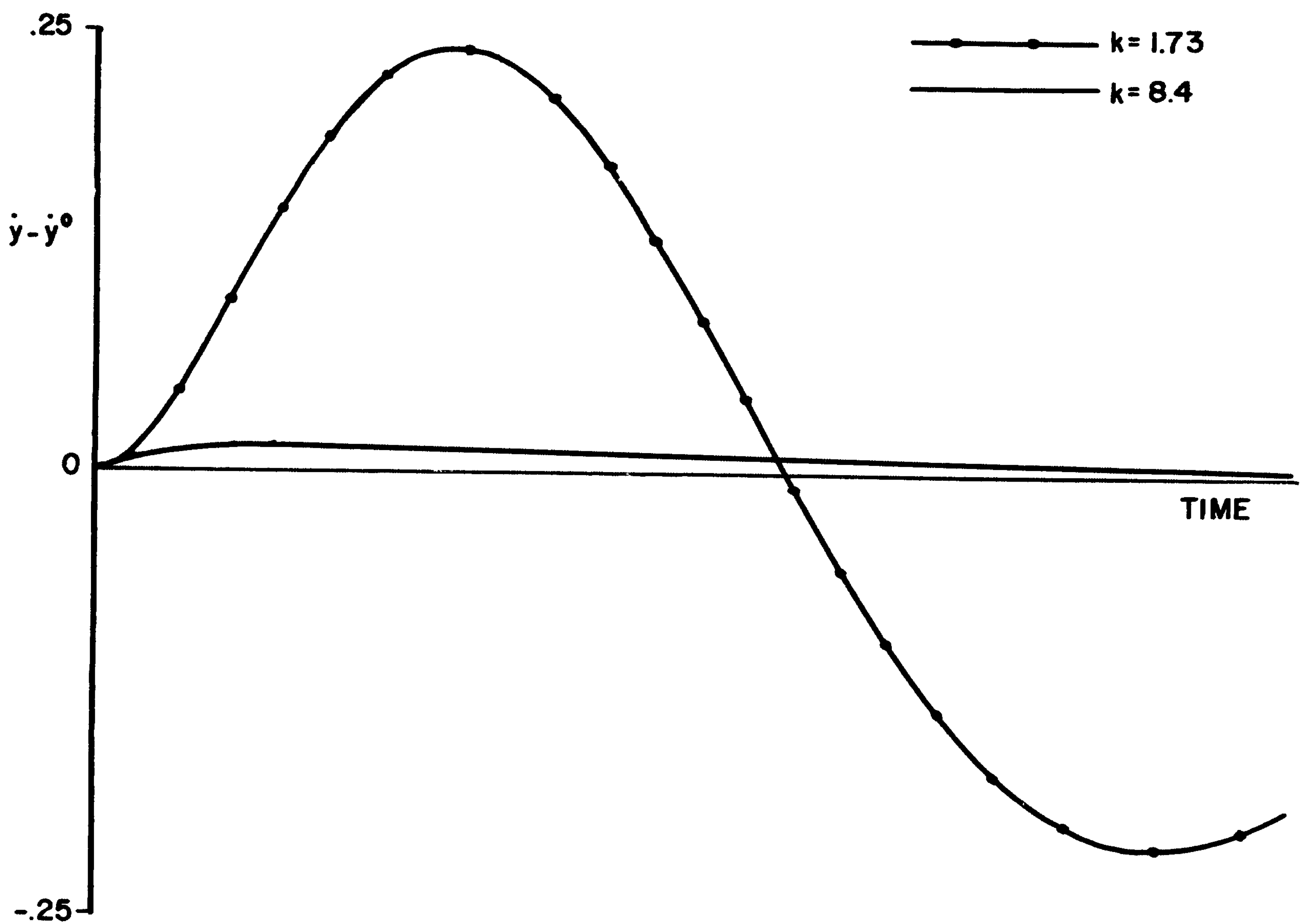


FIG. 6.3-a RATE TRAJECTORY DISPERSION

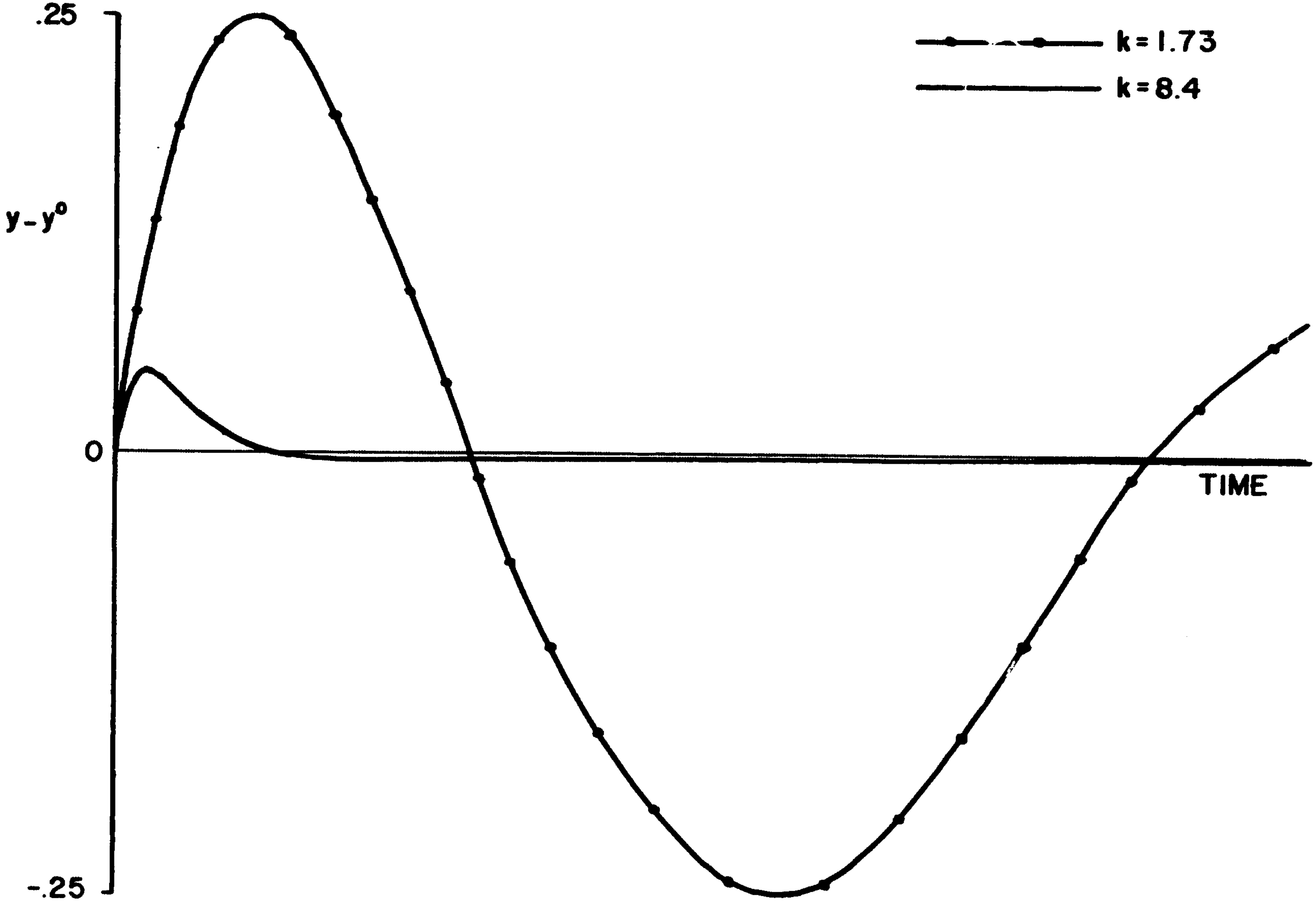


FIG. 6.3-b   POSITION TRAJECTORY DISPERSION

initial control law was

$$u = -2.44\, y - 2.93\, \dot{y}$$

These gains can be obtained with the SOC sensitivity approach with the following weightings.

$$q = 1.0 \qquad S_1 = \begin{bmatrix} 10.9 & 0 \\ 0 & 1.0 \end{bmatrix}$$

$$S_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad S_3 = \begin{bmatrix} 27.4 & 0 \\ 0 & 27.4 \end{bmatrix}$$

Using Dougherty's technique a desensitized control law

$$u = -2.78\, y - 4.16\, y$$

is obtained with an execution time of about <u>fifteen</u> <u>minutes</u> on an IBM model 360/50 digital computer. This same control law can be obtained with SOC with the following weightings; note the increase in sensitivity weighting.

$$q = 1 \;; \qquad S_1 = \begin{bmatrix} 13.8 & 0 \\ 0 & 1 \end{bmatrix}$$

$$S_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}; \qquad S_3 = \begin{bmatrix} 281. & 0 \\ 0 & 281. \end{bmatrix}$$

The execution time required to solve this problem using the SOCDES algorithm was <u>ten</u> <u>seconds</u>! As the size of the problem considered increases the execution time requirements of the SOC technique increase but they still remain reasonable as shown in Chapter VII where the technique is applied to the Saturn V launch vehicle problem.

Nomenclature

Matrices

A	System matrix: NS by NS
$\hat{A}$	Augmented system matrix: (NPA + 1)NS by (NPA + 1)NS
$\bar{A}$	Augmented system matrix: (NPA + 1)NS by (NPA + 1)NS
$A_{\bar{K}}$	Closed loop augmented system matrix: (NPA + 1)NS by (NPA + 1)NS
A_K	Closed loop system matrix: NS by NS
A_{q_J}	Partial derivative matrix: NS by NS
B	Control coefficient matrix: NS by NC
$\bar{B}$	Augmented system control coefficient matrix: (NBA + 1)NS by (NPA + 1)NC
$\hat{B}$	Augmented system control coefficiet matrix: (NPA + 1)NS by NC
D_i	Notational matrix
K	Feedback gain matrix: NS by NC
$\bar{K}$	Augmented system feedback gain matrix: (NPA + 1)NS by (NPA + 1)NC
P	Ricatti matrix: (NPA + 1)NS by (NPA + 1)NS
P_i	Component of Ricatti matrix: NS by NS
$\bar{P}_J$	Symmetric part of Ricatti component matrix: NS by NS
$\bar{\bar{P}}_J$	Skew-Symmetric part of Ricatti component matrix: NS by NS
Q	Symmetric control weighting matrix: (NPA + 1)NC by (NPA + 1)NC
$\hat{S}$	Symmetric state weighting matrix: (NPA + 1)NS by (NPA + 1)NS
S	Symmetric state weighting matrix; class two: (NPA + 1)NS by (NPA + 1)NS
S_i	Component of state weighting matrix: NS by NS
W	Bilinear weighting matrix: (NPA + 1)NS by (NPA + 1)NC
$\hat{W}$	Bilinear weighting matrix; class two: (NPA + 1)NS by (NPA + 1)NC
Z	Sensitivity matrix: NS by NPA

Vectors

$\underline{m}_i$	Control vector of i^{th} sensitivity vector: NC
$\underline{q}$	Parameter vector: NPA
$d\underline{q}$	Perturbation parameter vector: NPA
$\underline{u}$	System control vector: NC
$\hat{\underline{u}}$	Augmented system control vector: (NPA + 1)NC
$\underline{x}$	State vector: NS
$\hat{\underline{x}}$	Augmented system vector: (NPA + 1)NS
$\underline{z}_i$	Sensitivity vector of i^{th} parameter: NS

Chapter VII

CASE STUDY: THE LAUNCH VEHICLE PROBLEM

7.1 Introduction

In this chapter, the techniques developed in the preceding sections are demonstrated by their application to the significant engineering problem of the altitude control of a large launch vehicle of the Saturn class. The vehicle configuration is shown in Fig. 7.1. The first stage propulsion is obtained from five liquid fuel engines each of which generates about 1.5 million pounds of thrust. Control is obtained by gimballing or swivelling four of the five engines. This vehicle is a large complex system which is difficult to control. Neither classical nor currently available modern techniques have been particularly effective in solving this problem.

There are two major sources of difficulty. The first stems from the physical characteristics of the vehicle and is independent of any design technique. The basic objective of this control problem is to force the vehicle to remain in the neighborhood of the programmed nominal trajectory despite environmental disturbances. Each new generation of launch vehicles is larger than the last; the length to width ratio decreases corresponding to an increase in the flexible nature of the vehicle. For the Saturn V vehicle this length to width rate is about 10 to 1 and the flexible modes pose a serious problem. Under certain flight conditions it is possible to excite these modes to such an extent that the vehicle destroys itself. Thus an important objective of the control system is stability of the bending motions as well as control of the rigid motions of the vehicle.

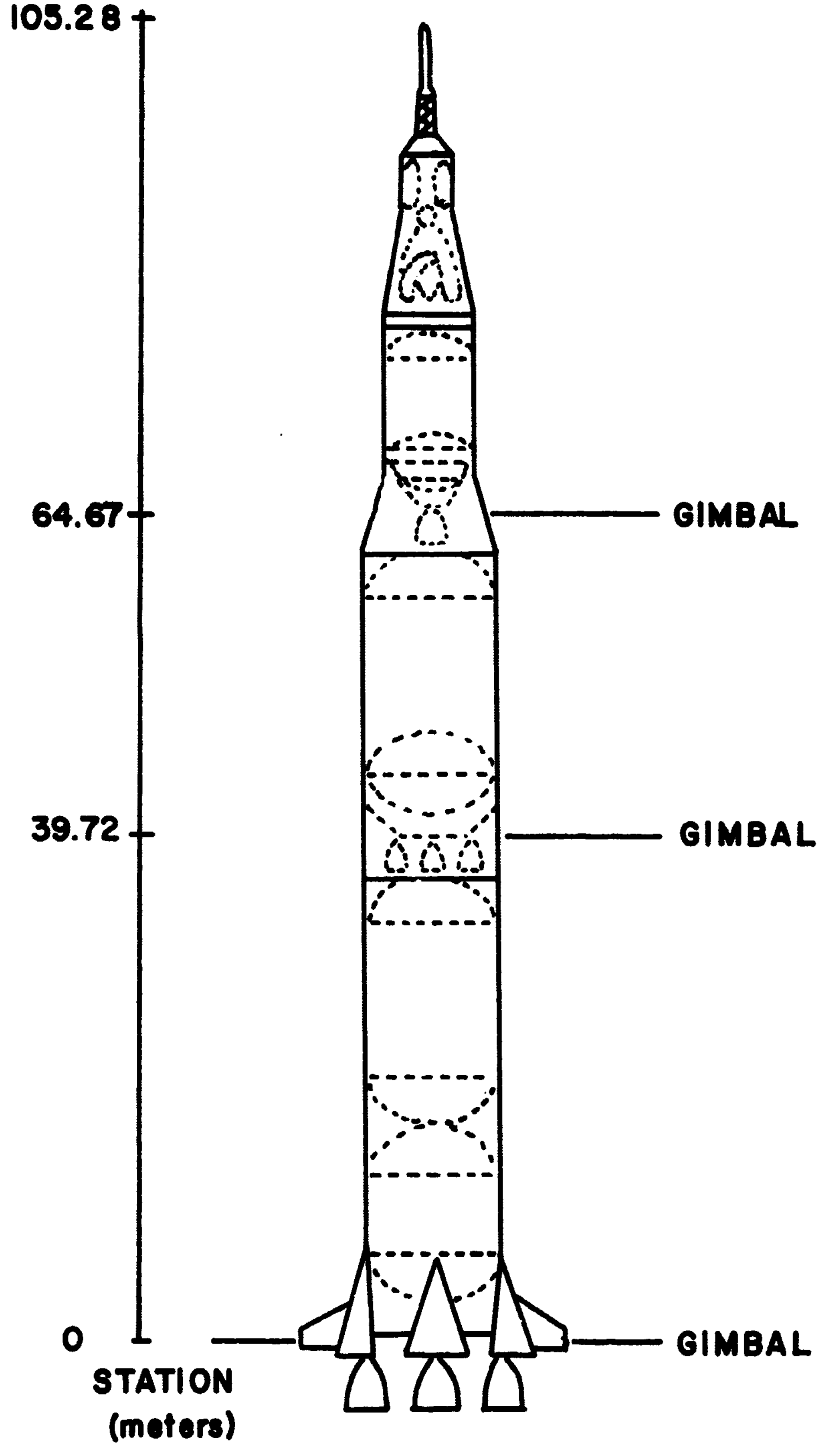


FIG. 7-1 VEHICAL CONFIGURATION

The study of the launch vehicle involves a significant modeling problem. Even after a reasonably satisfactory model structure has been determined, the physical size of the vehicle inhibits the accurate evaluation of the model parameters. Some of these parameters, such as bending frequencies, may be critical with respect to the accuracy of the model in that off nominal values of these parameters may render ineffective the control designed on the basis of the nominal values. Present techniques for estimating these parameters include physically shaking the vehicle and noting its behavior. For vehicles larger than the Saturn V, this does not appear to be a feasible approach and analytic techniques will have to be used. Moreover, the bending frequencies are functions of the physical configuration of the vehicle and hence the payload which changes from mission to mission. It would be advantageous to be able to use the same launch vehicle control system for a variety of missions. Thus it is important to be able to design a control system which is insensitive to inaccurate knowledge of the bending frequencies. More specifically the system will be designed to give adequate control for variation in the bending frequencies of $\pm$ 20%.

The fuel for the liquid-fuel engines of the Saturn V booster is stored in tanks. The dynamics of the vehicle are influenced by the movement or sloshing of the fuel in the partially filled tanks. For the present study it is assumed that the slosh modes are adequately damped by tank baffles.

In Fig. 7.2 the frequency spectrum of the launch vehicle is shown. The spectra of the engine and gimbal dynamics are indicated as well as those of the bending and slosh. Some of the spectra are represented by bands indicating that the frequencies change with time.

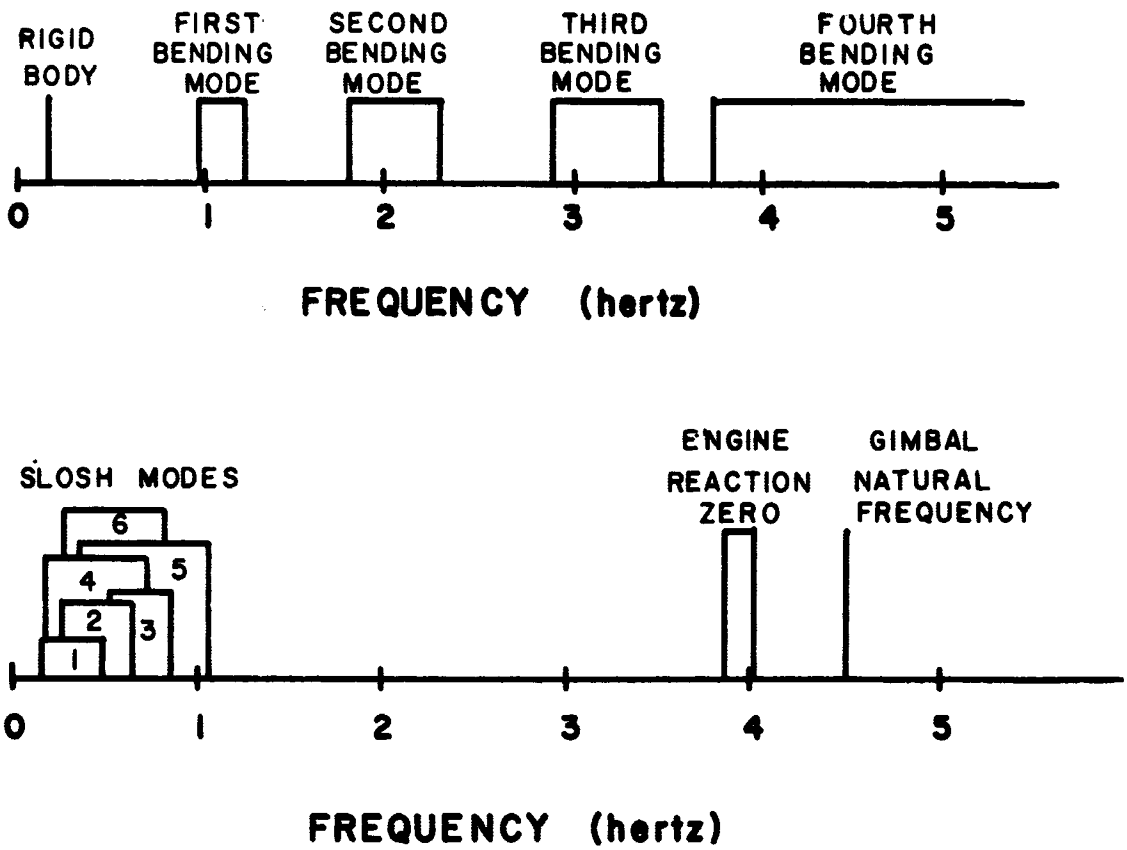


FIG. 7-2 BOOSTER FREQUENCY SPECTRUM

The control problem is further complicated since the booster is aerodynamically unstable for most of the launch trajectory. This is caused by the center of pressure being forward of the center of gravity. The center of pressure is a point at which the normal aerodynamic force is assumed to act while the booster rotates around the center of gravity. Thus the force of the wind tends to topple the vehicle.

The flexible nature of the vehicle introduces a measurement problem. At present, position and rate gyros are the available sensors. Unfortunately, these devices measure local movements and thus their output is a combination of rigid and bending motions. Previous design approaches have used filters to separate the rigid and bending signals, however this approach is hampered by the lack of knowledge about the bending frequencies.

The second major source of difficulty becomes obvious when an attempt is made to choose a satisfactory design technique. Many of the classical design techniques are not suitable due to the complexity of the system and the parameter variation problem. The current modern techniques are not satisfactory from a computational point of view as well as the lack of an unavailable state capability. Even if the rigid and bending modes are separated, the usual optimal control approach would require the use of sensors to measure all of the states including the angle-of-attack, engine dynamics, and any compensator states. This is clearly an unreasonable requirement since adequate control has been obtained using only pitch and pitch rate feedback.

The SOC approach is shown to be very useful in the design of control systems for the launch vehicle since many of the difficulties discussed above are eliminated. In the following sections the equations of motion of the vehicle are derived, a state variable model is chosen, a control structure is proposed, and the various SOC techniques are applied.

7.2 Launch Vehicle Model[48,49]

In order to design a control system for the launch vehicle it is necessary to derive a mathematical model of its dynamical behavior. This model should be complicated enough to allow an accurate description of the physical situation and yet not so complicated as to prevent analysis.

The launch vehicle has six degrees of freedom, three translational, and three rotational. In this study only the motion of the vehicle in the pitch plane is considered and a flat earth with constant gravity is assumed. The inertial co-ordinate system (X, Y), is located at the launch point and defines the local verticle. A second co-ordinate system (x, y) is aligned with the longitudinal axis of the vehicle and centered at the center of gravity. A third co-ordinate system (X_n, Y_n) defines the nominal trajectory of the vehicle; if the vehicle follows a nominal trajectory the (x, y) and (X_n, Y_n) co-ordinates will coincide, $\chi_c = 0$. See Fig. 7.3. It should be emphasized that the equations of motions are written in the inertial space defined by (X, Y) but the nature of the investigation requires that the equations be expressed in terms of the other co-ordinate systems.

The result of the following derivation will be a set of linear differential equations which will characterize the motion of the vehicle about its nominal trajectory. These equations are obtained by applying the laws of Newtonian mechanics. The basic assumption is made that the rigid and bending motions may be modeled separately and then added to give an accurate representation of the behavior of the vehicle.

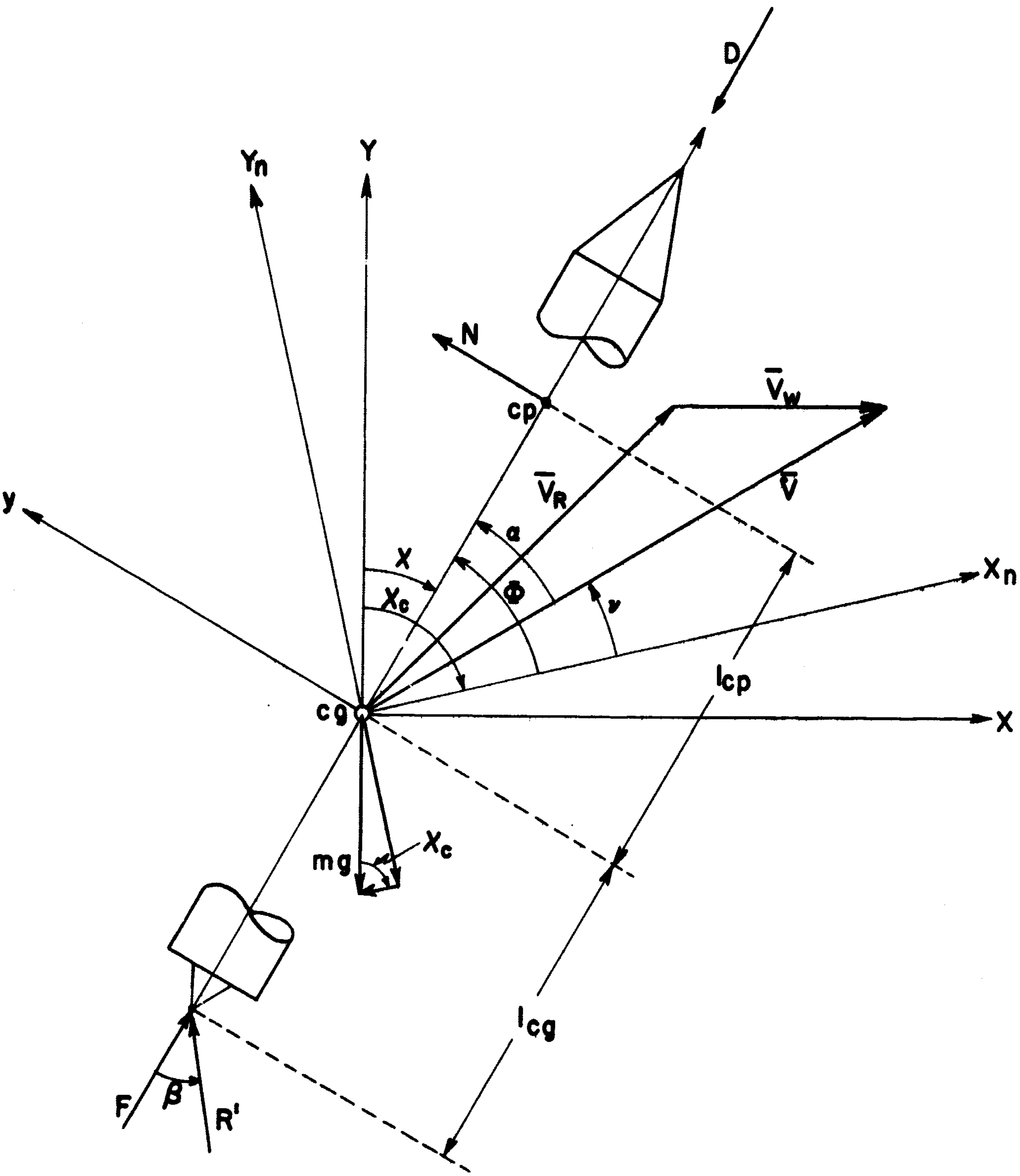


FIG. 7.3 FREE BODY DIAGRAM

Since the control is obtained by gimballing some of the engines, a portion, F, of the thrust acts along the longitudinal axis of the vehicle, while the gimballed thrust, R', acts at an angle of β degrees with respect to the centerline. The aerodynamic force is decoupled into two components; the drag force, D, acts along the centerline of the vehicle while the normal force, N, acts in a orthogonal direction to the centerline at the center of pressure. The sum of the forces in the X_n direction is

$$F_{X_n} = (F + R' \cos \beta - D) \cos \phi - mg \cos \chi_c - (N + R' \sin \beta) \sin \phi \quad (7.2.1)$$

while the sum of the forces in the Y_n direction is

$$F_{Y_n} = (F + R' \cos \beta - D) \sin \phi + (N + R' \sin \beta) \cos \phi - mg \sin \chi_c \quad (7.2.2)$$

while the sum of the moments about the center of gravity is given by

$$I \ddot{\phi} = - R' l_{cg} \sin \beta - N l_{cp} \quad (7.2.3)$$

The velocity of the vehicle, $\underline{v}$, is measured in the inertial frame but expressed in the nominal frame

$$\underline{v} = v \cos \gamma \; \underline{i} + v \sin \gamma \; \underline{j} \quad (7.2.4)$$

where $v = \| \underline{v} \|$ and $\underline{i}$ and $\underline{j}$ are unit vectors in the X_n and Y_n directions respectively.

Since (X_n, Y_n) is not an inertial frame of reference, the unit vectors are timevarying and thus the acceleration of the vehicle expressed in this frame is given by

$$\underline{a} = \frac{dv}{dt} \cos \gamma \; \underline{i} - v \sin \gamma \; \frac{d\gamma}{dt} \underline{i} + v \cos \gamma \; \frac{d\underline{i}}{dt}$$

$$+ \frac{dv}{dt} \sin \gamma \; \underline{j} - v \cos \gamma \; \frac{d\gamma}{dt} \underline{j} + v \sin \gamma \; \frac{d\underline{j}}{dt} \quad (7.2.5)$$

The angular velocity of the nominal co-ordinate system with respect to inertial space is given by $-\dot{\chi}_c$ in the $\underline{k}$ direction out of the pitch plane. Thus

$$\frac{d\underline{i}}{dt} = -\dot{\chi}_c\, \underline{k} \times \underline{i} = -\dot{\chi}_c\, \underline{j}$$

$$\frac{d\underline{j}}{dt} = -\dot{\chi}_c\, \underline{k} \times \underline{j} = \dot{\chi}_c\, \underline{i}$$

The acceleration may be expressed as

$$\underline{a} = (\dot{v} \cos \vartheta - v \cos \vartheta\, \dot{\vartheta} + v \sin \vartheta\, \dot{\chi}_c)\, \underline{i}$$
$$+ (\dot{v} \sin \vartheta - v \cos \vartheta\, \dot{\vartheta} - v \cos \vartheta\, \dot{\chi}_c)\, \underline{j} \qquad (7.2.6)$$

The acceleration can be decomposed into components lying in the X_n and Y_n directions.

$$\underline{a} \cdot \underline{i} = (\ddot{X}_n + v \sin \vartheta\, \dot{\chi}_c)$$

$$\underline{a} \cdot \underline{j} = (\ddot{Y}_n - v \cos \vartheta\, \dot{\chi}_c)$$

where

$$\ddot{X}_n = \frac{d}{dt} \dot{X}_n = \frac{d}{dt} (v \cos \vartheta)$$

$$\ddot{Y}_n = \frac{d}{dt} (\dot{Y}_n) = \frac{d}{dt} (v \sin \vartheta)$$

Although the equations of motion are written in the inertial space, they may be expressed in the nominal co-ordinate system.

$$m(\ddot{X}_n + v \sin \vartheta\, \dot{\chi}_c) = (F + R' \cos \beta - D) \cos \phi - N \sin \phi - R' \sin \beta \sin \phi$$
$$- mg \cos \chi_c \qquad (7.2.7)$$

$$m(\ddot{Y}_n - v \cos \vartheta\, \dot{\chi}_c) = (F + R' \cos \beta - D) \sin \phi + N \cos \phi + R' \sin \beta \cos \phi$$
$$- mg \sin \chi_c \qquad (7.2.8)$$

The normal aerodynamic force is proportional to the angle of attack.

$$N = N' \alpha$$

Using this relation Eq. (7.2.7) and (7.2.8) can be solved for $\ddot{X}_n$ and $\ddot{Y}_n$ respectively.

$$\ddot{X}_n = \frac{(F + R' \cos \beta - D)}{m} \cos \phi - \frac{N'\alpha}{m} \sin \phi - v \sin \nu \, \dot{\chi}_c - \frac{R'}{m} \sin \beta \sin \phi - g \cos \chi_c$$

$$\ddot{Y}_n = \frac{(F + R' \cos \beta - D)}{m} \sin \phi + \frac{N'\alpha}{m} \cos \phi + \frac{R'}{m} \sin \beta \cos \phi - g \sin \chi_c + v \cos \nu$$

These equations are linearized by using the following small angle approximations.

$$\sin \phi = \phi \qquad \sin \beta = \beta \qquad \sin \nu = \nu \qquad \sin \beta \sin \phi = 0$$

$$\cos \phi = 1 \qquad \cos \beta = 1 \qquad \cos \nu = 1 \qquad \alpha \sin \phi = 0$$

$$\ddot{X}_n = \frac{F + R' - D}{m} - v \, \nu \, \dot{\chi}_c - g \cos \chi_c \tag{7.2.9}$$

$$\ddot{Y}_n = \frac{(F + R' - D)}{m} \phi + \frac{N'\alpha}{m} + \frac{R'}{m} \beta + v \, \dot{\chi}_c \, g \sin \chi_c \tag{7.2.10}$$

$$\ddot{\phi} = - \frac{(R' l_{cg})}{I} \beta - N' \, l_{cp} \, \alpha \tag{7.2.11}$$

Since disturbances do not seriously effect the motions of the vehicle in the X_n direction, the equations are simplified by assuming that the origin of the nominal co-ordinate system moves with the vehicle in that direction.[49] Also, the nominal trajectory involves a gravity turn, that is

$$\dot{\chi}_c = \frac{g \sin \chi_c}{v}$$

Then Eq. (7.2.10) and (7.2.11) become

$$\ddot{Y}_n = \frac{(F + R' - D)}{m} \phi + \frac{N'}{m} \alpha + \frac{R'}{m} \beta \tag{7.2.12}$$

$$\ddot{\phi} = - \frac{(R' \, l_{cg})}{I} \beta - \frac{(N' \, l_{cp})}{I} \alpha \qquad (7.7.13)$$

The main source of additive disturbances is provided by the wind which is assumed to blow in a horizontal plane only. The wind induces an additional contribution, α_w, to the angle-of-attack. Figure 7.3 indicates an angular relationship which relates the angle-of-attack to the variables of the above equations.

$$\alpha - \alpha_w = \phi - \frac{\dot{Y}_n}{v} \qquad (7.7.14)$$

Equations (7.2.12)-(7.2.14) describe the rigid body motions of the vehicle about the nominal trajectory.

The bending equations are derived by the application of simple beam analysis to the booster which is considered to be a slender beam with uniform mass and stiffness. The model for each normalized bending mode is assumed to be a linear second order lightly damped oscillator with a forcing term proportional to the engine gimbal angle.[49]

$$\ddot{\eta}_i + 2 \, \zeta_i \, \omega_i \, \dot{\eta}_i + \omega_i^2 \, \eta_i = R' \, \frac{y_i'(x_\beta)}{M_i} \beta$$

To determine the actual bending at a point along the centerline of the vehicle, η_i must be multiplied by the mode slope coefficient corresponding to that point and the i^{th} bending mode.

The pitch and pitch rate gyros are located at specified points on the vehicle and measure local movement composed of rigid and bending motions. For this study it was assumed that the first three modes dominate, hence the pitch gyro output is

$$\phi_D = \phi + \sum_{i=1}^{3} y_i'(x_D)\, \eta_i$$

and the rate gyro output is

$$\dot{\phi}_R = \dot{\phi} + \sum_{i=1}^{3} y_i'(x_R)\, \dot{\eta}_i$$

where x_D = 79.8 meters and x_R = 67.3 meters are the position and rate gyro locations respectively, measured from the gimbal plane of the vehicle.

In summary, the linearized equations of motion which describe the vehicle are given below.

$$\ddot{Y}_n = \frac{(F + R' - D)}{m}\,\phi + \frac{N'}{m}\,\alpha + \frac{R'}{m}\,\beta \tag{7.2.15}$$

$$\ddot{\phi} = -\frac{R'\, l_{cg}}{I}\,\beta - \frac{N'\, l_{cp}}{I}\,\alpha \tag{7.2.16}$$

$$\ddot{\eta}_i + 2\,\zeta_i\,\omega_i\,\dot{\eta}_i + \omega_i^2\,\eta_i = R'\,\frac{y_i'(x_\beta)}{M_i}\,\beta \qquad i = 1,\ 2,\ 3 \tag{7.2.17}$$

$$\alpha - \alpha_w = \phi - \frac{\dot{Y}_n}{v} \tag{7.2.18}$$

$$\phi_D = \phi + \sum_{i=1}^{3} y_i'(x_D)\, \eta_i \tag{7.2.19}$$

$$\dot{\phi}_R = \dot{\phi} + \sum_{i=1}^{3} y_i'(x_R)\, \dot{\eta}_i \tag{7.2.20}$$

7.3 Control Structure

Current control schemes use a feedback structure employing only pitch and pitch rate information which is obtained by filtering the gyro outputs. This work proposes a new approach in which the actual sensor outputs are fed back without attempting to filter out the individual bending frequencies. A second order low pass filter is used as a forward loop compensator in order to roughly separate the rigid and bending motions. The outputs of the gyros are fed back to the input of the filter as shown in Fig. 7.4.

$$\beta_c = - k_1 \phi_D - k_2 \dot{\phi}_R \tag{7.3.1}$$

The filter chosen for this study had the following transfer function.[45]

$$\frac{\beta(s)}{\beta_c(s)} = \frac{50}{s^2+10s+50}$$

where the breakpoint was chosen to fall between the lowest bending frequency and highest slosh frequency.

The differential equation describing the filter is given by

$$\ddot{\beta} + 10\,\dot{\beta} + 50\,\beta = 50\,\beta_c \tag{7.3.2}$$

7.4 State Equations

The equations of motion have been written using variables which relate the movements of the vehicle to the nominal co-ordinate system. This viewpoint was taken since it is desired to regulate the motion of the vehicle about the nominal trajectory and hence drive these variables to zero.

There are two basic philosophies guiding the altitude control design, minimum drift and load relief. In the former, the objective is to keep the vehicle as close as possible to the nominal trajectory. However the excitation of the bending frequencies results in bending motions which must be limited in

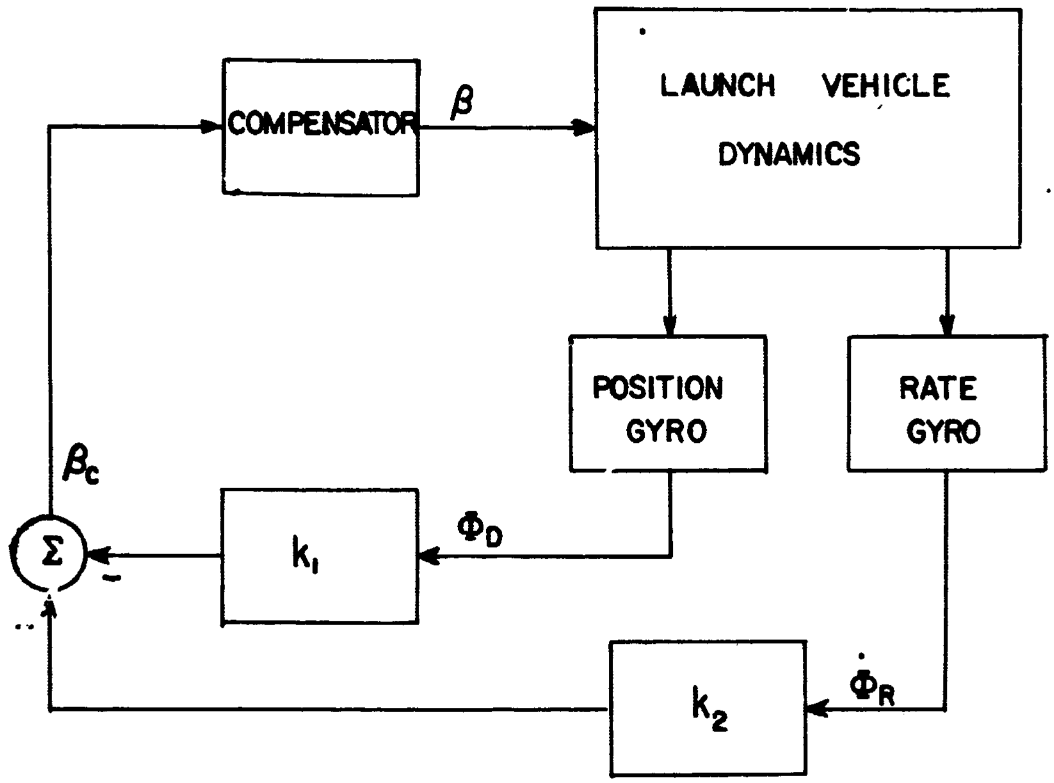


FIGURE 7.4 LAUNCH VEHICLE BLOCK DIAGRAM

order to preserve the structural integrity of the vehicle; hence the latter approach. These two approaches are by nature somewhat in conflict. A design objective of this study was to insure that the allowable bending moments did not exceed certain limits over the entire flight of the vehicle despite inaccurate knowledge of the bending frequencies. Since the bending moment is a function of the gimbal angle, β, and the angle-of-attack, α, the angle-of-attack was chosen as a state variable instead of the position variable Y_n. With a proper choice of weighting on α and β the SOC procedure may be used to limit the bending moment.

One possible choice of state variables is indicated below where for convenience only one bending mode is considered.

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} = \begin{bmatrix} \phi \\ \dot{\phi} \\ \alpha \\ \eta_1 \\ \dot{\eta}_1 \\ \beta \\ \dot{\beta} \end{bmatrix} \qquad (7.4.1)$$

The state variable formulation requires that a system of first order differential equations describing the states be derived. In order to eliminate Y_n and derive an equation describing α, multiply Eq. (7.2.18) by v and differentiate with respect to time.

$$\ddot{Y}_n = v\,\dot{\phi} + \dot{v}\,\phi - \dot{v}(\alpha - \alpha_w) - v(\dot{\alpha} - \dot{\alpha}_w)$$

This equation and Eq. (7.2.15) are used to eliminate Y_n and the resulting equation is solved for $\dot{\alpha}$.

$$\dot{\alpha} = -\left(\frac{F + R' - D}{mv} - \frac{\dot{v}}{v}\right)\phi + \dot{\phi} - \left(\frac{N'}{mv} + \frac{\dot{v}}{v}\right)\alpha - \frac{R'}{mv}\beta + \left(\frac{\dot{v}}{v}\alpha_w + \dot{\alpha}_w\right)$$

This equation along with Eq. (7.2.16) and (7.3.2) are used to formulate the state variable model.

$$\underline{\dot{x}} = A\,\underline{x} + \underline{b}\,\beta_c + \underline{v}(t); \qquad \underline{x}(t_o) = \underline{c} \tag{7.4.2}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{N' l_{cp}}{I} & 0 & 0 & -\frac{R' l_{cg}}{I} & 0 \\ -\left(\frac{F+R'-D}{mv} - \frac{\dot{v}}{v}\right) & 1 & -\left(\frac{N'}{mv} + \frac{\dot{v}}{v}\right) & 0 & 0 & -\frac{R'}{mv} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\omega_1^2 & -2\rho_1\omega_1 & \frac{R' y_1(x_\beta)}{M_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -50 & -10 \end{bmatrix}$$

$$\underline{b} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 50 \end{bmatrix}; \qquad \underline{v}(t) = \begin{bmatrix} 0 \\ 0 \\ \frac{\dot{v}}{v}\alpha_w + \dot{\alpha}_w \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The outputs of the system, $\underline{y}$, that is the quantities measured by the sensors are given by

$$\underline{y} = C\,\underline{x}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & y_1'(x_D) & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & y_1'(x_R) & 0 & 0 \end{bmatrix}$$

It is possible to redefine the state vector so that the measureable quantities appear as states. This new formulation is consistent with the SOC approach in which only the measurable or available states of the state vector are fed back. Define the following state vector where again only one bending mode is considered.

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} = \begin{bmatrix} \phi_D \\ \dot{\phi}_R \\ \alpha \\ \eta_1 \\ \dot{\eta}_1 \\ \beta \\ \dot{\beta} \end{bmatrix}$$

where

$$\phi_D = \phi + y_1'(x_D)\,\eta_1$$

$$\dot{\phi}_R = \dot{\phi} + y_1'(x_R)\,\dot{\eta}_1$$

The first order differential equations describing these states are derived by rearrangement of Eq. (7.4.2). It is assumed that the bending coefficients are time invariant; this is a reasonable approximation for the first bending mode or for a fixed time point model.

$$\dot{\eta}_1 = \dot{\eta}_1 \quad (7.4.3)$$

$$\ddot{\eta}_1 = -\omega_1^2 \eta_1 - 2 \zeta_1 \omega_1 \dot{\eta}_1 + R' \frac{y_1'(x_\beta)}{M_1} \beta \quad (7.4.4)$$

$$\dot{\phi}_D = \dot{\phi} + y_1'(x_D) \dot{\eta}_1 = \dot{\phi}_R + (y_1'(x_D) - y_1'(x_R)) \dot{\eta}_1 \quad (7.4.5)$$

$$\ddot{\phi}_R = \ddot{\phi} + y_1'(x_R) \ddot{\eta}_1 = -\frac{N' l_{cp}}{I} \alpha - \frac{R' l_{cg}}{I} \beta - y_1'(x_R)\, \omega_1^2 \eta_1 - 2 \zeta_1 \omega_1 y_1'(x_R) \dot{\eta}_1 + y_1'(x_R)\, R' \frac{y_1(x_\beta)}{M_1} \beta \quad (7.4.6)$$

$$\dot{\alpha} = -\left(\frac{F + R' - D}{mv} - \frac{\dot{v}}{v}\right) \phi + \dot{\phi} - \left(\frac{N'}{mv} + \frac{\dot{v}}{v}\right) \alpha - \frac{R'}{mv} \beta + \frac{\dot{v}}{v} \alpha_w + \dot{\alpha}_w$$

or

$$\dot{\alpha} = -\left(\frac{F + R' - D}{mv} - \frac{\dot{v}}{v}\right) \phi_D + \dot{\phi}_R - \left(\frac{N'}{m} + \frac{\dot{v}}{v}\right) \alpha + y_1'(x_D)\left(\frac{F + R' - D}{mv} - \frac{\dot{v}}{v}\right)\eta_1 + \frac{\dot{v}}{v} \alpha_w + \dot{\alpha}_w - y_1'(x_R) \dot{\eta}_1 - \frac{R'}{mv} \beta \quad (7.4.7)$$

These equations plus those describing the filter states can be written in a more compact form with the state variable notation.

$$\dot{\underline{x}} = A\, \underline{x} + \underline{b}\, \beta_c + \underline{v}(t); \qquad \underline{x}(t_o) = \underline{c} \quad (7.4.8)$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & a_{15} & 0 & 0 \\ 0 & 0 & a_{23} & a_{24} & a_{25} & a_{26} & 0 \\ a_{31} & 1 & a_{33} & a_{34} & a_{35} & a_{36} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & a_{54} & a_{55} & a_{56} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -50 & -100 \end{bmatrix}$$

$$a_{31} = -\left(\frac{F + R' - D}{mv} - \frac{\dot{v}}{v}\right) ; \qquad a_{23} = -\frac{N' \, l_{cp}}{I}$$

$$a_{33} = -\left(\frac{N'}{mv} + \frac{\dot{v}}{v}\right); \qquad a_{24} = -y_1'(x_R)\,\omega_1^{\,2}$$

$$a_{34} = y_1'(x_D)\left(\frac{F+R'-D}{mv} - \frac{\dot{v}}{v}\right) ; \qquad a_{54} = \omega_1^{\,2}$$

$$a_{15} = y_1'(x_D) - y_1'(x_R) ; \qquad a_{25} = -2\,\zeta_1\,\omega_1\,y_1'(x_R)$$

$$a_{35} = -y_1'(x_R) \qquad a_{55} = -2\,\zeta_1\,\omega_1$$

$$a_{26} = -\frac{R' \, l_{cg}}{I} + y_1'(x_R)\,R'\,\frac{y_1'(x_\beta)}{M_1} ; \qquad a_{36} = R'\,\frac{y_1'(x_\beta)}{M_1}$$

$$a_{56} = R'\,\frac{y_1'(x_\beta)}{M_1}$$

$$\underline{b} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 50 \end{bmatrix} ; \quad \underline{v}(t) = \begin{bmatrix} 0 \\ 0 \\ \frac{\dot{v}}{v}\alpha_w + \dot{\alpha}_w \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

These two representations are equivalent and may be used interchangeably.

7.5 Computational Considerations

Various engineering considerations require that the feedback control law employ constant values of feedback gains. The solutions to the SOC problems with time invariant models will have this property. The nominal flight of the booster extends from liftoff at $t = 0.$ to shutdown of the first stage at $t = 140.$ seconds. The model of the booster for this trajectory is time-varying; it was discovered that a suitable time invariant model could be generated by freezing the coefficients at $t = 80$. This approach proved to be satisfactory since designs made on the basis of this model provided adequate control for the time-varying model over the entire trajectory. Appendix F contains a table with the parameters as a function of the trajectory flight time.

The simulations were carried out on the R.P.I. model 360/50 digital computer using a fourth order Gill version of the Runge Kutta algorithm. For the time-varying simulations, linear interpolation was used to obtain the unspecified values of the model parameters. The acceptability of the designs was judged by initial condition responses of the fixed time point model. To provide a more realistic test of the proposed designs, time-varying simulations over the entire flight in the presence of a realistic wind disturbance were made.

From Eq. (7.4.8) it is clear that the only external disturbance acting on the vehicle is wind. The wind is assumed to change the apparent angle-of-attack by an amount equal to α_w. This angle is related to the velocity of the vehicle, $\underline{v}$, and the velocity of the wind, $\underline{v}_w$. Figure 7.5 portrays the relationship between the velocity vectors when the booster is on its nominal trajectory. ($\alpha = \phi = 0$) From this figure it is clear that

$$\alpha_w = \frac{V_w \cos \chi_c}{v - V_w \sin \chi_c} \tag{7.5.1}$$

Thus by knowing the nominal trajectory parameters and the velocity of the wind it is possible to construct a realistic forcing function. From the data provided by Marshall Space Flight Center[48], a 95% synthetic wind profile was constructed as shown in Fig. 7.6. The 95% notation indicates that the magnitudes of these winds exceed those of 95% of the actual winds measured from May to November at Cape Kennedy. To further test the effectiveness of the control schemes, a wind gust was added to the profile in the region of maximum dynamic pressure (max. q). The wind induced angle-of-attack, α_w, obtained from this wind profile via Eq. (7.5.1) is indicated in Fig. 7.7.

7.6 Application of the SOC Techniques

7.6.1 Design Objectives

As described in Chapter IV, the SOC design procedure may be used to calculate linear feedback controllers for linear systems with unavailable states. Recall that the position and rate gyros measure a mixture of rigid and bending motions; angle-of-attack meters are available but their use is to be avoided if

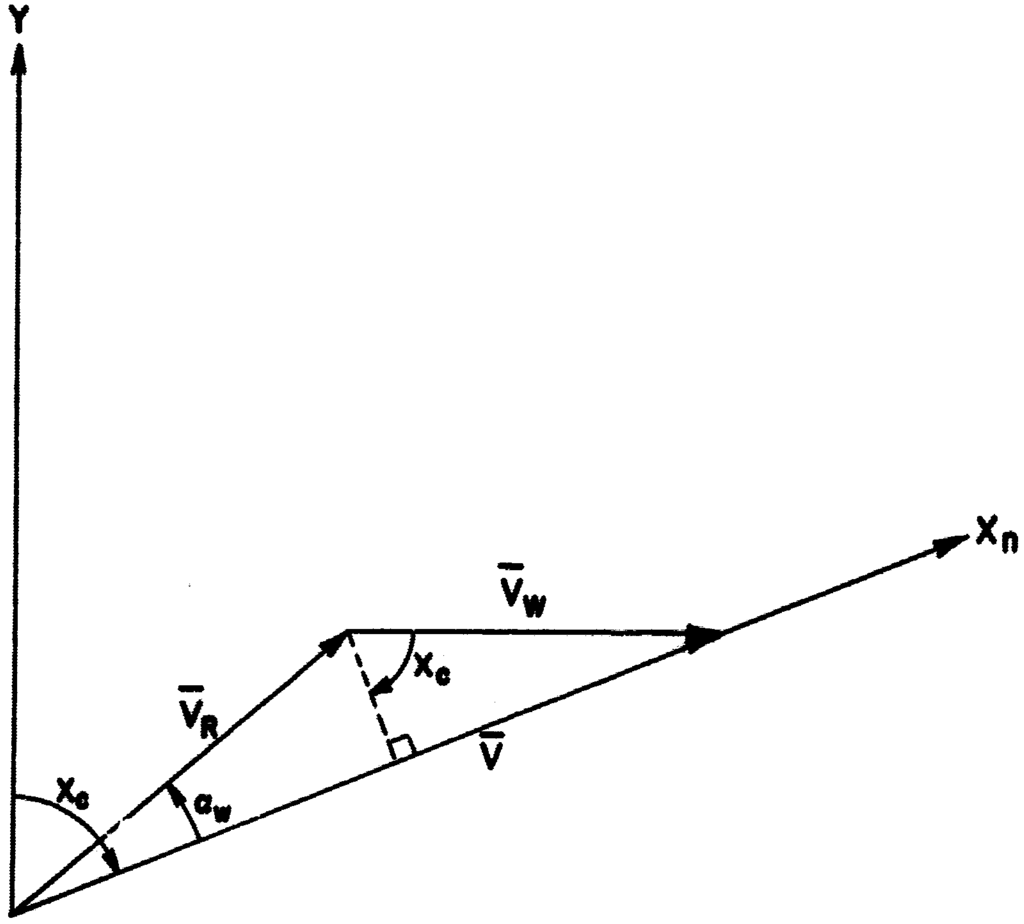


FIG. 7.5 WIND ANGLE OF ATTACK

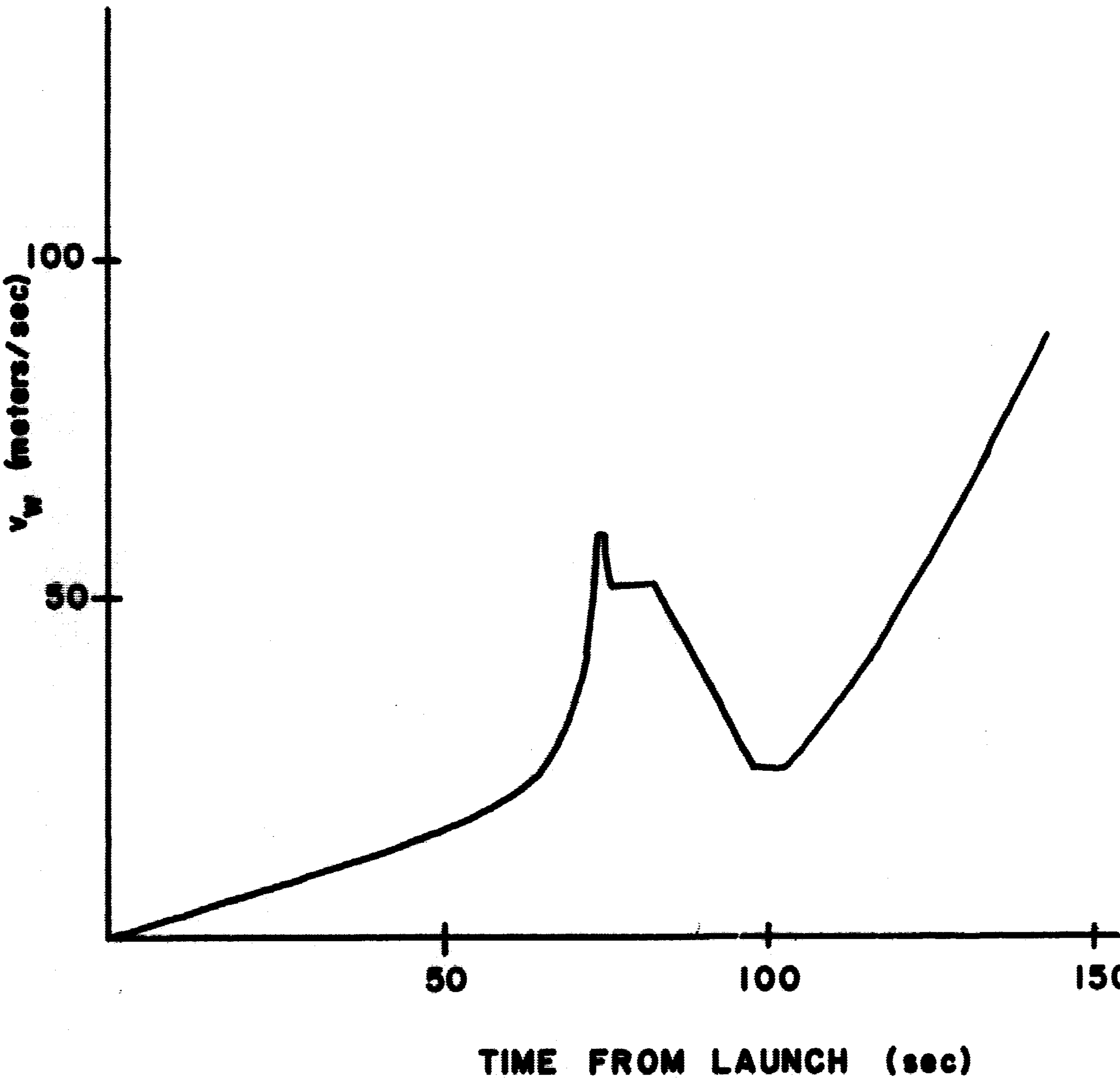


FIG. 7-6 SYNTHETIC WIND

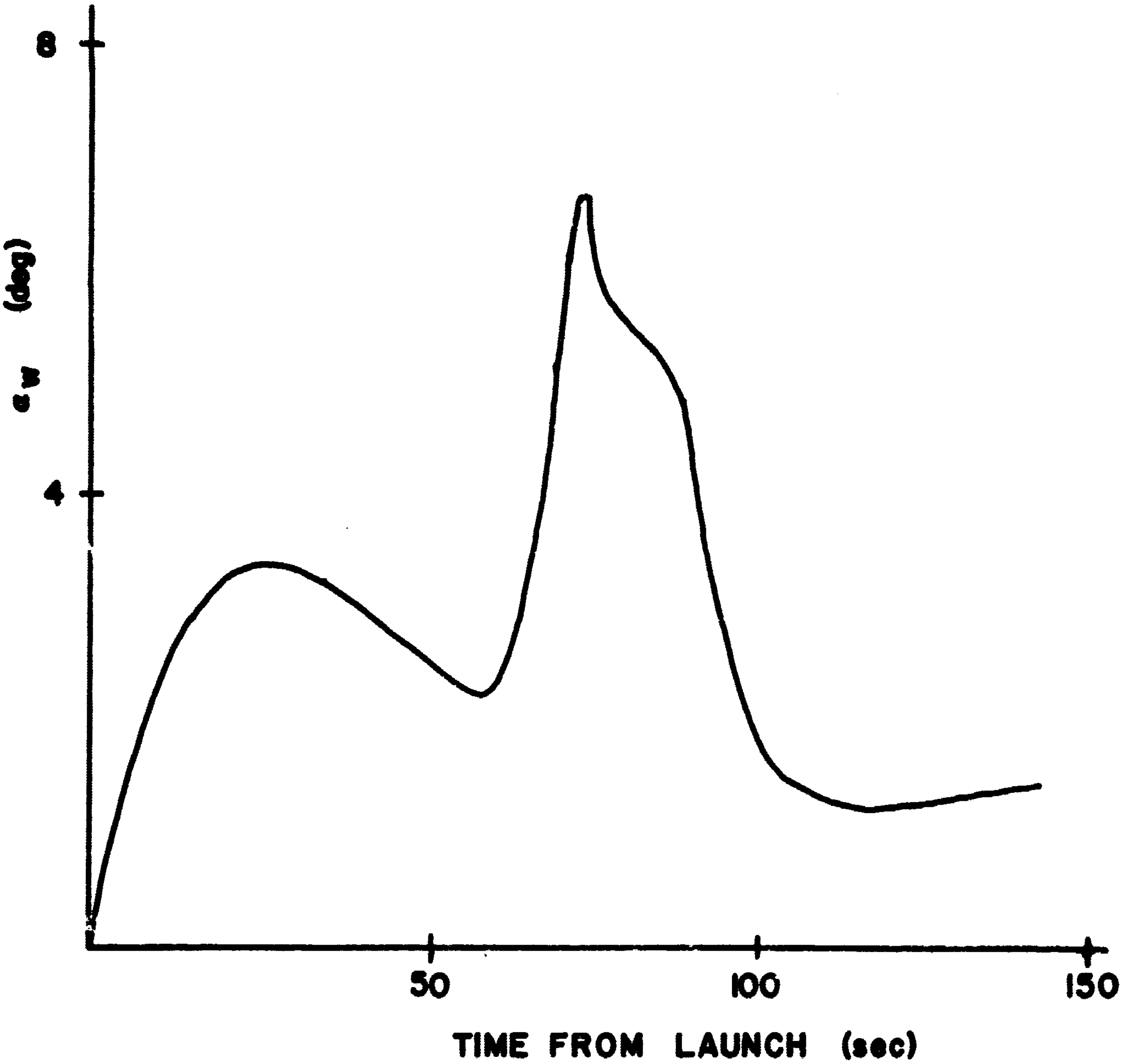


FIG. 7-7 WIND ANGLE OF ATTACK

possible. Consider the corrupted state model, that is the state vector which contains the gyro outputs. This formulation is consistent with the SOC approach when it is assumed that only the first two states of the state vector are available.

The actual design specifications are stated in terms of the time domain response and are summarized below.

General Requirements

1. Stable closed loop system with respect to the fixed time point model.
2. Well behaved initial condition responses.
3. Limits on the maximum absolute values of the states must be maintained for the duration of the wind forced time varying simulation.
4. At cut-off the pitch and pitch rate quantities must be small to allow smooth staging.

Specific Requirements

For the time varying simulation with the design wind the following limits must be maintained.

1. Engine deflection: $|\beta| < 5^\circ$
2. Engine deflection rate: $|\dot{\beta}| < 5^\circ$/second
3. Angle-of-attack: $|\alpha| < 10^\circ$
4. Pitch angle: $|\phi| < 10^\circ$
5. Engine cut-off: $|\phi| < 1^\circ$; $|\dot{\phi}| < 1^\circ$/second
6. Bending magnitude: $|\eta| < .25$ meters
7. Bending moment (Station 3256): $BM < 5.45 \times 10^5$ kg.m.

7.6.2 The SOC Design Procedure

In order to determine the effectiveness of the SOC design approach, it was applied to the launch vehicle problem. The SOC problem was formulated so that the feedback control law depended only on the noisy outputs of the two sensors. The SOCDES program was used in an automatic mode, that is a series of SOC problems were calculated with slightly different weightings. The results were analyzed and compared via the graphical aids described in Chapter IV.

Control Weighting Perturbations

To study the effect of variations in the control weighting, the reverse problem was solved for the optimal design of reference 45, which was obtained as a result of an "optimal" analog computer study, using the following set of weightings.

$$u = - k_1 \phi_D - k_2 \dot{\phi}_R$$

$$k_1 = - 0.8$$

$$k_2 = -0.8$$

$$Q = 10.0$$

$$S = \begin{bmatrix} s_{\phi_D} & 0 & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & s_{\dot{\phi}_R} & \cdot & & & & \cdot \\ \cdot & \cdot & s_\alpha & \cdot & & & \cdot \\ \cdot & & \cdot & s_{\eta_1} & \cdot & & \cdot \\ \cdot & & & \cdot & s_{\dot{\eta}_1} & \cdot & \cdot \\ \cdot & & & & \cdot & s_\beta & 0 \\ 0 & \cdot & \cdot & \cdot & \cdot & 0 & s_{\dot{\beta}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdot & \cdot & 0 \\ 0 & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & & \cdot & \cdot & \cdot \\ \cdot & & & \cdot & \cdot \\ \cdot & & \cdot & \cdot & 0 \\ 0 & \cdot & \cdot & 0 & 1 \end{bmatrix}$$

In Fig. 7.8 the k locus is shown, the entire locus may be obtained in about five minutes of 360/50 execution time. The solid line indicates the region of stable gains. In Fig. 7.10 the root locus corresponding to this k locus is presented. The same scale increments are used for all curves. Parts a and c show rigid body poles, part b corresponds to the first mode poles, and the filter poles are graphed in part d. Note the interesting portion of the locus of part c in which the rigid complex roots approach the real axis, remain there for a while and then branch out into the complex region again. This result can be obtained by a conventional root locus analysis but not without considerable effort.[50] The effect of the variation of Q on the integral square control effort is pictured in Fig. 7.9; as expected the control effort increases as the control weighting is decreased. The examination of these figures points out a basic property of the booster.

Result:

The design of the launch vehicle altitude control system involves a tradeoff between relative stability of the bending modes, measured by the real part of the first mode complex root pair, and the rigid body damping ratio. (See Fig. 7.11)

As the control weighting is decreased the relative stability is increased and rigid body damping is decreased. This tradeoff appears throughout the study of this booster problem. If the bending frequences are below nominal then the bending poles tend to migrate toward the imaginary axis and instability. Although these designs were calculated for a seven state fixed time point model at $t = 80$ sec., final evaluations were obtained by simulating the controls for

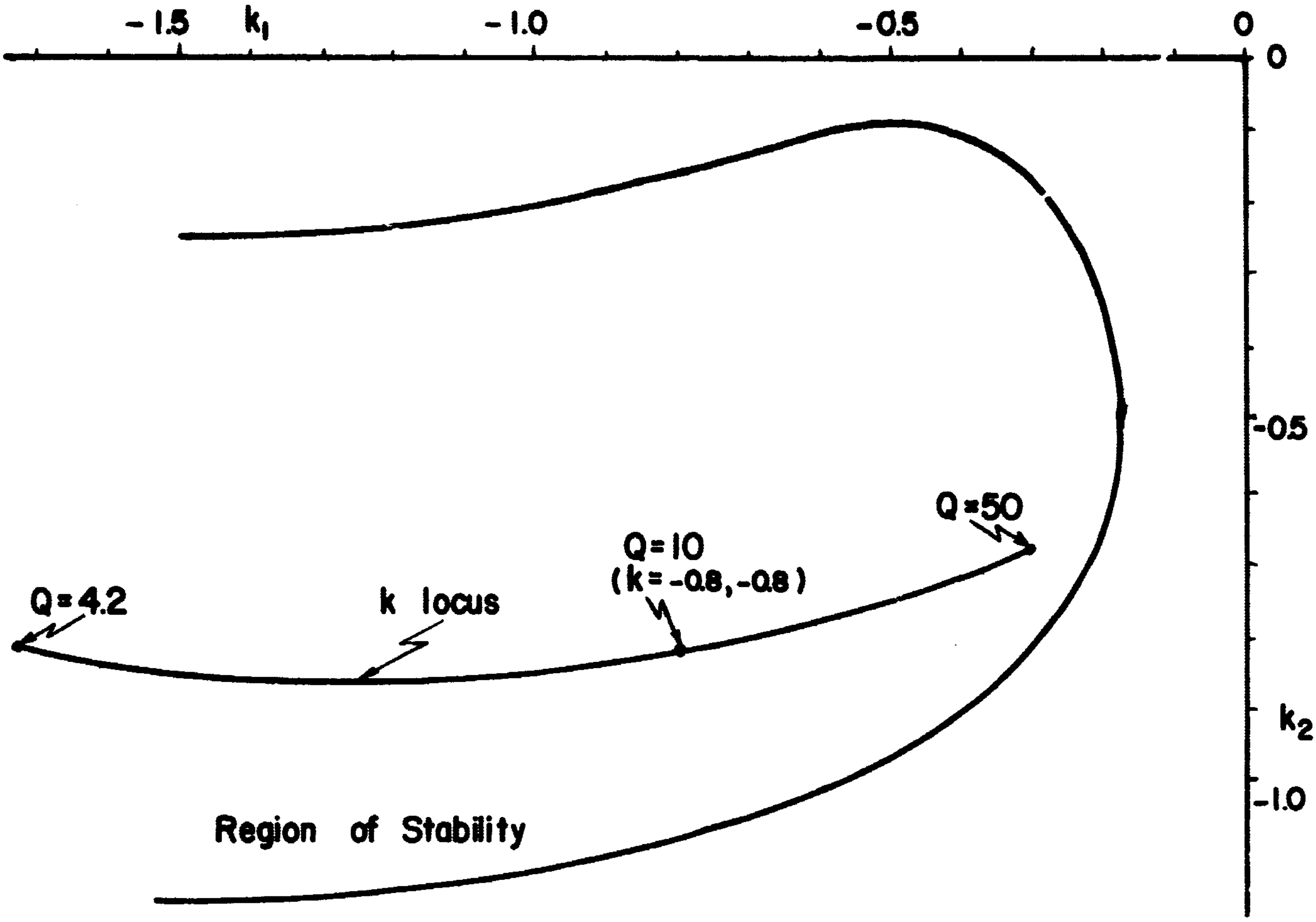


FIGURE 7.8 Q-k LOCUS

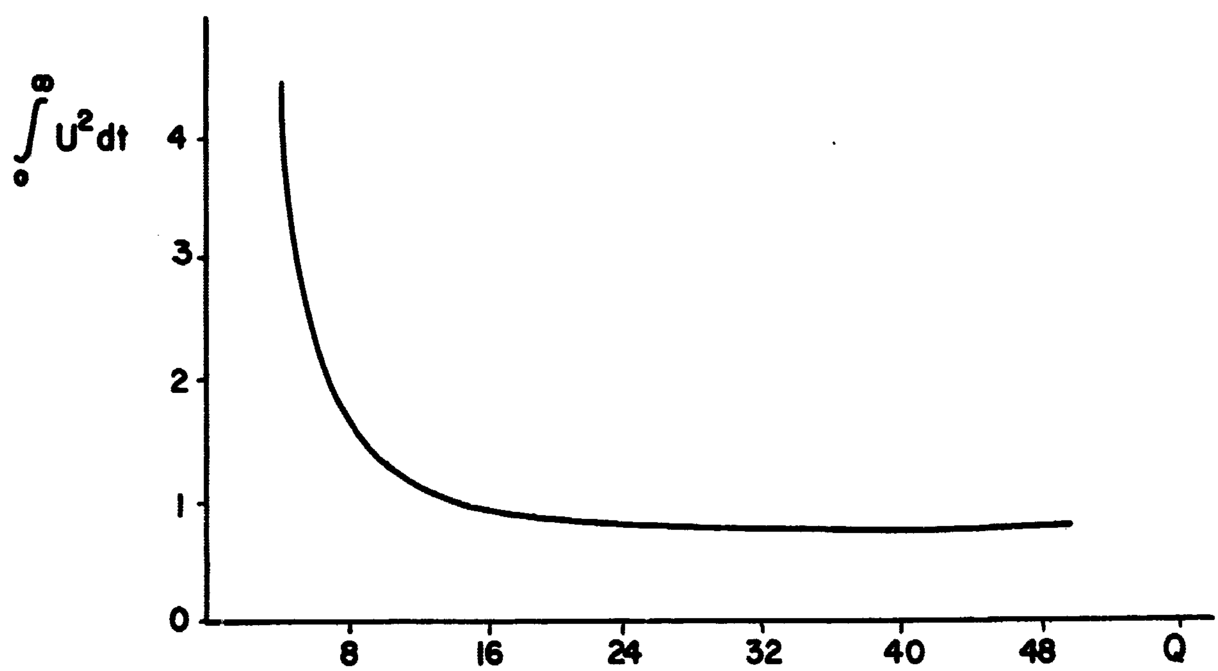


FIGURE 7.9 $\int_0^\infty U^2 dt$ vs. Q

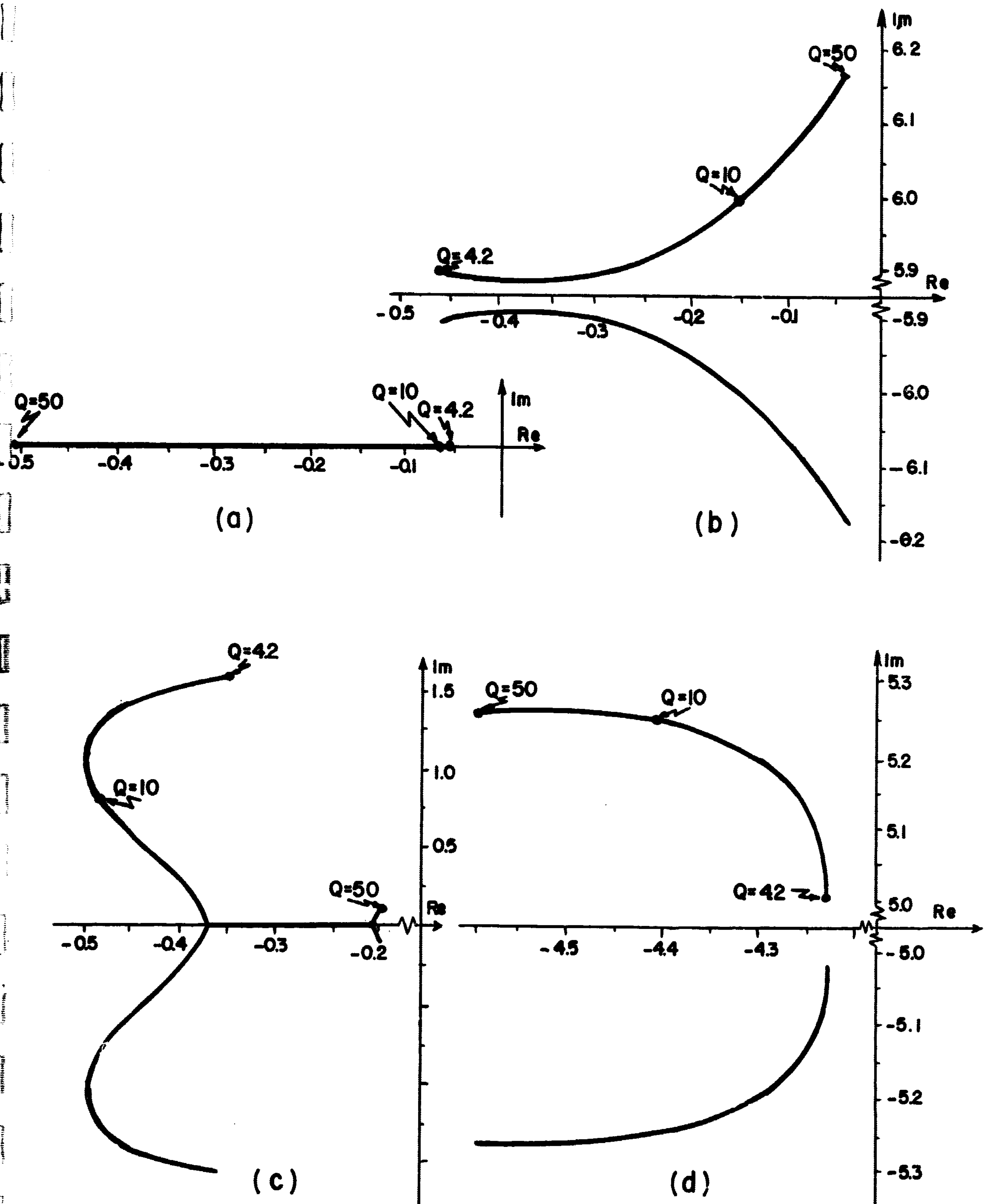


FIGURE 7.10 Q ROOT LOCUS

a time varying model with three bending modes and the design wind. These simulations corresponding to the various control laws were remarkably similar in shape, with the only major differences being the magnitude of the peaks. The nominal response is pictured in Fig. 7.22. To avoid the monotony of page after page of similar graphs, only a few responses are included along with tables containing values of peak magnitudes and integral square state values. For example, Table 7.1 indicates that responses corresponding to various points along the k locus are similar except that the peak value of the pitch decreases as Q decreases.

Insight into this problem may be obtained by varying the relative magnitudes of the state weightings and then varying the control weightings as indicated in Fig. 7.13. In this case the pitch state weighting is increased and the loci generated by reducing the control weighting. For this problem the entire stable gain space may be probed by changing the relationships between the weightings and generating the gain loci.

State Weighting Perturbations

A similar approach can be taken for state weighting loci. For example, the reverse problem was solved for the following set of weightings

$$S = \begin{bmatrix} 3 & 0 & \cdot & \cdot & \cdot & & 0 \\ 0 & 1 & \cdot & & & & \\ \cdot & \cdot & 1 & \cdot & & & \cdot \\ \cdot & & & 0 & & & \cdot \\ \cdot & & & \cdot & 0 & \cdot & \\ \cdot & & & & \cdot & 0 & 0 \\ 0 & \cdot & & \cdot & \cdot & 0 & 0 \end{bmatrix} ; \quad Q = 1.0$$

with

$$\underline{k} = \begin{bmatrix} -0.8 \\ -0.8 \end{bmatrix}$$

Q	k	max $\|\phi\|$	max $\|a\|$	max $\|\eta_1\|$	$\int_0^{140} \phi^2 dt$	$\int_0^{140} a^2 dt$	$\int_0^{140} \eta_1^2 dt$
30.0	-.394 -.692	2.47°	6.61°	.11 m.	1115.	$2.49\ 10^4$	1.91
10.0	-.800 -.800	1.16°	6.11°	.10 m.	197.	$2.23\ 10^4$	1.41
4.5	-1.588 -.830	0.55°	5.88°	.11 m.	32.8	$2.20\ 10^4$	1.28

TABLE 7.1 Q VARIATION SIMULATIONS

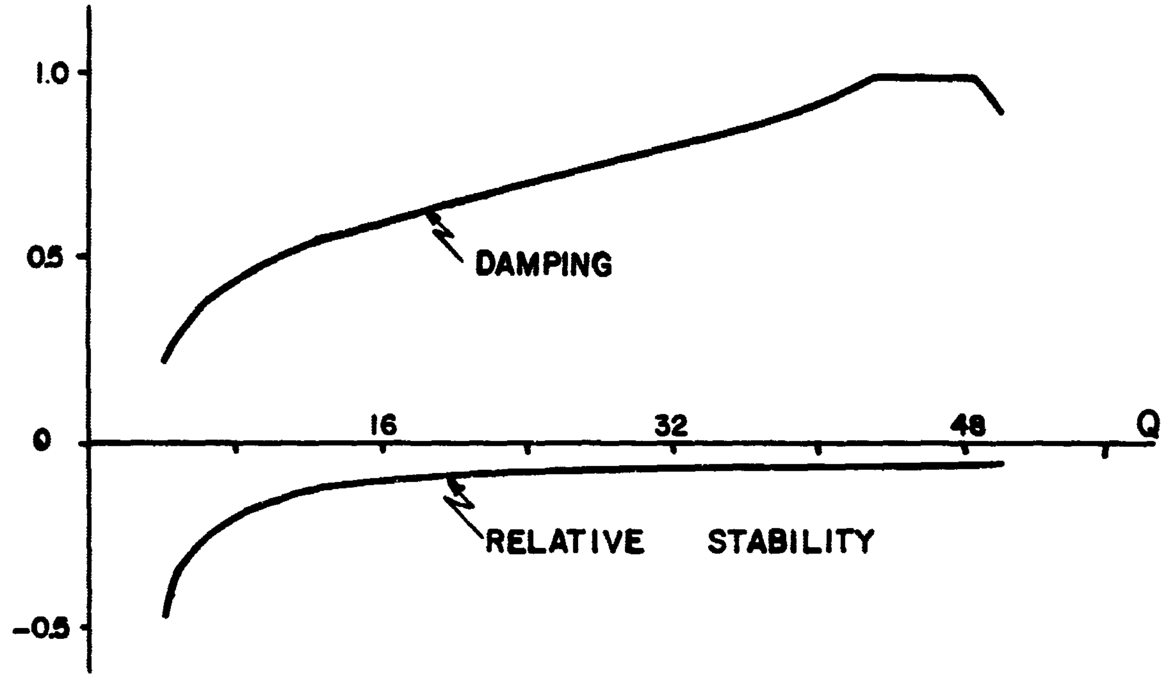


FIGURE 7.11 DAMPING and RELATIVE STABILITY vs. Q

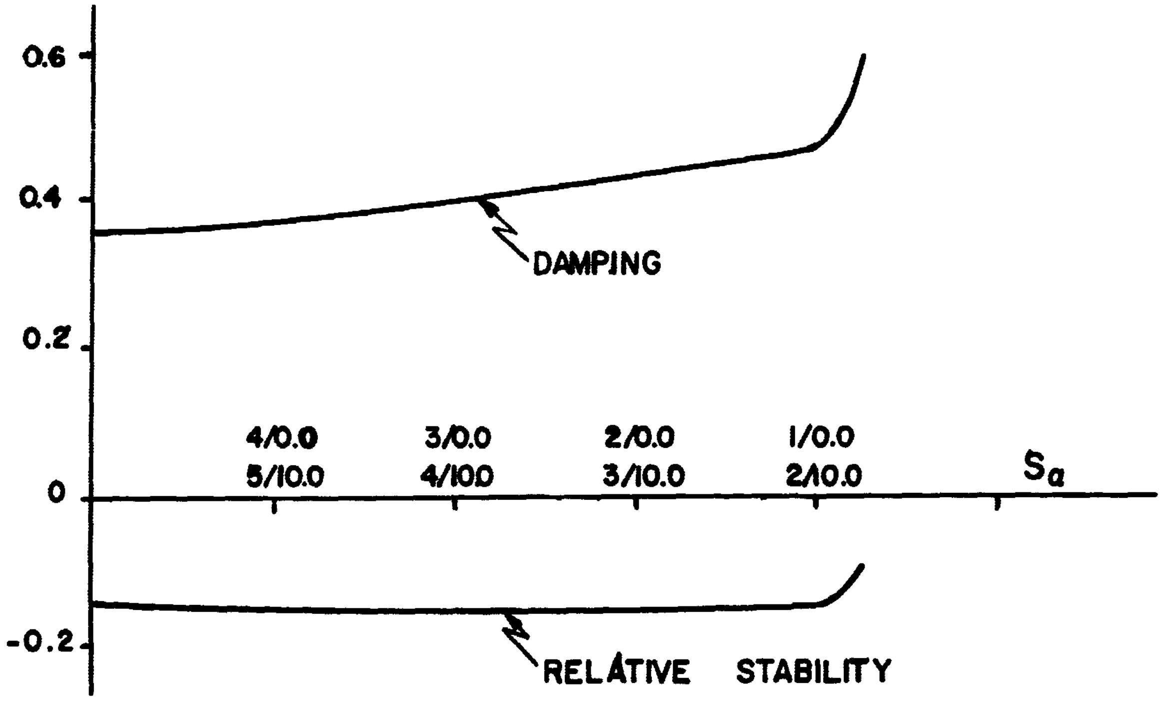


FIGURE 7.12 DAMPING and RELATIVE STABILITY vs. S_a

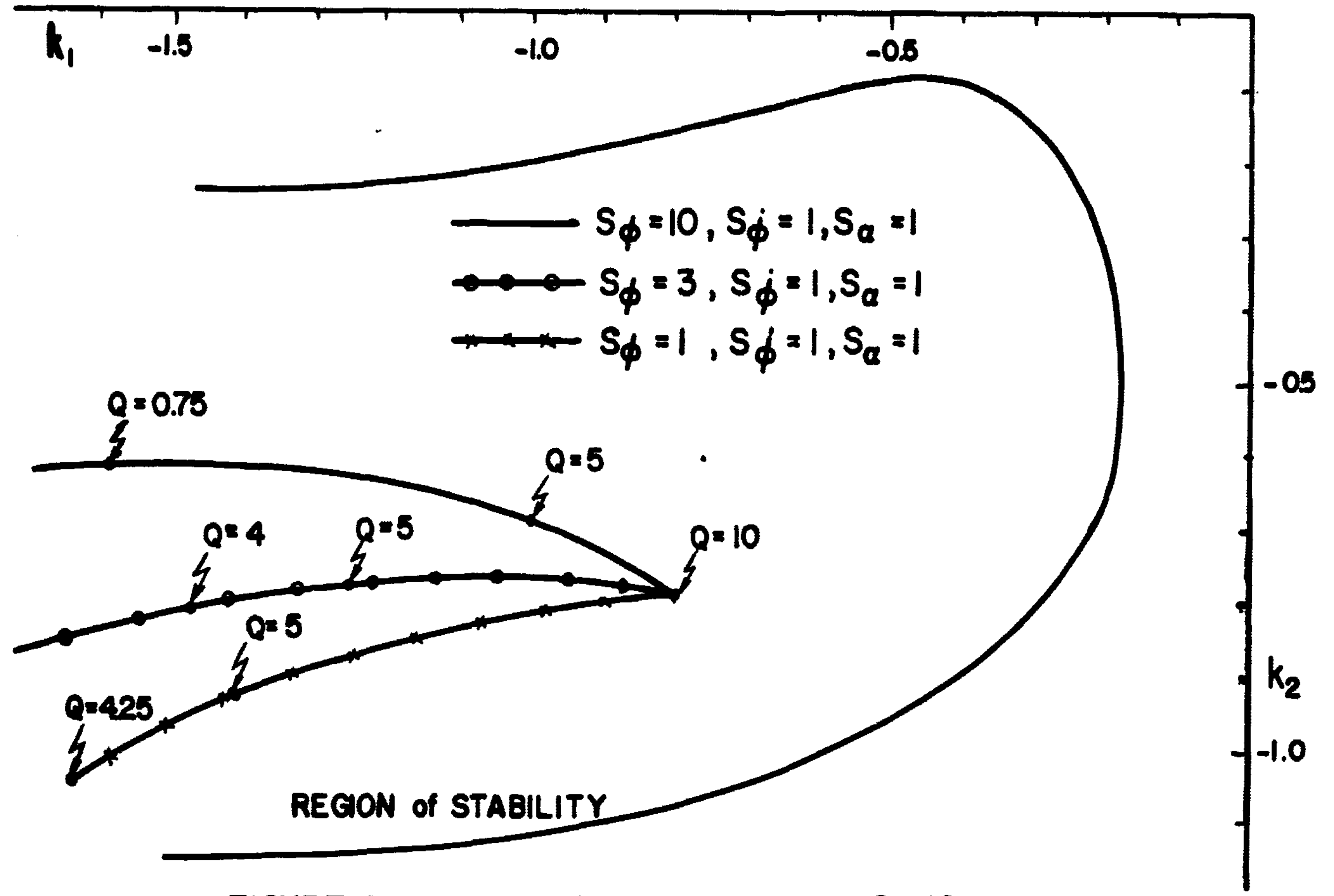


FIGURE 7.13 VARIOUS k ROOT LOCUS Q

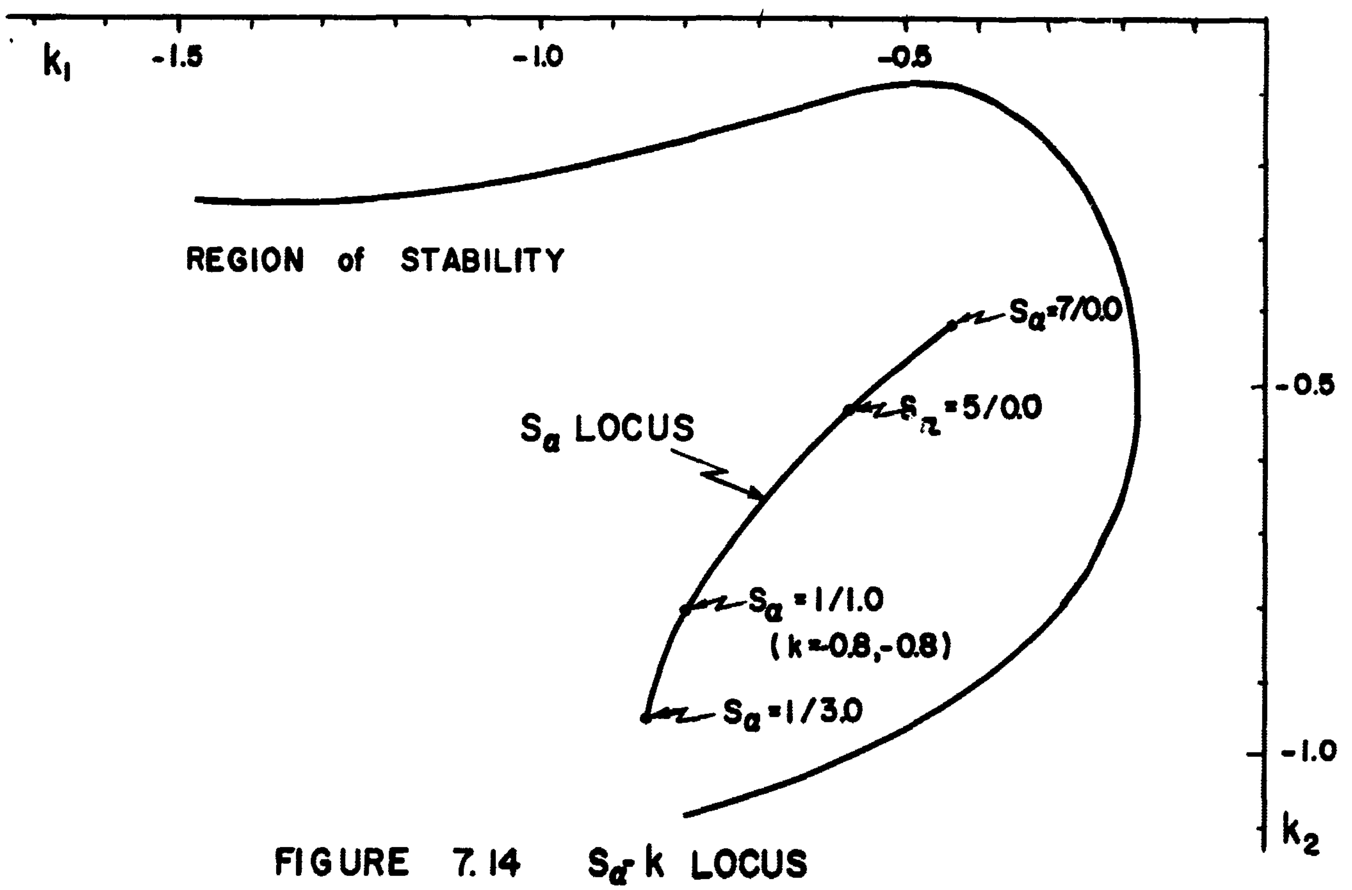


FIGURE 7.14 S_α-k LOCUS

and the s_α weighting was varied to generate the locus shown in Fig. 7.14. In general, as the gain locus approaches the stable gain boundary, the SOC equations become numerically sensitive and a new reverse problem may be solved and the perturbations continued to extend the locus. As the s_α weighting is decreased, the k locus moves upward; when $s_\alpha = 0$, the reverse problem is resolved for the following weightings.

$$S = \begin{bmatrix} 30 & 0 & \cdot & & \cdot & \cdot & 0 \\ 0 & 10 & \ddots & & & & \cdot \\ \cdot & \ddots & 10 & & & & \\ & & & 0 & \cdot & & \cdot \\ \cdot & & \cdot & & 0 & \cdot & \cdot \\ \cdot & & & \cdot & & 0 & 0 \\ 0 & \cdot & \cdot & \cdot & & 0 & 0 \end{bmatrix} \; ; \; Q = 10$$

and s_α is again decreased to zero. This process is repeated and the locus is extended. Again the tradeoff between damping and relative stability is evident as shown in Fig. 7.12. As s_α is decreased, the relative stability increases and rigid body damping decreases. In addition, the integral square value of α decreases as s_α increases. (Fig. 7.15) If different reverse problem weighting combinations are employed or other weightings are varied, different areas of the gain space are probed. The root locus corresponding to this k locus is shown in Fig. 7.17. The results of the full wind simulations are shown in Table 7.2. As the angle-of-attack weighting is decreased the peak value and integral square value of α decrease.

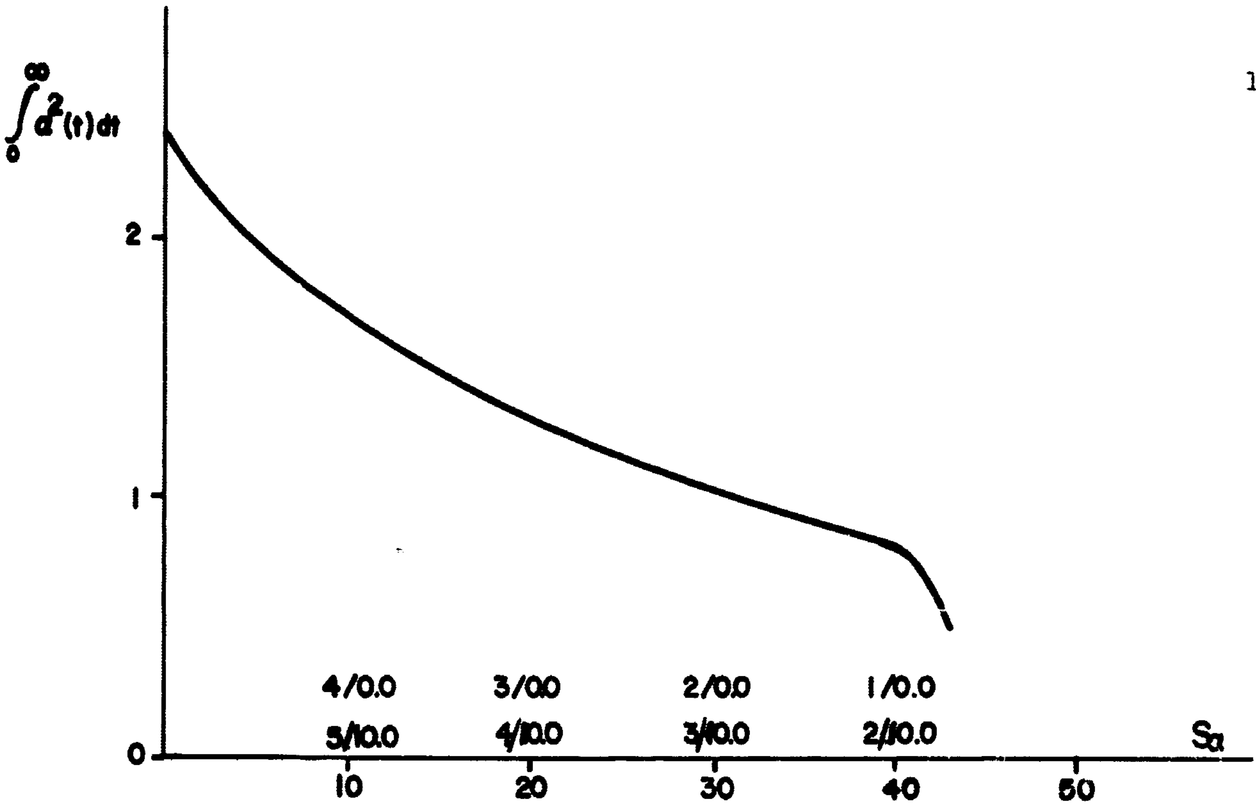


FIGURE 7.15 $\int_0^\infty \alpha^2 dt$ vs. S_α

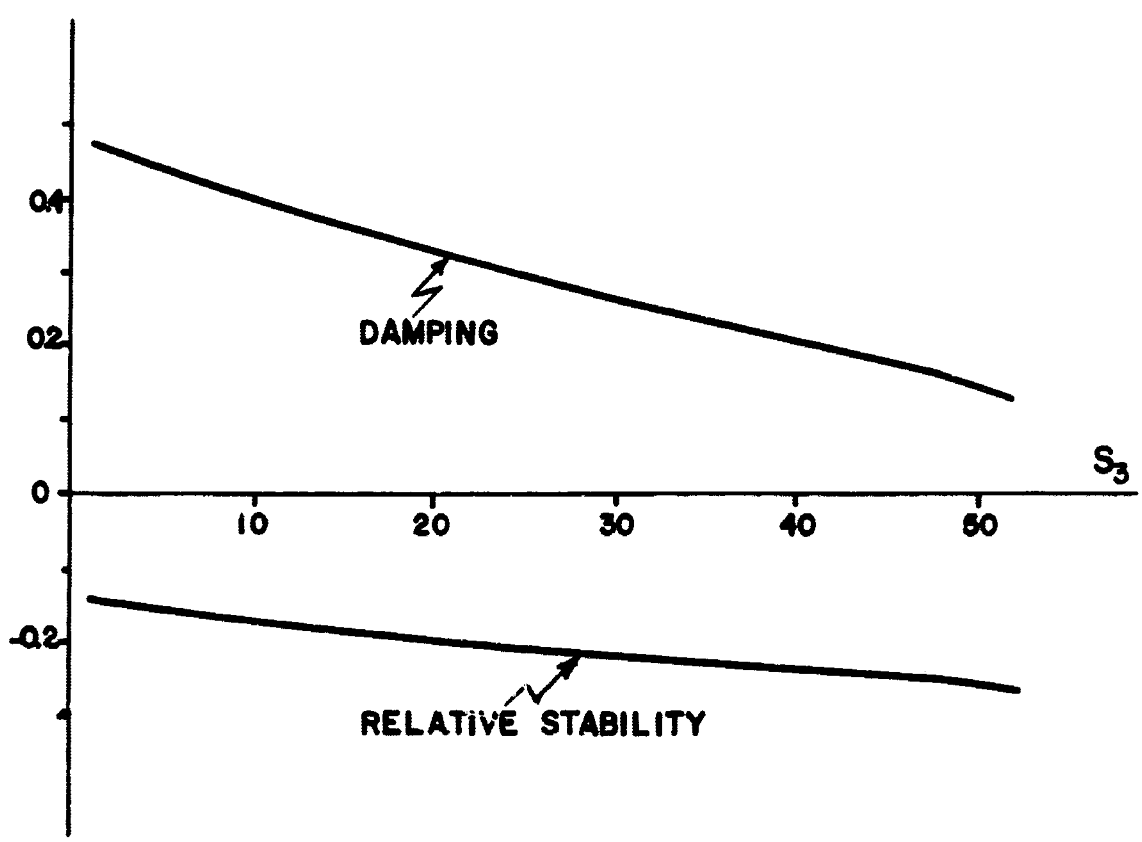


FIGURE 7.16 DAMPING and RELATIVE STABILITY vs. S_3

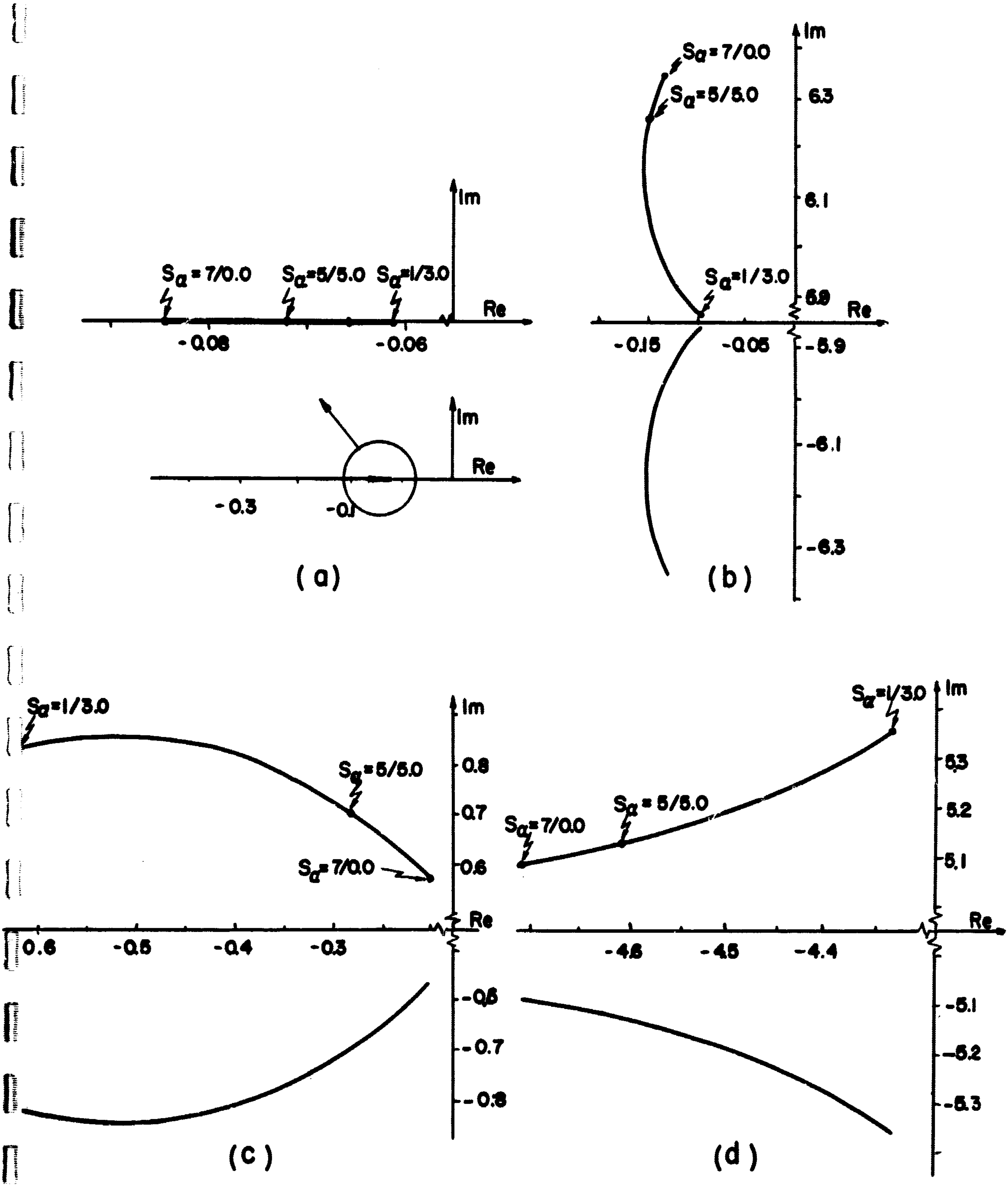


FIGURE 7.17 S_α ROOT LOCUS

S_a	k	max $\|\phi\|$	max $\|a\|$	max $\|\eta_1\|$	$\int_0^{140} \phi^2 dt$	$\int_0^{140} a^2 dt$	$\int_0^{140} \eta_1^2 dt$
1|3.0	-.849 -.941	1.05°	6.07°	.10 m.	164.	2.22 10^4	1.40
1|1.0	-.800 -.800	1.16°	6.11°	.10 m.	197.	2.23 10^4	1.41
4|10.	-.655 -.612	1.49°	6.23°	.11 m.	328.	2.25 10^4	1.51
6|5.0	-.522 -.482	2.03°	6.44°	.12 m.	558.	2.28 10^4	1.66
7|0.0	-.454 -.421	2.46°	6.62°	.12 m.	821.	2.32 10^4	1.79

TABLE 7.2 S VARIATION SIMULATIONS

7.6.3 Application of SOC Sensitivity

Using the method outlined in Chapter VI, it is possible for the first time to use sensitivity considerations in the design of control laws for realistic problems. For the launch vehicle problem, the parameters of concern are the bending frequencies. The SOC sensitivity problem was formulated as described in Chapter VI and the reverse problem was solved for the following set of weightings.

$$S_1 = \begin{bmatrix} s_{\phi_D} & 0 & \cdot & & \cdot & \cdot & 0 \\ 0 & s_{\dot{\phi}_R} & \cdot & & & & \cdot \\ \cdot & \cdot & s_\alpha & & \cdot & & \cdot \\ \cdot & & \cdot & s_{\eta_1} & & & \cdot \\ & & & & s_{\dot{\eta}_1} & \cdot & \cdot \\ \cdot & & & & \cdot & s_\beta & 0 \\ 0 & \cdot & \cdot & & \cdot & 0 & s_{\dot{\beta}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdot & & & \cdot & 0 \\ 0 & 1 & \cdot & & & & \cdot \\ \cdot & \cdot & 1 & \cdot & & & \cdot \\ \cdot & & \cdot & 0 & & \cdot & \cdot \\ & & & & 0 & & \cdot \\ \cdot & & & \cdot & & 0 & 0 \\ 0 & \cdot & & \cdot & \cdot & 0 & 0 \end{bmatrix}$$

$$S_3 = \begin{bmatrix} s_{Z_1} & 0 & \cdot & 0 \\ 0 & \ddots & & \cdot \\ \cdot & & \ddots & 0 \\ 0 & \cdot & 0 & s_{Z_7} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdot & & \cdot & \cdot & 0 \\ 0 & 1 & \cdot & & & & \cdot \\ \cdot & \cdot & 1 & \cdot & & & \\ & & \cdot & 0 & & & \\ \cdot & & & & 0 & \cdot & \cdot \\ & & & & \cdot & 0 & 0 \\ 0 & \cdot & \cdot & & \cdot & 0 & 0 \end{bmatrix}$$

and the nominal control law

$$u = -k_1 \phi_D - k_2 \dot{\phi}_R$$

$$k = \begin{bmatrix} -0.8 \\ -0.8 \end{bmatrix}$$

To obtain the gain root locus the sensitivity weighting is increased. (Fig. 7.19)

$$s_{Z_1} = s_{Z_2} = s_{Z_3} = s$$

The locus moves almost vertically indicating that k_1, the pitch gain, has little effect on the sensitivity of the system. Note the point $\underline{k} = \begin{bmatrix} -.5 \\ -.4 \end{bmatrix}$ which was obtained by the Analog Sensitivity Design (ASD) method.[45] In Fig. 7.18 the root locus is depicted while in Fig. 7.16 the damping and relative sensitivity curves are pictured. The desensitization is obtained by increasing the relative stability at the expense of the rigid body damping. This result is in contrast with the SOC sensitivity results for simple examples in which the magnitudes of the feedback gains were increased to "swamp" out the effect of the parameter. Intuitively, the inclusion of control effort weighting forces the SOC procedure to produce the more subtle result if one exists.

To place the SOC solutions in perspective, they are compared with the ASD result and the nominal control law. Evidence of the reduction in sensitivity can be obtained from a number of points of view. Figure 7.20 indicates that as the sensitivity weighting is increased the integral square of the sensitivity variables decreases. However, this curve does not indicate the accuracy of the sensitivity variables in modeling the actual desensitization of the trajectories.

The design objectives require that the control system maintain adequate control for bending frequency variations of ± 20%. An increase in bending frequency has a beneficial effect on the system performance since the relative stability is increased. However, the reduction of bending frequencies poses a serious problem. As shown in Fig. 7.21 the closed loop system for the

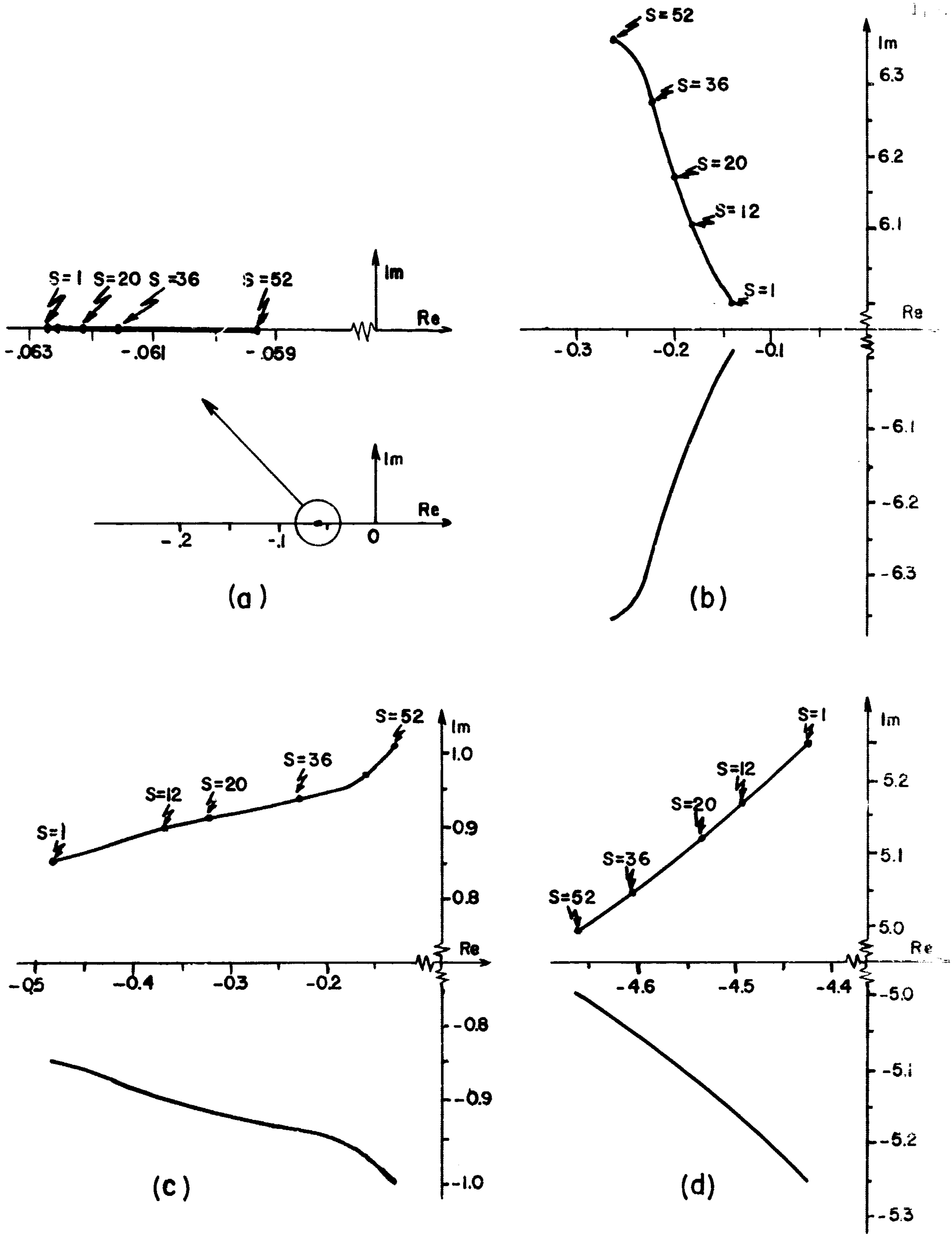


FIGURE 7.18 SENSITIVITY ROOT LOCUS

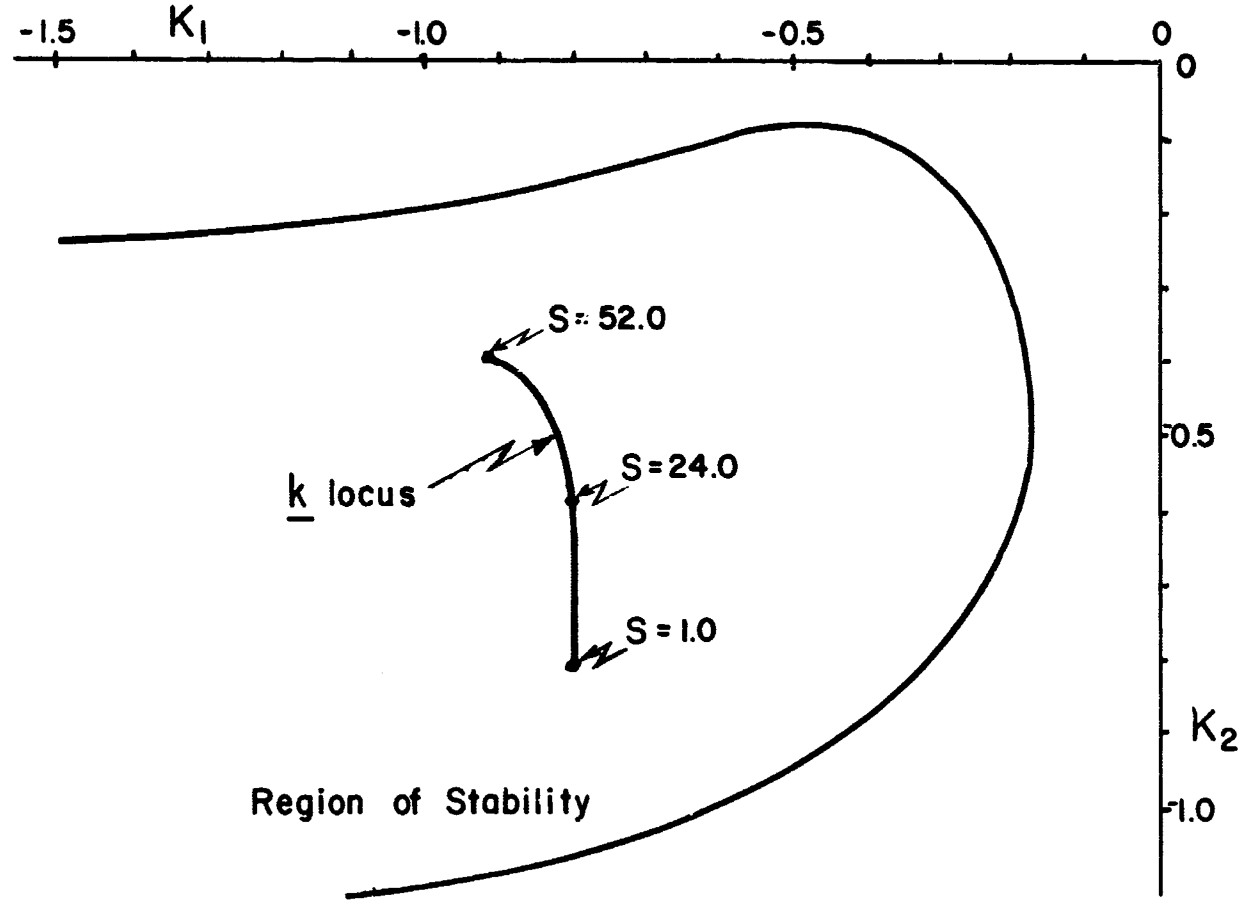


FIGURE 7.19 SENSITIVITY $\underline{k}$ LOCUS

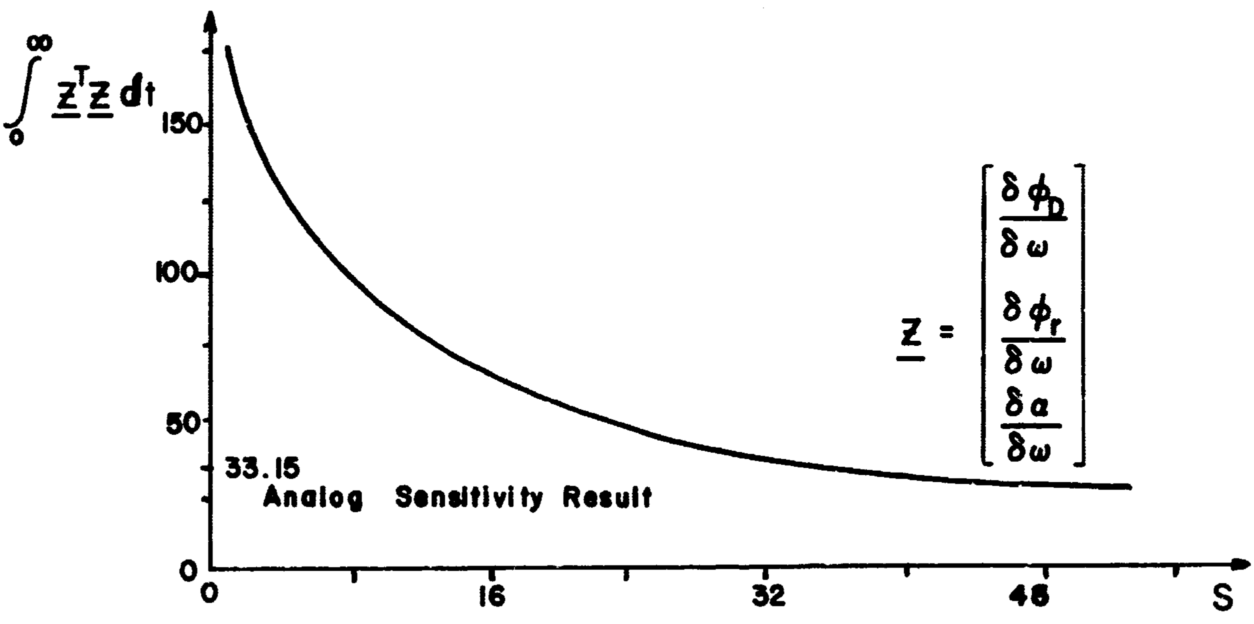


FIGURE 7.20 $\int_0^\infty \underline{z}^T \underline{z}\, dt$ vs. SENSITIVITY WEIGHTING

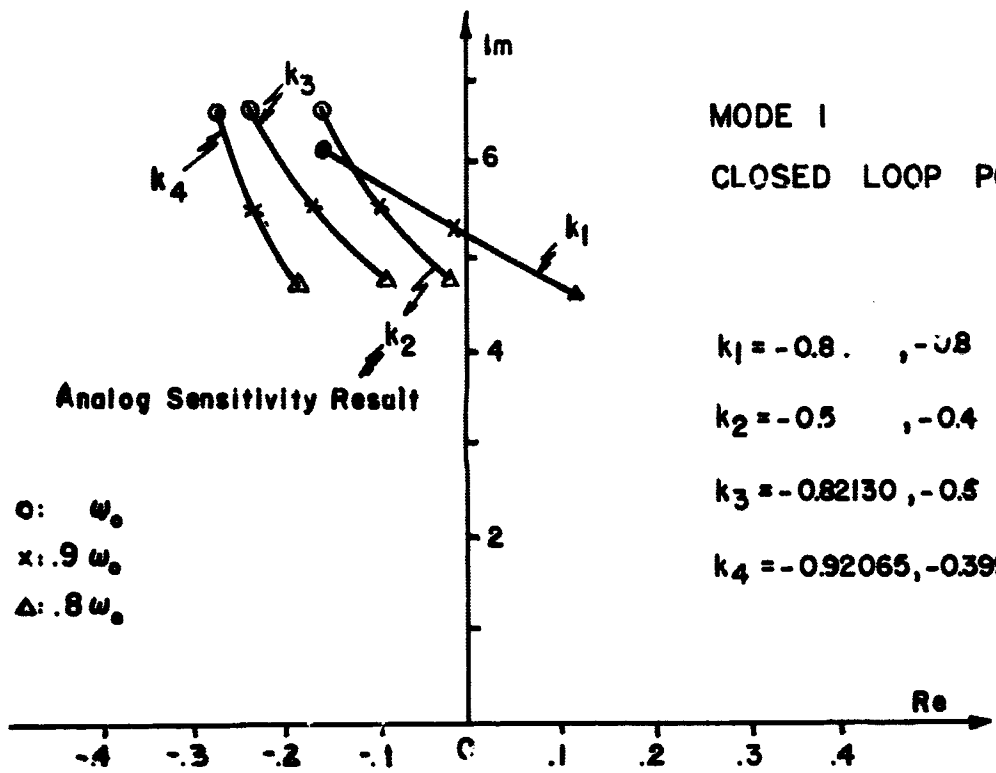


FIGURE 7.21 ω ROOT LOCUS

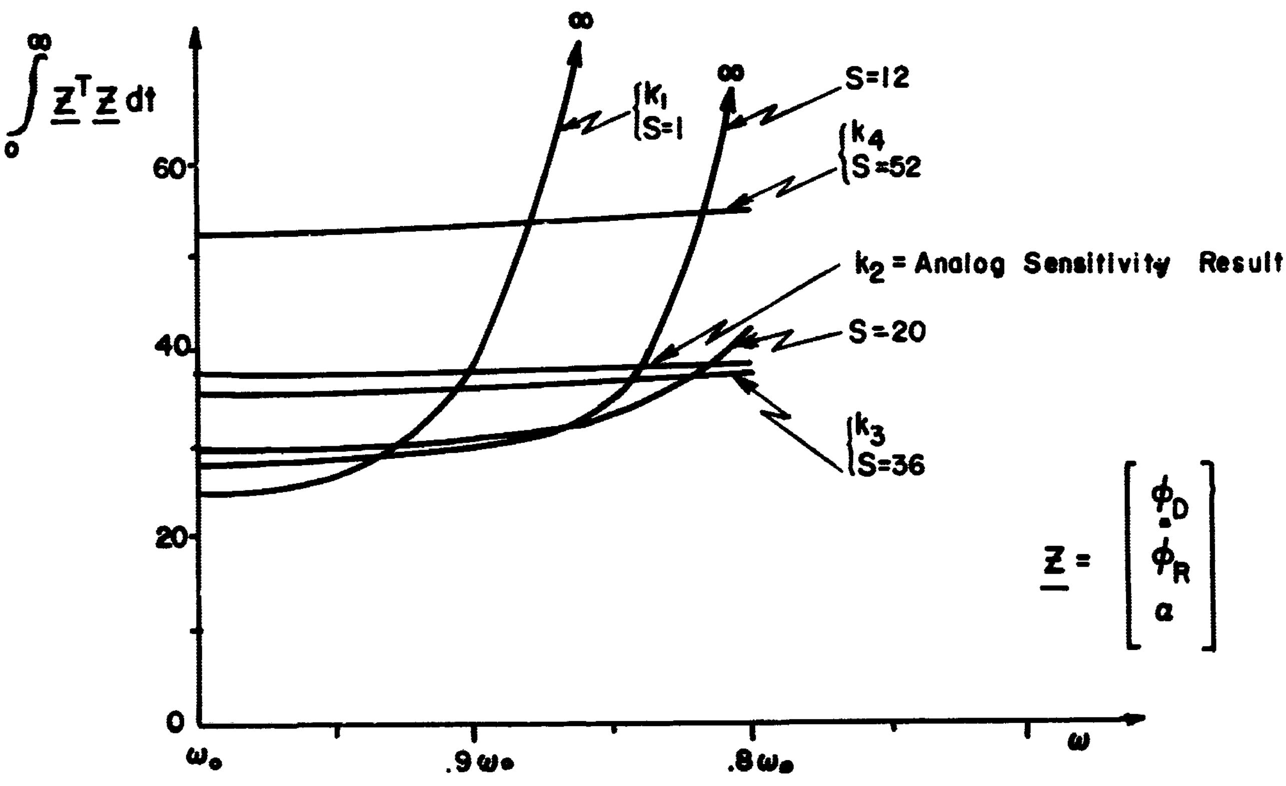


FIGURE 7.22 J_x vs. ω

nominal control (s = 1) becomes unstable as $\omega \rightarrow .8\,\omega_o$. Various SOC sensitivity control laws as well as the ASD design are compared. Note that SOC does not appreciably reduce the root dispersions, rather the nominal pole position is located so that as the bending frequencies are reduced, the closed loop system remains stable.

Based on the fact that the desensitized control laws are obtained by increasing the sensitivity weighting it would appear that a tradeoff between nominal performance modeled by,

$$J_x = \int_0^{\infty} \underline{x}^T \underline{x}\, dt,$$

and sensitivity characterized by changes in J_x is obtained. The conjecture is verified graphically in Fig. 7.21. In this figure values of J_x are plotted versus the bending frequency. As the sensitivity weighting is increased the nominal performance deteriorates slightly while the variations of J_x with respect to ω remain finite and eventually become small.

This deterioration in nominal performance is relatively low as evidenced by the responses of Fig. 7.23, 7.24 and 7.25. That the actual trajectory dispersion is low is verified by Fig. 7.26 in which the ϕ and α state dispersions are plotted. The time varying simulation indicates that for 80% nominal bending frequency and the nominal control law the launch vehicle is unstable. The SOC control laws

$$\underline{k}_1 = \begin{bmatrix} -.821 \\ -.500 \end{bmatrix} \quad \text{and} \quad \underline{k}_2 = \begin{bmatrix} -.921 \\ -.394 \end{bmatrix}$$

are somewhat more desirable than the ASD design

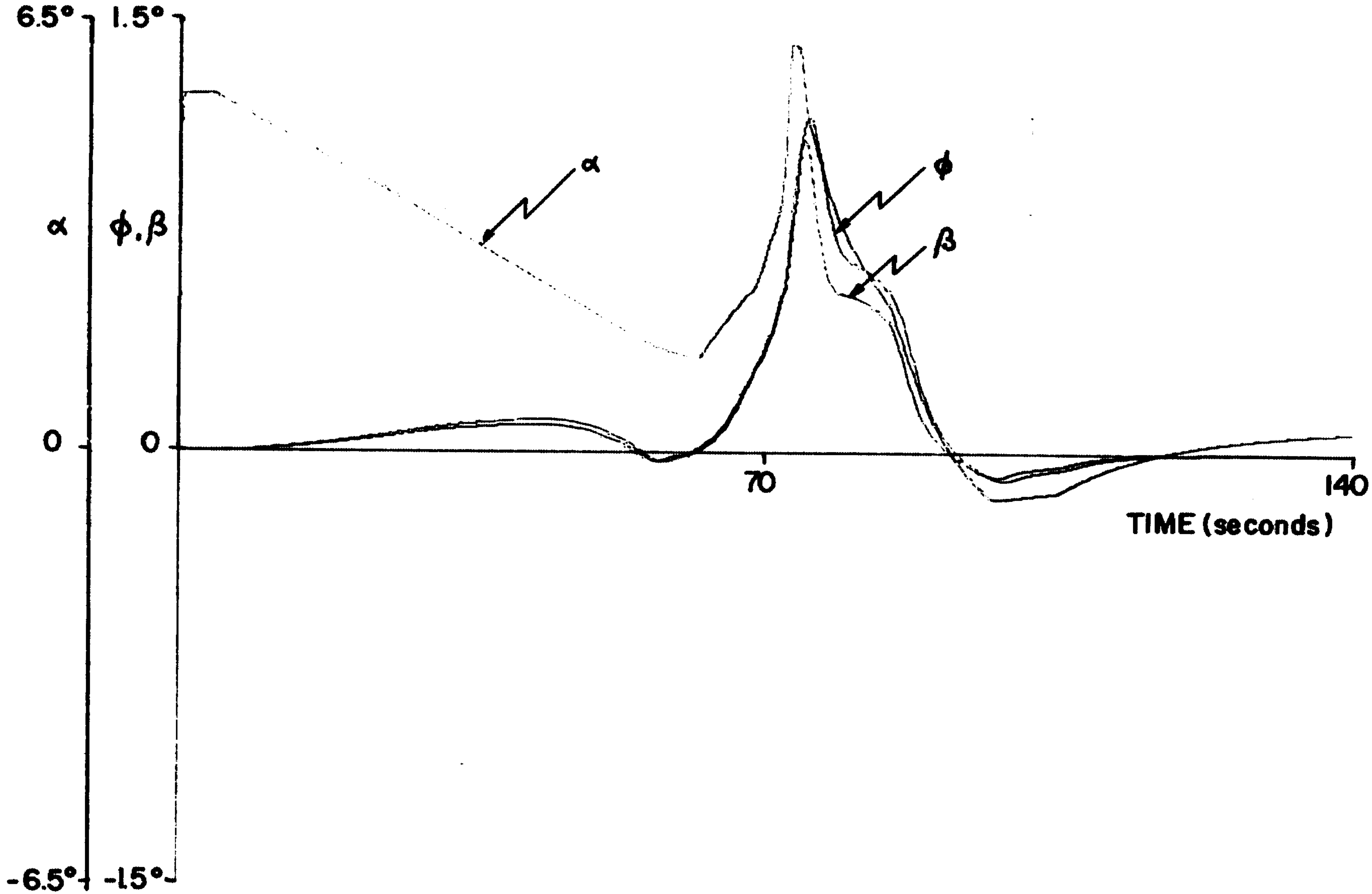


FIGURE 7.23-a NOMINAL CONTROL LAW ω_0 : RIGID BODY

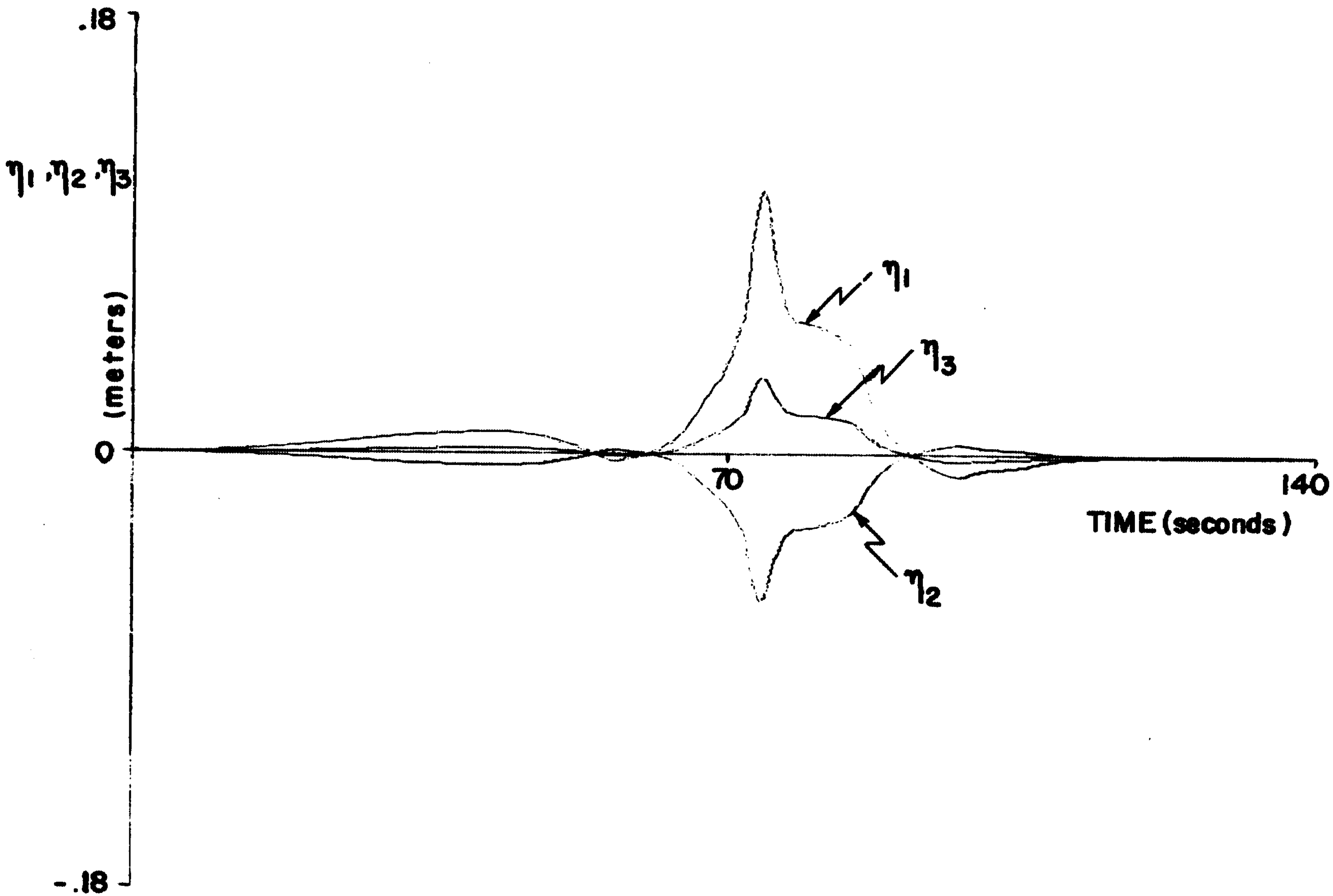


FIGURE 7.23-b NOMINAL CONTROL LAW ω_0 : BENDING

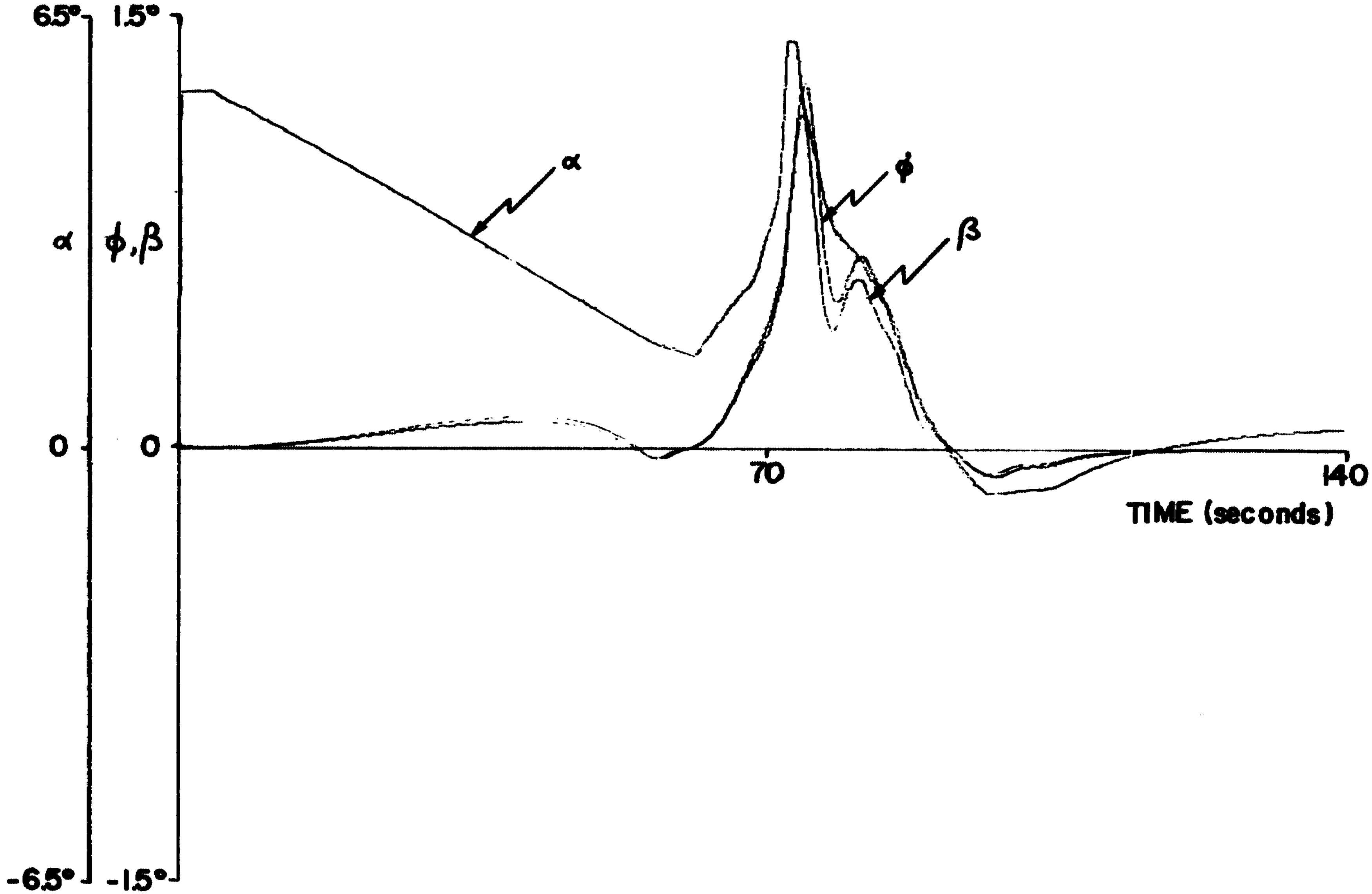


FIGURE 7.24-a SOC SENSITIVITY CONTROL LAW ω_o : RIGID BODY

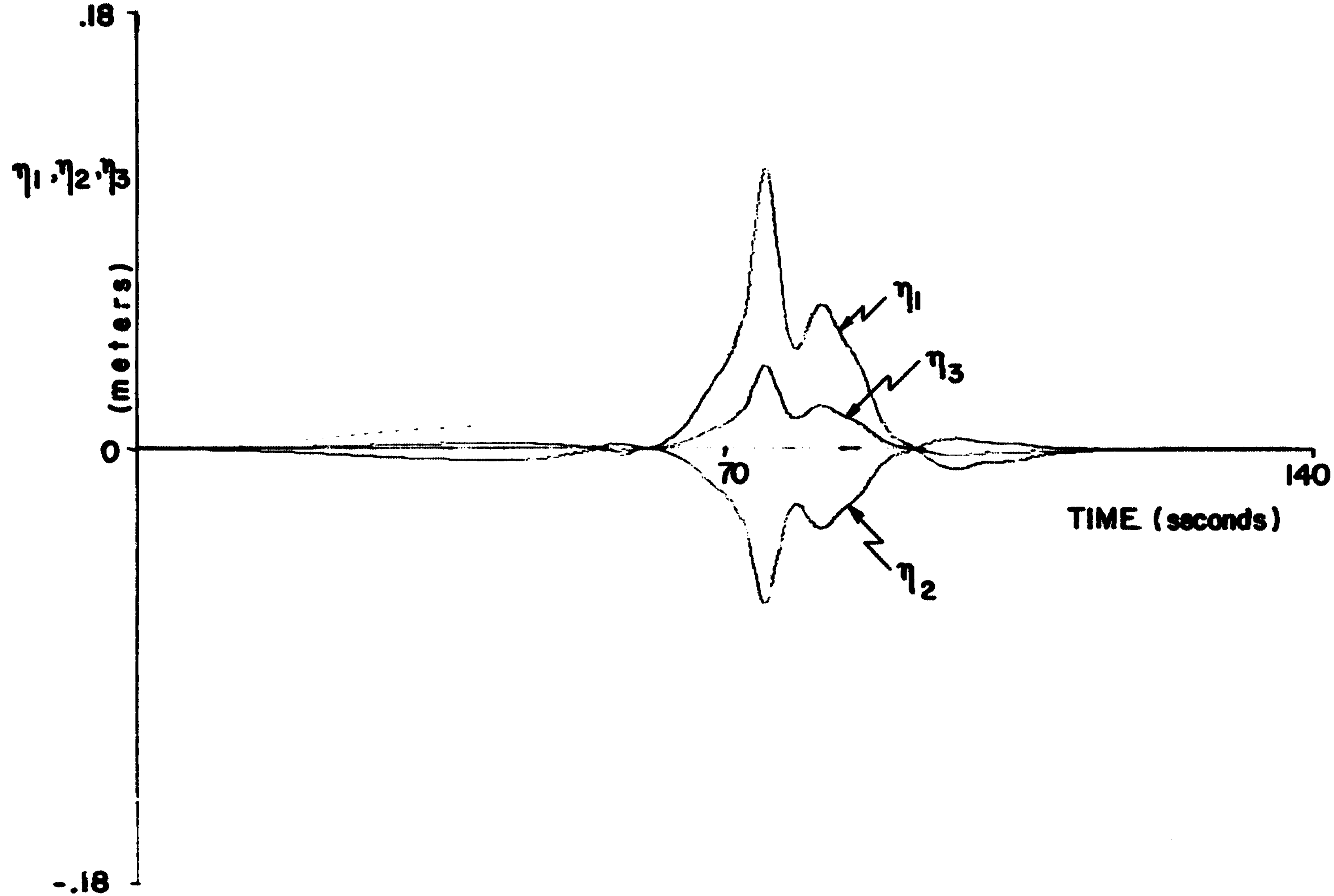


FIGURE 7.24-b SOC SENSITIVITY CONTROL LAW ω_e : BENDING

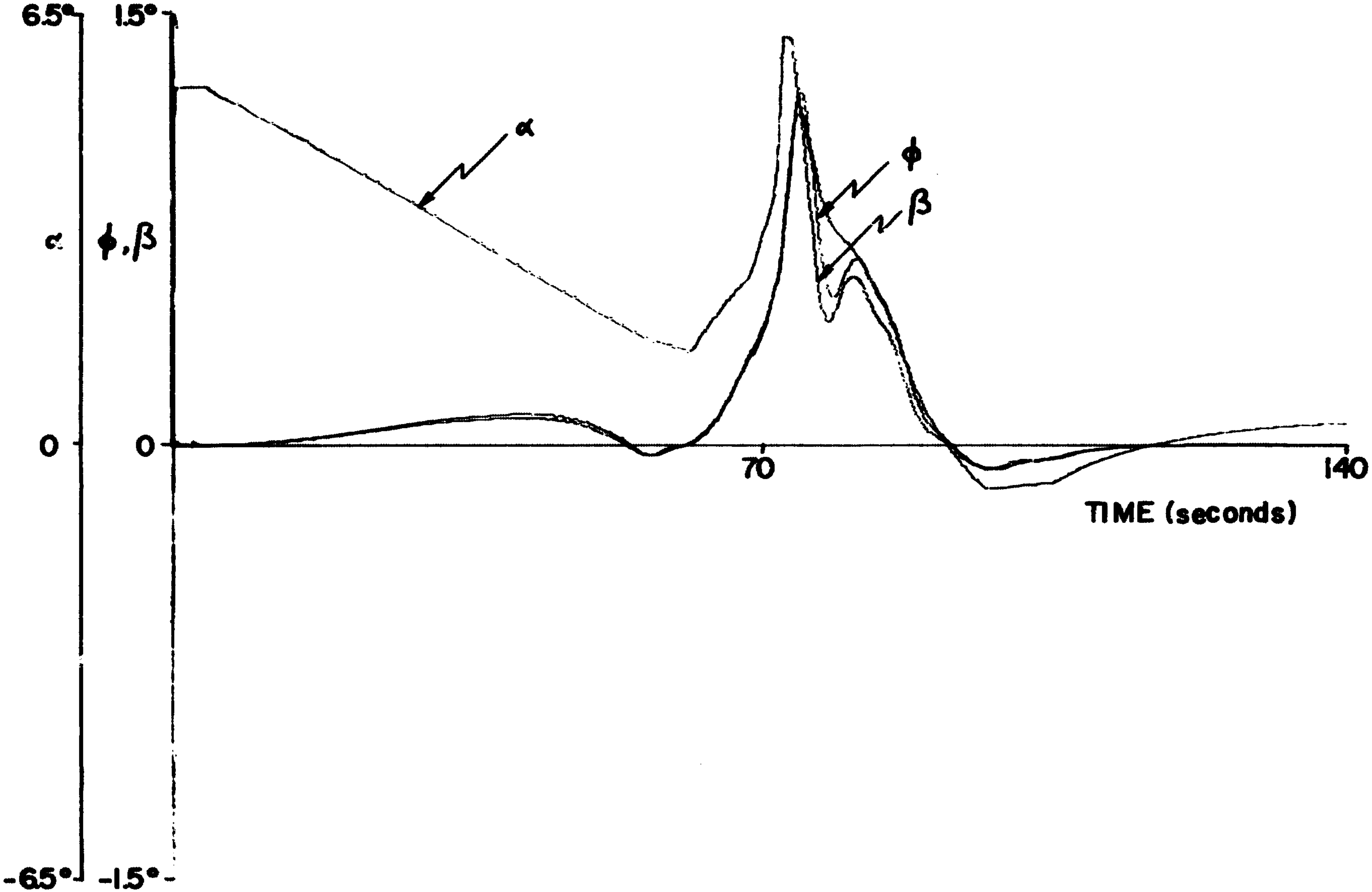


FIGURE 7.25-a SOC SENSITIVITY CONTROL LAW .8 ω_o: RIGID BODY

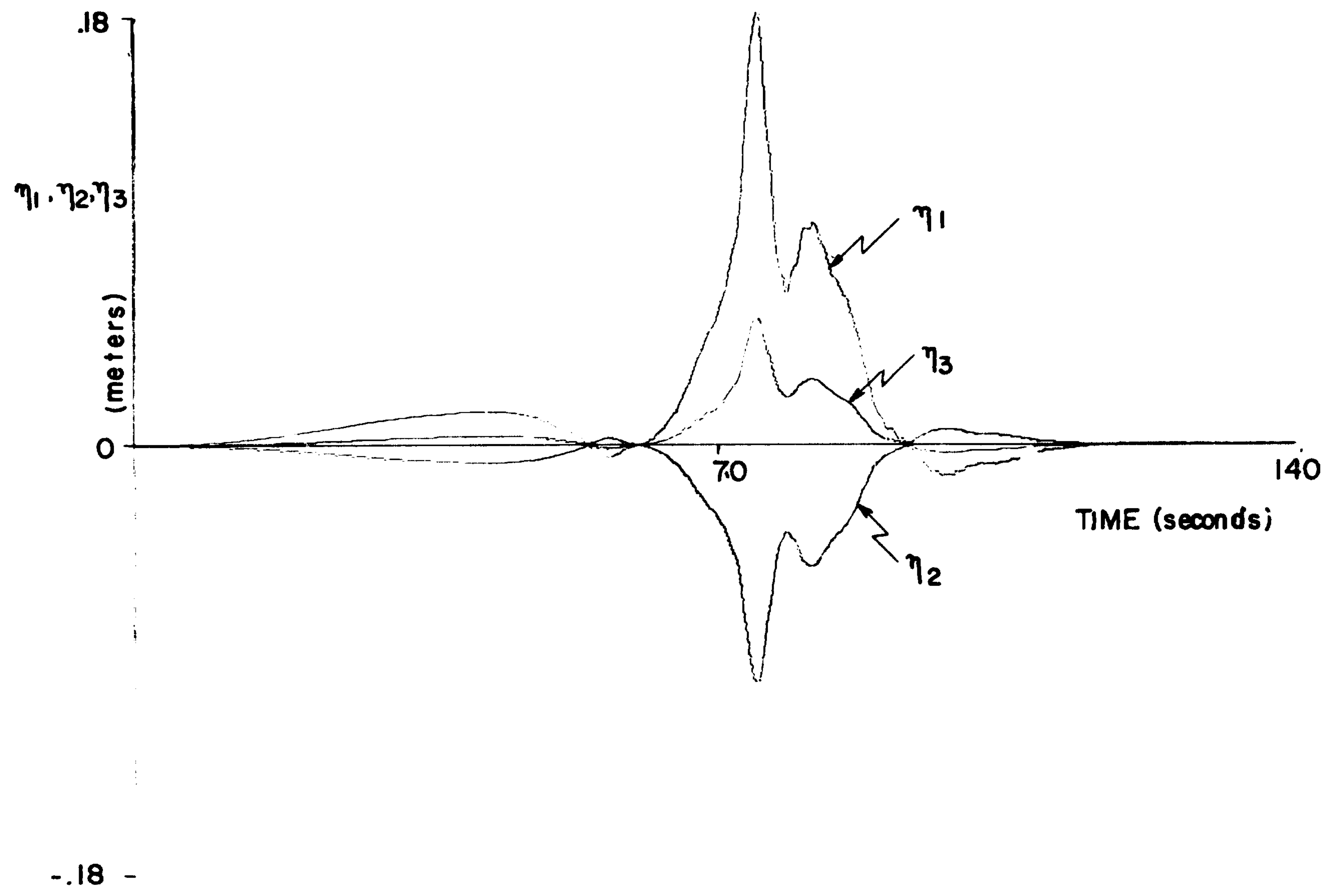


FIGURE 7.25-b SOC SENSITIVITY CONTROL LAW .8 ω_o : BENDING

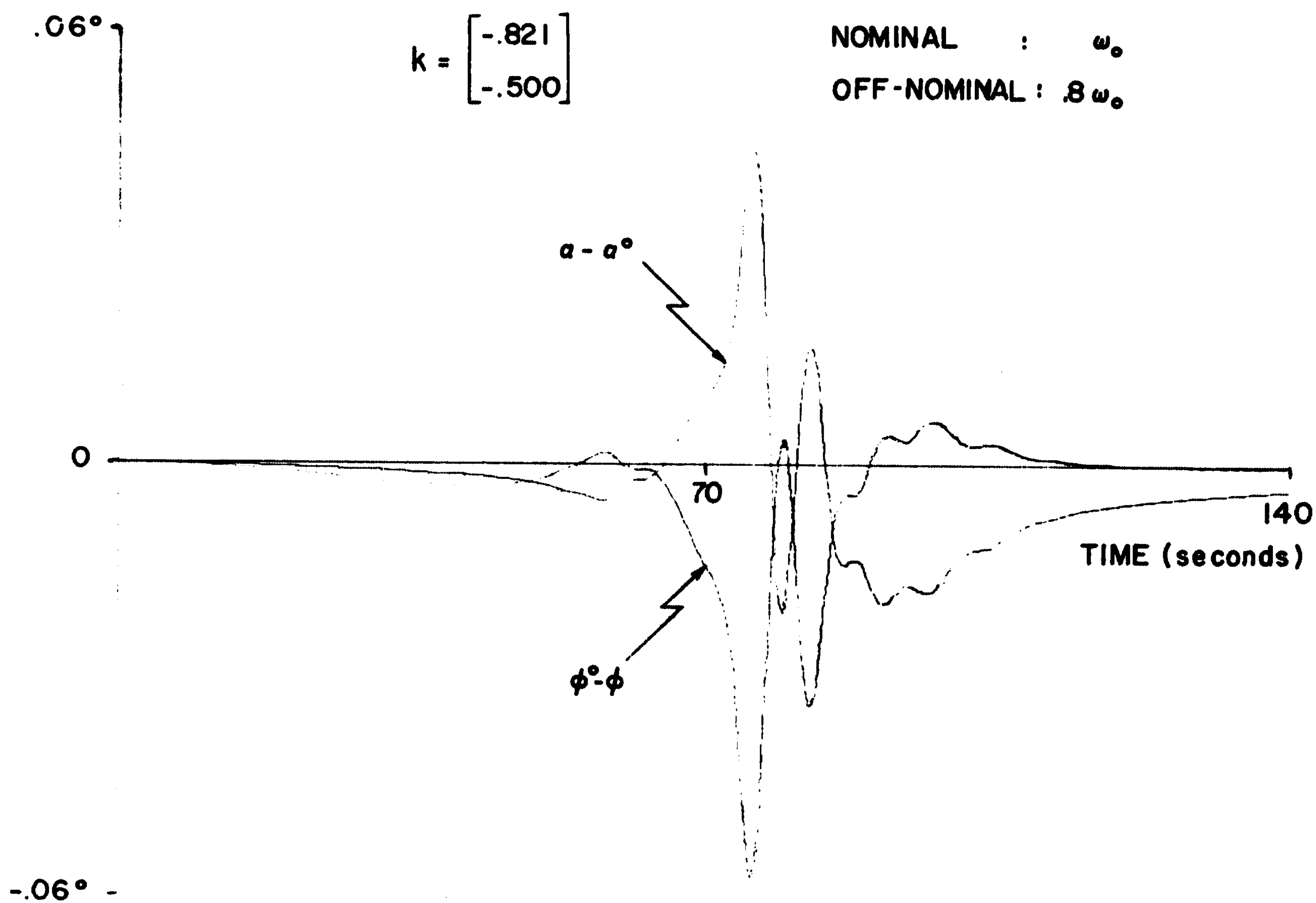


FIGURE 7.26 TRAJECTORY DISPERSIONS

$$\underline{k} = \begin{bmatrix} -.500 \\ -.400 \end{bmatrix}$$

since the peak value of the pitch response is reduced for $\underline{k}_1$ and $\underline{k}_2$. Table 7.3 indicates that for the time varying simulation with design wind these control laws do indeed reduce the sensitivity of the trajectory with very little degradation of performance.

Result:

The SOC sensitivity problem involves a tradeoff between nominal performance and sensitivity. For this launch vehicle problem the tradeoff is mild and leads to a very acceptable desensitized control law.

It should be noted that the designs were made using the seven state fixed time point model with only one bending mode, but were checked by application to the time varying model with three bending modes. Moreover, rapid computation is a feature of the SOC sensitivity method since the entire $\underline{k}$ locus may be calculated in about ten minutes.

7.6.4 Application of SOC Model Reference

Based on the insight obtained from the analysis of Chapter V it was decided not to apply the model reference technique directly but rather combine the model reference and sensitivity approaches. The basic idea is that the deterioration of nominal performance encountered in the sensitivity approach may be eliminated. The inner loop gains are designed to provide the best nominal performance while the model reference gains are found from the difference between the composite and inner loop gains. The composite gains are calculated using the SOC sensitivity approach.

$$\underline{k}_1 = \begin{bmatrix} -.821 \\ -.500 \end{bmatrix} ; \qquad \underline{k}_2 = \begin{bmatrix} -.921 \\ -.394 \end{bmatrix}$$

The performance of the model reference systems with the desensitized composite loop gains is compared with the performance of the nominal and pure sensitivity control systems. In Fig. 7.27 the responses are displayed for $\omega = 0.8\ \omega_o$ with nominal inner loop gains and

$$k_1 = \begin{bmatrix} -.821 \\ -.5 \end{bmatrix}$$

as the composite gains. Compare these curves with the sensitivity results of Fig. 7.24. By definition the nominal responses of this model reference system will be identical to the nominal response of Fig. 7.23. As the bending frequency is decreased the rigid body performance improves slightly while the bending performance deteriorates. As shown in Table 7.3 the model reference scheme is slightly more effective in reducing the trajectory dispersions but these slight improvements do not justify the implementation of the more complex model reference control system.

7.6.5 Conclusions

As a result of the application of the SOC techniques, four general points can be made.

1. Even for a practical problem, such as the booster, the computational effort required by SOC is small. One iteration of the SOCDES algorithm required four seconds; the solution of a typical SOC problem required five iterations (20 seconds).

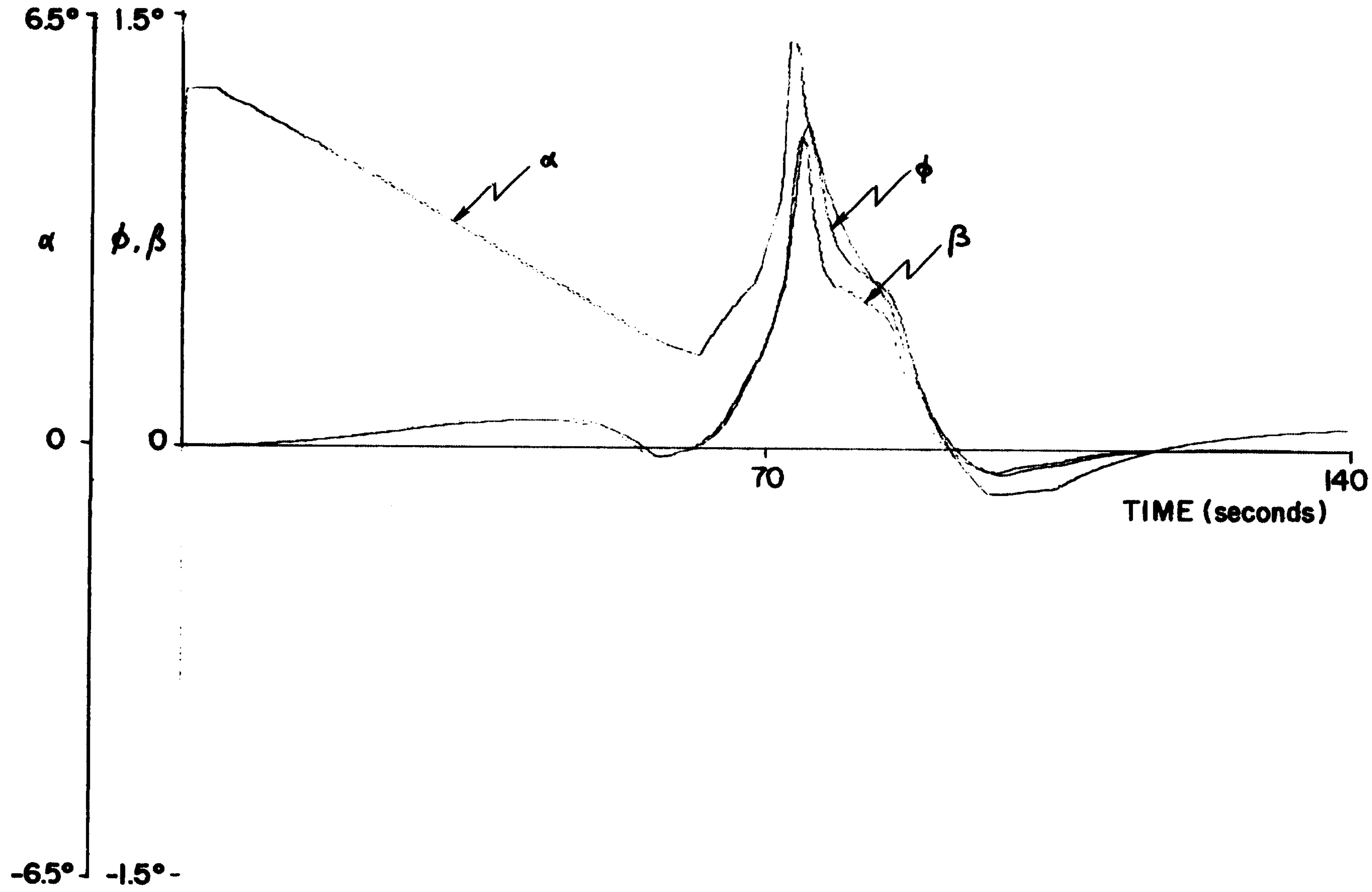


FIGURE 7.27-a MODEL REFERENCE SYSTEM RESPONSE .8 ω_o
RIGID BODY

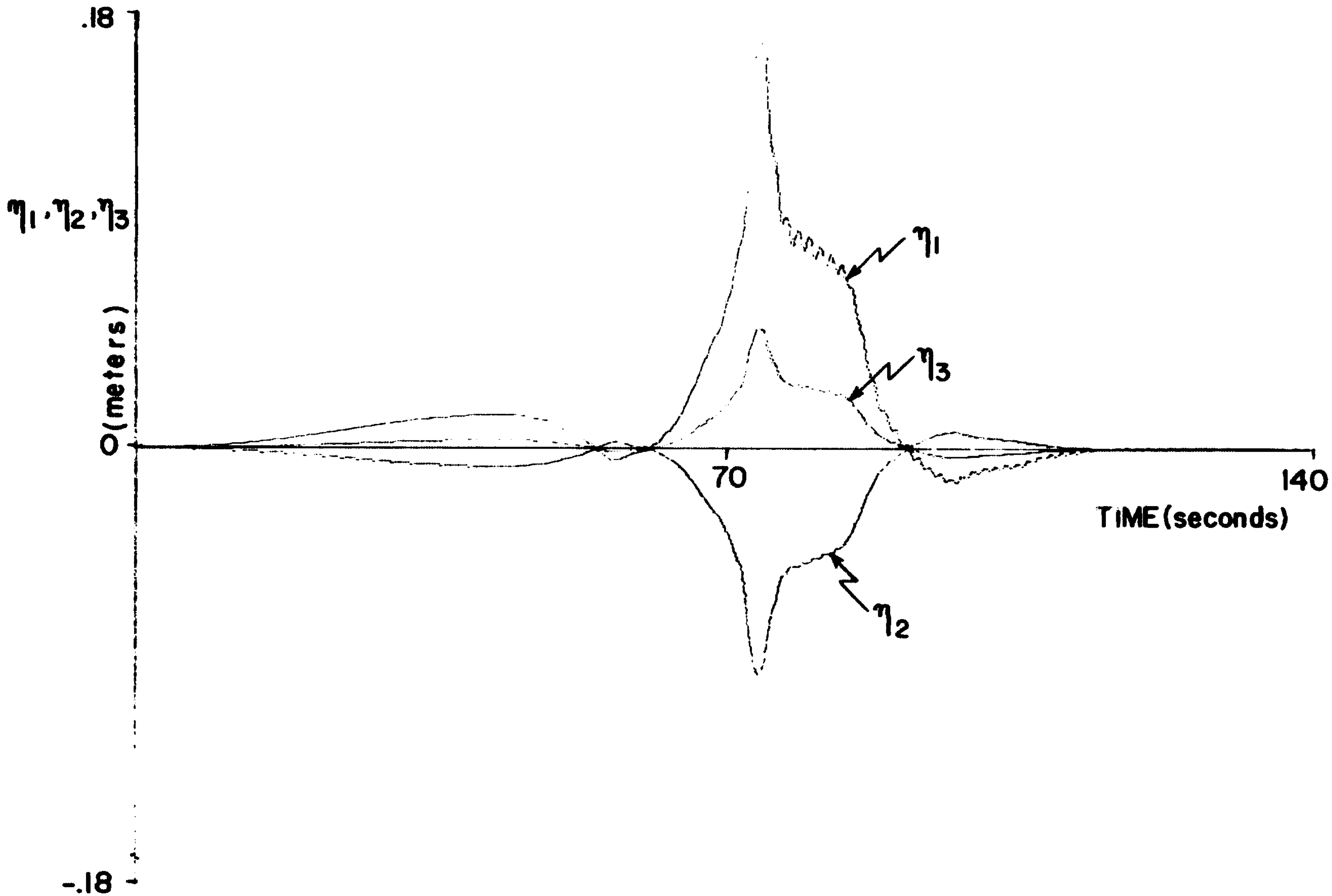


FIGURE 7.27-b MODEL REFERENCE SYSTEM RESPONSE $.8\omega_0$: BENDING

$k_1 = \begin{bmatrix} -.821 \\ -.500 \end{bmatrix}$ $k_3 = \begin{bmatrix} -.8 \\ -.8 \end{bmatrix}$

$k_2 = \begin{bmatrix} -.921 \\ -.394 \end{bmatrix}$ $k_4 = \begin{bmatrix} -.5 \\ -.4 \end{bmatrix}$

MR = MODEL REFERENCE ; S = SOC SENSITIVITY
ASD = ANALOG SENSITIVITY DESIGN ; NOM = NOMINAL

	max $\lvert\phi\rvert$	max $\lvert a\rvert$	max $\lvert\eta_1\rvert$	$\int_0^{140} \phi^2 dt$	$\int_0^{140} a^2 dt$	$\int_0^{140} \eta_1^2 dt$
k_1, MR , $\omega = \omega_o$	1.16	6.11	.10	197	2.22 10^4	1.41
k_1, MR , $\omega = .8\ \omega_o$	1.11	6.09	.17	197	2.22 10^4	3.43
k_2, MR , $\omega = \omega_o$	1.16	6.11	.10	197	2.22 10^4	1.41
k_2, MR , $\omega = .8\ \omega_o$	1.11	6.09	.16	197	2.22 10^4	3.43
k_1, S , $\omega = \omega_o$	1.26	6.13	.11	164	2.22 10^4	1.44
k_1, S , $\omega = .8\omega_o$	1.21	6.12	.18	164	2.22 10^4	3.48
k_2, S , $\omega = \omega_o$	1.18	6.10	.12	164	2.21 10^4	1.44
k_2, S , $\omega = .8\omega_o$	1.13	6.08	.18	131	2.21 10^4	3.50
k_3, NOM, $\omega = \omega_o$	1.16	6.11	.10	197	2.22 10^4	1.41
k_4, ASD, $\omega = \omega_o$	2.23	6.53	.12	624	2.30 10^4	1.72
k_4, ASD, $\omega = .8\ \omega_o$	2.17	6.50	.18	624	2.30 10^4	4.16

TABLE 7.3 SENSITIVITY and MODEL REFERENCE SIMULATIONS

2. The SOC procedures are very easy to use. Through the use of the reverse SOC problem concept, very little effort is required to initiate the computational procedure. The reverse problem generates an initial set of weightings which correspond to equations which are numerically well behaved.

3. The use of the SOC approach to calculate a number of designs and the interpretation of the results by using the graphic aids affords an insight into and generates explicit information about complex problems.

4. By varying the relationship between the weightings and calculating the various loci, it is possible to probe all the areas of the stable gain space and thus determine the properties of the system being studied.

The solutions generated by the SOC approach are comparable to those obtained from other methods with respect to the satisfaction of design specifications. It appears that for this particular problem the SOC sensitivity control law is to be preferred over that of the SOC model reference. The marginal improvement in performance does not warrant the additional complexity of the implementation of the model reference scheme.

Result:

Based on the preceding analysis the following SOC sensitivity feedback control law is proposed.

$$\underline{k} = \begin{bmatrix} -.821 \\ -.500 \end{bmatrix}$$

With this control law the following limits are maintained for the duration of the time varying simulation with 95% design wind for any value of bending frequency between nominal and 80% of nominal.

$$|\phi| < 1.3^\circ$$

$$|\dot{\phi}| < .35^\circ/\text{second}$$

$$|\alpha| < 6.15^\circ$$

$$|\beta| < 1.3^\circ$$

$$|\eta_1| < .18 \text{ meters}$$

$$|\eta_2| < .10 \text{ meters}$$

$$|\eta_3| < .06 \text{ meters}$$

Two areas of this work that should be pursued are the further development of the digital computer programs of the SOC procedures and the investigation of additional numerical methods. For example, using a recently proposed[51] algorithm, it appears that a sensitivity gain root locus such as that pictured in Fig. 7.18 could be generated for a 20th order system with about forty-five minutes of 360/50 computation.

Nomenclature

Matrices

A	Booster system matrix: 7 by 7
C	Booster observation matrix: 7 by 7
S	Symmetric state weighting matrix: 7 by 7
S_1	Symmetric state weighting matrix: 7 by 7
S_3	Symmetric sensitivity state weighting matrix: 7 by 7

Vectors

$\underline{a}$	Acceleration of vehicle
$\underline{b}$	Booster control coefficient vector: 7 elements
D	Drag force
F	Centerline thrust
g	Gravity
$\underline{i}$	Unit vector X_n direction
$\underline{j}$	Unit vector Y_n direction
$\underline{k}$	Unit vector perpendicular to X_n Y_n plane
$\underline{k}$	Feedback gain vector
N	Normal aerodynamic force
R'	Gimballed thrust
$\underline{v}$	Additive disturbance vector
$\underline{v}$	Velocity of vehicle
$\underline{v}_w$	Wind velocity
$\underline{v}_R$	Velocity of vehicle relative to wind
$\underline{x}$	State vector: 7 elements

Scalars

J_x	Cost index
k_1, k_2	Feedback gains
l_{cg}	Thrust moment arm
l_{cp}	Normal force moment arm
m	Mass of vehicle
M_i	Equivalent engine mass
N'	Normal force coefficient
Q	Control weighting
X_n	Nominal frame co-ordinate
X	Inertial frame co-ordinate
x	Airframe co-ordinate
Y_n	Nominal frame co-ordinate
Y	Inertial frame co-ordinate
y	Airframe co-ordinate
$y_i(x)$	Slope of i^{th} mode at point x
α	Angle-of-attack
α_w	Wind induced angle-of-attack
β	Gimbal angle
β_c	Control signal
η_i	Normalized bending
υ	Angle between velocity vector and X_n co-ordinate
ϕ	Pitch angle
ϕ_D	Output of pitch gyro
$\dot{\phi}$	Pitch rate
$\dot{\phi}_R$	Output of pitch rate gyro
χ_c	Nominal trajectory angle
ζ_i	Bending damping coefficient of i^{th} mode
ω_i	Bending frequency of i^{th} mode

Chapter 8

SUMMARY AND CONCLUSIONS

8.1 Contributions of This Work

The underlying theme of this work has been the Specific Optimal Control concept. This approach allows the advantages of the modern and classical techniques of control theory to be combined by formulating optimal control problems in which the primary goal is a solution control law with certain specified properties. This control law is obtained by the minimization of a cost index which has been structured to insure that the optimal solution will possess these properties.

This concept was applied to the problem of calculating control laws for systems in which not all of the states are available, the unavailable state problem. The important feature of the linear SOC problem and its solutions are listed below.

1. The linear SOC problems are a class of linear optimal control problems in which some of the weighting matrices are chosen to provide a specified structure while others are chosen to obtain satisfactory system response.

2. The basic control structure is linear feedback and the gains are independent of system initial conditions.

3. The SOC approach has unavailable state capabilities since those feedback gains corresponding to unavailable states may be structured to be zero.

4. The steady state SOC control laws for the time invariant problem are asymptotically stable.

5. The linear SOC problem has desirable computational properties. a) The optimal solution for the time invariant steady state problem is characterized by systems of nonlinear algebraic equations. b) The simple structure of these equations is independent of size or complexity of the system. c) Efficient numerical methods are available for the solution of the SOC necessary condition equations.

The linear SOC problem is justified from a mathematical point of view by the study of the existence and uniqueness of the solutions to the SOC necessary condition equations. It was shown that for any system which can be controlled with a control law of the specified structure, there are classes of weighting matrices for which solutions to the SOC problem exist and are unique. One class of these weightings may be determined by the solution of the Reverse SOC problem. That is, given any control law for which the system response is square integrable, the corresponding SOC problem with this control law as the optimal solution can be found. Using this Reverse SOC problem as a starting point, it is possible to vary the weightings and redesign the system response.

In addition, the concept of the Reverse problem may have application to a wide range of optimal control problems. One of the main difficulties concerned with the optimal approach is of a computational nature. It is often very difficult to determine the proper computational parameters or initial guesses which result in a well behaved numerical solution.

For example, a unique solution to the ordinary allstate linear quadratic problem exists for any positive definite state and control weightings. However for most problems, many choices of these weightings result in necessary condition equations which are numerically difficult to solve. The Reverse problem generates a set of well behaved equations which have the known control law as a solution. The equations corresponding to new problems obtained by perturbing the weightings, are usually well behaved. Thus the effort and skill needed to use the method is reduced since numerically well behaved problems are automatically formulated. This technique is especially effective when the optimal procedure is being used to improve or modify an existing control law.

Most of the optimal control approaches are computationally bound since a large amount of computational effort is required to solve even simple problems. An important feature of SOC is the relatively low computational effort requirement. This feature is due to the basic structure of the equations defining the optimal solutions and to the new computational procedure, the SOCDES algorithm, which has been introduced in this work. This algorithm solves the algebraic matrix Ricatti equation which characterizes the steady state optimal solution. The control concept of SOCDES is the indirect solution of the Ricatti equation; the feedback gain equation is solved by Newton-Raphson iteration while the Ricatti equation acts as a constraint relating the Ricatti Matrix and the feedback gains. Although the execution time per iteration is longer than that of the straight forward Newton Raphson solution of the Ricatti equation, the rate of convergence of SOCDES measured in number of iterationsis faster. The superiority of the SOCDES algorithm becomes apparent in most practical problems in which there are many states with only a few control variables.

The Reverse SOC problem and these computational features have been combined to form a systematic procedure for the analysis and synthesis of linear feedback control systems. The synthesis is carried out by a systematic trial and error procedure in which the Reverse problem is solved to obtain an initial set of weightings and the weightings are perturbed to obtain a more satisfactory design.

Analysis of and insight into a linear system is obtained by allowing the SOCDES algorithm to calculate the solution for a number of weighting matrices and interpreting the results in terms of the following graphical aids, the feedback gain root locus which is a plot of the feedback gains as a function of the weighting matrices and the weighting root locus which is a plot of the poles (characteristic roots) of the closed loop system as a function of the weightings.

The SOC concept was applied to the model reference control problem in which a control law is designed to maintain the trajectory of a system in the neighborhood of the nominal or model reference trajectory despite enviromental disturbances. The result of the SOC application is a model reference control system with two loops, an inner loop designed to obtain a nominal response and an outer loop designed with SOC which operates on the difference between the actual and model trajectories. An important feature of this technique is that the outer loop gains are independent of the nominal trajectory as well as system initial conditions. After these feedback gains are chosen, the model reference trajectories may be changed or modified without any redesign of the feedback gains.

Another approach to the problem of the effect of enviromental changes on the controlled system is the use of sensitivity considerations. Previous efforts employing the optimal control approach to sensitivity have not been effective for realistic problems because of difficulties encountered in formulation and computation. The SOC sensitivity technique introduced in this work substantially reduces these difficulties. In addition to the computational reduction resulting from the nature of SOC, the sensitivity problem has been formulated so that the computational effort required is about the same as for the unavailable state SOC problem without sensitivity considerations. Moreover, this effort is relatively independent of the number of parameters considered. Furthermore, this technique has the unavailable state capability so that the unavailable states do not have to be measured or estimated nor do the sensitivity variables have to be generated.

The efficacy of the SOC theory and the techniques described above was demonstrated by simple examples and the study of a significant engineering problem, the control of the Saturn V launch vehicle. As indicated in Chapter VII, which describes the launch vehicle problem in detail, the SOC approach may be very useful with respect to the study of practical problems. The actual designs are comparable to other techniques with respect to satisfying the design specifications with the advantages of reduced computational effort and increased insight. The SOC sensitivity approach appears to be especially effective.

8.2 Future Work

In this work the SOC concept was applied to linear systems with emphasis on the time invariant case. Most of these ideas expressed in the previous chapters are directly applicable to the time varying case. This particular application of the SOC concept depends on the structure of the equations defining the optimal solution for the linear quadratic problem. A similar approach may be used to apply the SOC concept to any linear problem which employs the integral quadratic cost index. Thus, extensions to the discrete and stochastic problems are possible. Similarly, nonlinear problems may be attacked using the second variational or neighboring optimal control problem approaches. Some work has already been done in these areas with encouraging results.

The further development of the SOC procedure as an automated design technique appears to be feasible. There are indications that the use of SOC to choose an "optimal" compensator as well as the generation of an initial set of stable gains are promising areas of future investigation.

The digital computer programs currently available were written in a straight forward "brute force" manner to test the SOC techniques. No significant effort was made to optimize the execution times, memory requirements or the handling of input and output. Additional work along these lines might lead to sets of programs that would comprise a useful design tool suitable for time share library usage.

An effort should be made to investigate the relationships between the SOC techniques and other optimal and classical approaches. This work might involve a theoretical comparison as well as an empirical comparison involving the solution of a number of problems with the various methods.

LITERATURE CITED

1. Wiener, N. Extrapolation, Interpolation, and Smoothing of Stationary Time Series. Technology Press, Cambridge, Mass., 1949.

2. Newton, G., L. Gould and J. Kaiser. Analytic Design of Linear Feedback Controllers. Wiley and Sons, New York, 1957.

3. DeRusso, P., R. Roy and C. Close. State Variables for Engineers. Wiley and Sons, New York, 1965.

4. Schultz, D. and J. Melsa. State Functions and Linear Control Systems. McGraw Hill, New York, 1967.

5. Kalman, R. and R. Bucy. "New Results in Linear Filtering and Prediction Theory." Transactions ASME, Journal of Basic Engineering, Series D, Vol. 83, pp. 95-108, March 1961.

6. Luenburger, D. "Observing the State of a Linear System." IEEE Trans. on Military Electronics, Vol. MIL-8, No. 2, pp. 74-80, April 1964.

7. Merriam, C. W. III. Optimization Theory and the Design of Feedback Control Systems. McGraw Hill, New York, 1964.

8. Hass, V. and S. Murtuza. "On the Design of Specific Optimal Controllers." Purdue University School of Electrical Engineering, May 1967.

9. Agarwal, G. and R. Sridhar. "Design of Specific Optimal Control Systems." IEEE Region Six Conference, Tucson, Arizonia, 1966.

10. Rekasius, Z. "Optimal Linear Regulators with Incomplete State Feedback." IEEE Trans. on Automatic Control, Vol. AC-12, pp. 296-299, June 1967.

11. Kalman, R. "Contributions to the Theory of Optimal Control." Bol. Soc. Mat. Mexicana, pp. 102-119, 1960.

12. Kalman, R. "When is a Linear System Optimal?" RIAS Technical Rept. 63-5, March 1963.

13. Athans, M. and P. Falb. Optimal Control. McGraw Hill, New York, 1966.

14. Zadeh, L. and C. Desoer. Linear System Theory. McGraw Hill, New York, 1963.

15. Ayres, F. Matrices. Schaum Outline Series, New York,1962

16. Rynaski, E., P. Reynolds and W. Shed. "Design of Linear Flight Control Systems Using Optimal Control Theory." Air Force Flight Dynamics Laboratory, Wright Patterson A.F.B., ASD-TDR-63-376, 1964.

17. Osborn, P. and H. Whitaker and A. Kezer. "New Developments in the Design of Model-Reference Adaptive Control Systems." Paper No. 61-39, IAS 29th Annual Meeting, January 1961.

18. Donalson, D. and C. Leondes. "A Model Referenced Parameter Tracking Technique for Adaptive Control Systems." IEEE Trans. on Applications and Industry, Vol. 82, pp. 241-262, September 1963.

19. Dressler, R. "An Approach to Model-Referenced Adaptive Control Systems." IEEE Trans. on Automatic Control, Vol. AC-12, pp. 75-80, February 1967.

20. Parks, P. "Lyapunov Redesign of Model Reference Adaptive Control Schemes." IEEE Trans. on Automatic Control, Vol. AC-11, pp. 363-367, July 1966.

21. Shackcloth, B. "Design of Model Reference Control Systems Using a Lyapunov Synthesis Technique." Proc. IEEE, Vol. 114, pp. 299-307, February 1967.

22. Tyler, J. "The Characteristics of Model-Following Systems as Synthesized by Optimal Control." IEEE Trans. on Automatic Control, Vol. AC-9, pp. 485-498, October 1964.

23. Asseo, S. "Application of Optimal Control to 'Perfect' Model Following." 1968 JACC Preprints.

24. Bode, H. Network Analysis and Feedback Amplifier Design. D. Van Nostrand, New York, 1945.

25. Horowitz, I. Synthesis of Feedback Systems. Academic Press, New York 1963.

26. Horowitz, I. "Plant Adaptive vs. Ordinary Feedback Control Systems." IRE Trans. on Automatic Control, pp. 48-57, January 1962.

27. Kuo, F. "Pole-Zero Sensitivity in Network Functions." IRE Trans. on Circuit Theory, December 1958.

28. Huang, J. "The Sensitivity of the Poles of Linear, Closed Loop Systems." AIEE Trans. Part II, Vol. 77, 1958.

29. Miller, K. and F. Murray. "A Mathematical Basis for an Error Analysis of Differential Analyzers." Journal of Math. and Physics, July 1953.

30. Dorato, P. "On Sensitivity in Optimal Control Problems." IEEE Trans. on Automatic Control, pp. 256-257, July 1963.

31. Rohrer, R. and M. Sobral. "Sensitivity Considerations in Optimal Systems Design." IEEE Trans. on Automatic Control, pp. 43-49, January 1965.

32. Pagurek, B. "Sensitivity of the Performance of Optimal Control Systems to Plant Parameter Variations." IEEE Trans. on Automatic Control, pp. 178-180, April 1965.

33. Holtzman, J. and S. Horing. "The Sensitivity of Terminal Conditions of Optimal Control Systems to Parameter Variations." IEEE Trans. on Automatic Control, pp. 420-427, October 1965.

34. Tomovic, R. Sensitivity Analysis of Dynamic Systems. McGraw Hill, New York, 1955.

35. Tuel, W., P. DeRusso, and I. Lee. "Synthesis of Optimal Control Systems with Sensitivity Considerations." 3rd IFAC Conference, London, June 1966.

36. Dougherty, H., I. Lee and P. DeRusso. "Synthesis of Optimal Feedback Control Systems subject to Parameter Variations." 1967 JACC Preprints.

37. Kalman, R. and J. Bertram. "Control System Analysis and Design Via the "Second Method" of Lyapunov." Trans. ASME, Journal of Basic Engineering, June 1960.

38. Coddington, E. and N. Levinson. Theory of Ordinary Differential Equations. McGraw Hill, New York, 1955.

39. Cooper, R. Functions of Real Variables. D. Van Nostrand, London, 1966.

40. MacFarlane, A. "An Eigenvector Solution of the Optimal Linear Regulator Problem." Journal of Electronics and Control, Vol. 14, pp. 643-654, June 1963.

41. Bass, R. "Machine Solution of High Order Matrix Ricatti Solutions." Douglass Paper No. 4538, 1967.

42. Blackburn, T. "Solution of the Algebraic Matrix Ricatti Equation via Newton Raphson Iteration." 1968 JACC Preprints.

43. Kleinman, D. "On an Iterative Technique for Ricatti Equation Computations." IEEE Trans. on Automatic Control, Vol. AC-13, pp. 114-115, February 1968.

44. Gantmakher, F. The Theory of Matrices. Chelsea, 1959.

45. Rillings, J. "Analog Sensitivity Design." Final Report, Vol. II, Contract No. NAS8-21131, NASA, 1968.

46. Chen, C. and L. Shiek. "A Note on Expanding $PA + A^TP = -Q$." IEEE Trans. on Automatic Control, Vol. AC-13, pp. 122-124, February 1968.

47. Gelfond, I. and Fomin, S. Calculus of Variations. Prentice Hall, New Jersey, 1965.

48. Data Pack for Saturn V - Appollo. NASA, Astrionics Laboratory, Marshall Spaceflight Center. Huntsville, Alabama, 1967.

49. Garner, D. "Control Theory Handbook." NASA TM X-53036, April 1964.

50. Smith, E. "Analysis of Saturn Booster Performance by Conventional Control Theory Techniques." Final Report Vol. I, Contract No. NAS8-21131, NASA, 1968.

51. Davison, E. and Man, F. "The Numerical Solution of $A^TQ + QA = -C$." IEEE Trans. on Automatic Control, Vol. AC-13, pp. 448-449, August 1968.

Appendix A

Derivation of the Formal SOC Necessary Conditions

This appendix is devoted to the derivation of the unreduced SOC necessary conditions by the application of the calculus of variations.[47] The SOC control law, $\underline{u}$, is chosen to minimize J

$$J = \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^T S \underline{x} + \underline{x}^T \hat{S} \underline{x} + \underline{x}^T W \underline{u} + \underline{x}^T \hat{W} \underline{u} + \underline{u}^T Q \underline{u}) \, dt \tag{A-1}$$

subject to the plant dynamics.

$$\dot{\underline{x}} = A \underline{x} + B \underline{u}; \qquad \underline{x}(t_o) = \underline{c} \tag{A-2}$$

Assume that the optimal control law is known.

$$\dot{\underline{x}}^o = A \underline{x}^o + B \underline{u}^o; \qquad \underline{x}^o(t_o) = \underline{c} \tag{A-3}$$

The necessary conditions will be determined by the consideration of a variation in the control, $\delta \underline{u}$. That is

$$\underline{u} = \underline{u}^o + \delta \underline{u}$$

The resulting system trajectory must satisfy the dynamical contraints in order to be admissable.

$$\underline{x} = \underline{x}^o + \delta \underline{x}$$

$$\dot{\underline{x}} = A \underline{x} + B \underline{u}; \qquad \underline{x}(t_o) = \underline{c} \tag{A-4}$$

By subtracting (A-1) from (A-2) a differential equation is obtained which characterizes any allowable variation about the optimal.

$$\delta \dot{\underline{x}} = \dot{\underline{x}} - \dot{\underline{x}}_o = A \, \delta \underline{x} + B \, \delta \underline{u}; \qquad \delta \underline{x}(t_o) = \underline{0} \tag{A-5}$$

For suitably small and admissable variations in the control and trajectory, the cost index may be expressed in terms of the optimal index and first and higher order variations.

$$J = J^o + \delta J + 0^2$$

where 0^2 represents second and higher order variations of the cost index and

$$\delta J = 2 \int_{t_o}^{t_f} \delta \underline{x}(S + \hat{S})\, \underline{x}^o + \tfrac{1}{2}\, \delta \underline{x}(W + \hat{W})\, \underline{u}^o$$

$$+ \tfrac{1}{2}\, \underline{x}^{o^T}(W + \hat{W})\, \delta \underline{u} + \delta \underline{u}\, Q\, \delta \underline{u})\, dt \qquad \text{(A-6)}$$

The calculus of variations requires that the first variation of the index be zero for any suitably small admissable variations about the optimal. This corresponds to the requirement of the first derivative being zero at an extremum of an ordinary calculus problem. The Euler-Lagrange equations may be derived by adjoining the variational dynamics to the first variation by use of the costate or Lagrange multiplier vector, $\underline{p}$.

$$\delta I = \delta J + 2 \int_{t_o}^{t_f} \underline{p}^T(A\, \delta \underline{x} + B\, \delta \underline{u} - \delta \dot{\underline{x}})\, dt = 0 \qquad \text{(A-7)}$$

Note that δI is zero for all admissable variations because the dynamics are satisfied. Integration by parts of Eq. (A-7) leads to the following expression for δI.

$$I = 2 \int_{t_o}^{t_f} [\, \delta\, \underline{x}^T((S + \hat{S})\, \underline{x}^o + \tfrac{1}{2}\, (W + \hat{W})\, \underline{u}^o + A^T\, \underline{p} + \dot{\underline{p}})$$

$$+ \delta\, \underline{u}^T(\tfrac{1}{2}\, (W^T + \hat{W}^T)\, \underline{x}^o + Q\, \underline{u}^o + B^T\, \underline{p})\,]\, dt$$

$$+ \delta \underline{x}^T\, \underline{p}\, \Big|_{t=t_f} = 0 \qquad \text{(A-8)}$$

If the costate vector is required to satisfy the following equation,

$$\dot{\underline{p}} + A^T \underline{p} + (S + \hat{S})\, \underline{x}^o + \frac{1}{2}(W + \hat{W})\, \underline{u}^o = \underline{0}; \qquad \underline{p}(t_f) = \underline{0} \tag{A-9}$$

and since the variation in the control is arbitrary, the optimal control law is described by the following equation.

$$Q\, \underline{u}^o + B^T \underline{p} + \frac{1}{2}(W^T + \hat{W}^T)\, \underline{x}^o = \underline{0} \tag{A-10}$$

or

$$u^o = -\, Q^{-1}(B^T \underline{p} + \frac{1}{2}(W^T + \hat{W}^T)\, \underline{x}^o) \tag{A-11}$$

Now, (A-1), (A-9), and (A-11) are the Euler-Lagrange or necessary condition equations for the formal SOC problem.

Appendix B

Newton Raphson Method

The Newton Raphson method is a powerful iterative numerical method which is used extensively to solve nonlinear algebraic equations. This method has a quadratic rate of convergence; convergence occurs provided the initial iterate is suitably "close" to the solution. The recurrence relation which defines the algorithm follows easily from the basic concept of the method as shown by the following derivation for a scalar nonlinear equation with one independent variable.

$$z = g(y) \tag{B-1}$$

The centrol concept involves linearizing the nonlinear equation about the current guess. That is, given a current solution guess, y_i, expand the equation in a Taylor series about y_i.

$$z = g(y) = g(y_i) + \frac{dg}{dy}(y_i)\, dy + y^2 \tag{B-2}$$

where y^2 represents second and higher order terms. Only the linear term is retained and a new guess is found by extrapolating along the tangent line until the approximate function is zero as shown in Fig. B-1. That is,

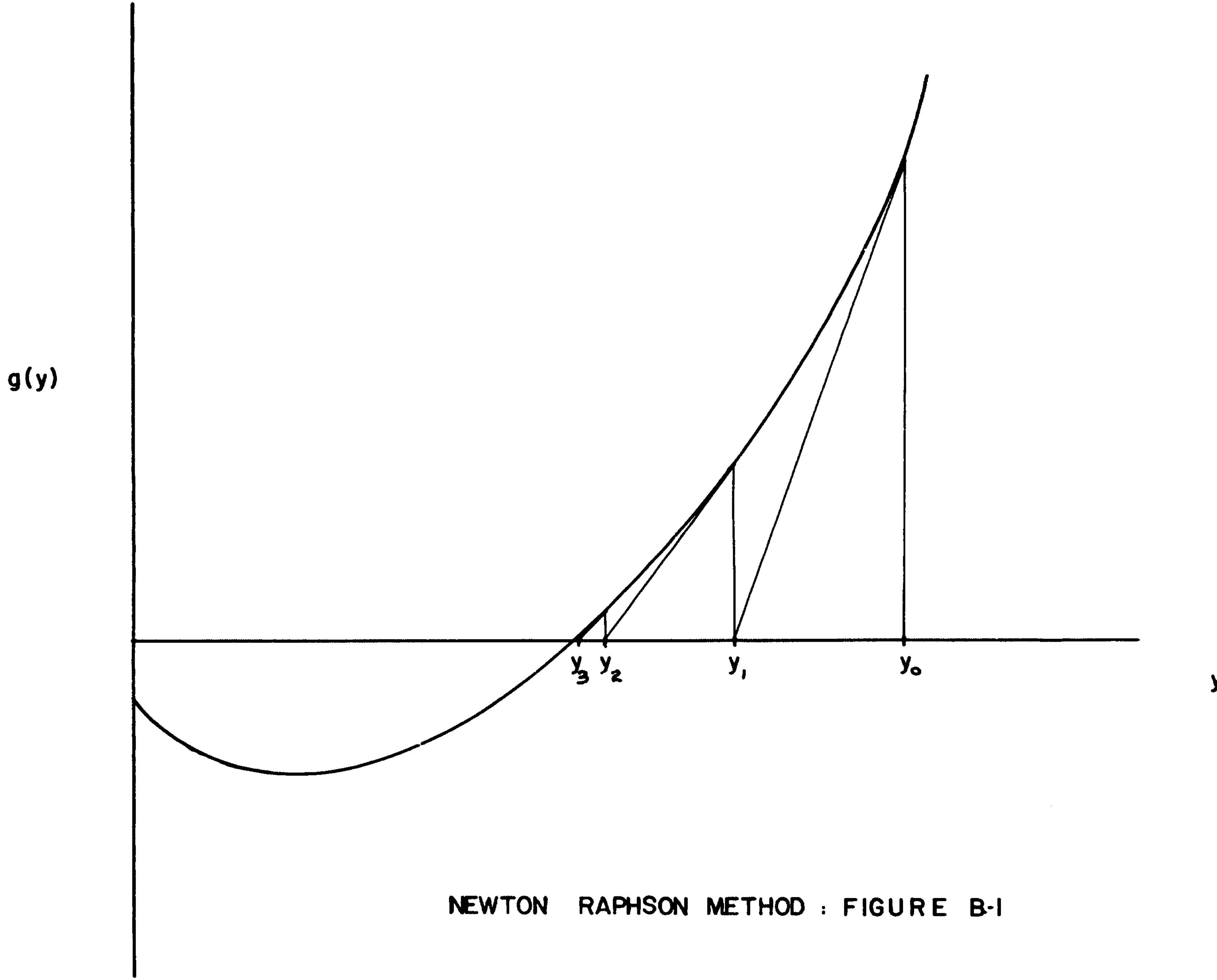
$$z = g(y) \approx \hat{g}(y) = g(y_i) + \frac{dg}{dy}(y_i)\, dy$$

The new guess is chosen by requiring that

$$\hat{g}(y_{i+1}) = 0$$

or

$$\hat{g}(y_{i+1}) = 0 = g(y_i) + \frac{dg}{dy}(y_i)\,(y_{i+1} - y_i) \tag{B-3}$$



NEWTON RAPHSON METHOD : FIGURE B-1

where $dy = y_{i+1} - y_i$. Then Eq. (B-3) may be solved for y_{i+1} to obtain the recursive relation defining the Newton Raphson algorithm,

$$y_{i+1} = y_i - g(y_i)/\frac{dg}{dy}(y_i)$$

Geometrically, this corresponds to finding the tangent to the equation at $y = y_i$ and extending the tangent line until it crosses the horizontal axis, $z = 0$. The intersection of these two lines determines the new guess, y_{i+1}. The process is continued by finding the tangent to $g(y)$ at $y = y_{i+1}$ and extrapolating to determine y_{i+2}.

If convergence problems are encountered, the use of a convergence factor, α, may help. By choosing values of α, $0 < \alpha \leq 1$, it may be possible to alleviate convergence problems at the expense of rate of convergence. Geometrically, the new iterate is found by only extrapolating part way along the tangent line. That is, y_{i+1} is determined by the intersection of the tangent line and $z = c$, where $c = (1-\alpha)\, g(y_i)$.

In a similar manner the Newton Raphson algorithm in function space may be derived. Consider a vector function equation with a vector of independent variables.

$$\underline{z} = \underline{f}(\underline{y}) = 0$$

Given a guess, $\underline{y}_i$, the vector equation is lineared about $\underline{y}_i$.

$$\underline{z} = f(\underline{y}) = f(\underline{y}_i) + \nabla_{\underline{y}}\, \underline{f}\, d\underline{y} + \underline{f}^2$$

where $\nabla_{\underline{y}}\, \underline{f}$ represents the Jacobian or gradient matrix of $\underline{f}$ with respect to $\underline{y}$ and $\underline{f}^2$ denotes second and higher order terms. The equation is linearized by neglecting the higher order terms.

$$\underline{z} = \underline{f}(\underline{y}) \approx \hat{\underline{f}}(\underline{y}) = f(\underline{y}_i) + \nabla_{\underline{y}} \underline{f} \, d\underline{y}$$

The new iterate or guess is determined by the intersection of the tangential plane and the plane, $\underline{z} = 0$.

$$\hat{\underline{f}}(\underline{y}) = 0 = \underline{f}(\underline{y}_i) + \nabla_{\underline{y}} \underline{f} \, d\underline{y}$$

$$d\underline{y} = \underline{y}_{i+1} - \underline{y}_i$$

The recursive relation is given by

$$\underline{y}_{i+1} = \underline{y}_i - (\nabla_{\underline{y}} \underline{f})^{-1} \underline{f}(\underline{y}_i)$$

The actual implementation of this algorithm does not require that the Jacobian matrix be inverted, rather the following linear system of equations is solved for $d\underline{y}$ which leads to $\underline{y}_{i+1}$.

$$\nabla_{\underline{y}} \underline{f} \, d\underline{y} = - \underline{f}(\underline{y}_i)$$

$$\underline{y}_{i+1} = d\underline{y} + \underline{y}_i$$

This is significant from a numerical point of view since fewer operations and hence execution time and error are required to solve a single system of equations as opposed to inverting the coefficient matrix of the system.

APPENDIX C

Digital Computer Programs

To test the effectiveness of the proposed theory, various digital computer programs were coded, debugged, and used. Much of the computational effort was devoted to the solution of steady state problems. Both the SOC-Kleinman and SOCDES algorithms déscribed in Chapter III were implemented and compared. The SOCDES algorithm was found to be superior to SOC-Kleinman especially for the launch vehicle problem. Two versions of the SOCDES algorithm are described in this appendix; SOCDES I solves the steady state unavailable state problem of Chapter IV as well as the unreduced sensitivity problem of Chapter VI while the SOCSEN version solves the reduced SOC sensitivity problem. The basic block diagram for both programs is shown in Fig. C-1. The only significant difference between the two versions is found in the structure of the solution of the Ricatti equations. In SOCDES I the Ricatti equation is formulated in terms of the equivalent linear vector system described in Appendix E while SOCSEN decouples the Ricatti equation into the reduced form and successively solves each of the partition equations via the equivalent vector approach.

The programs consist of a main program which is listed below, and various subroutine programs. The names, call statements, and purpose of the subroutines follow.

NAME: DLIN

USE: DLIN solves systems of linear equations by Gaussian elimination with full pivotal condensation.

$$CX = Y$$

The matrices C, N by N, and Y, N by M, are known and X, N by M, is to be found.

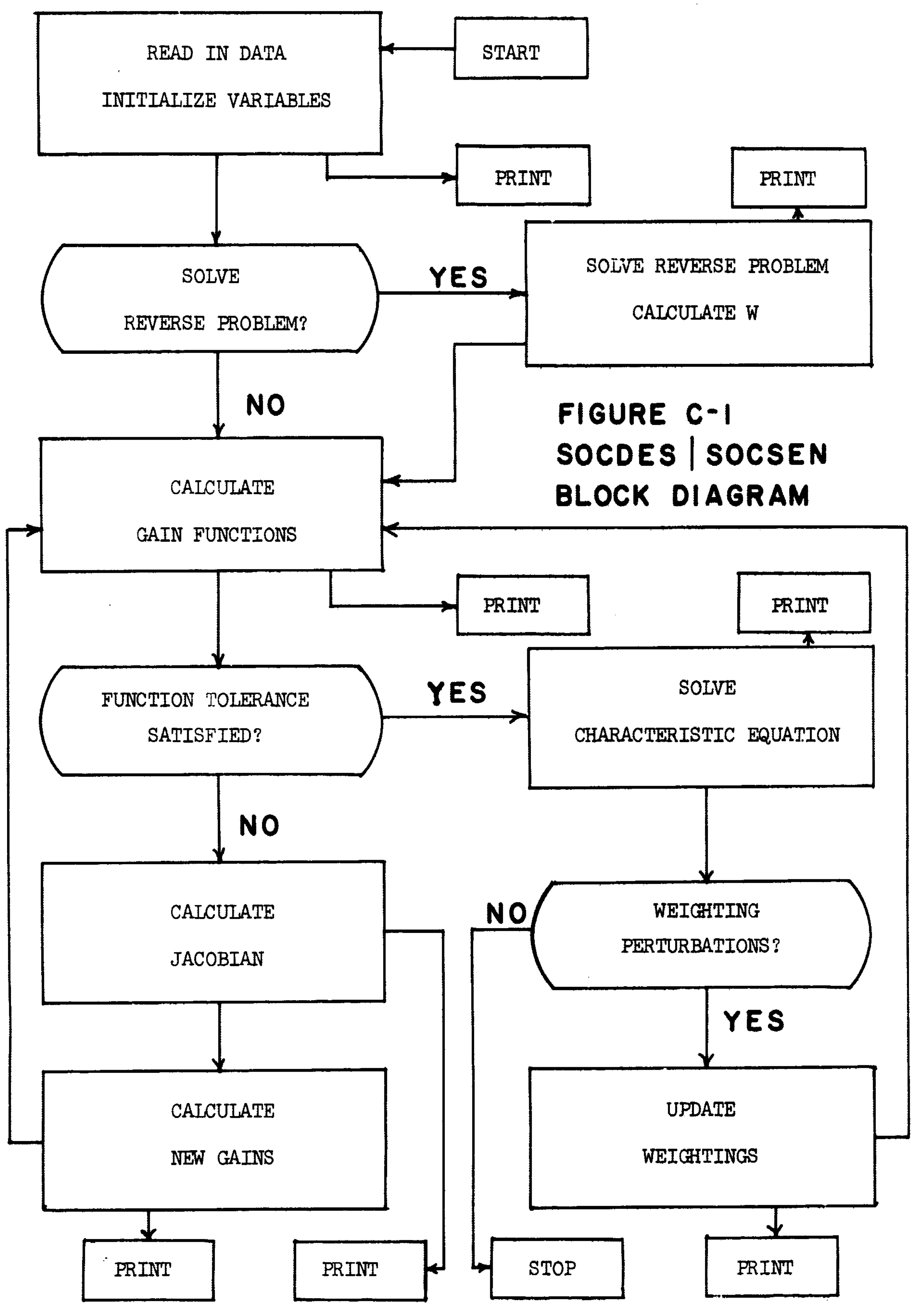


FIGURE C-1
SOCDES | SOCSEN
BLOCK DIAGRAM

CALL: CALL DLIN (R, A, M, N, EPS, IER, ICODE)

R: Y is placed by columns in R; after execution X is placed by columns in R.

A: C is placed by columns in A.

EPS: Pivot error tolerance.

IER: IER is set equal to zero before DLIN is called. If C is singular (pivot element less than EPS), rank of matrix is stored in IER and control is returned to Main.

ICODE: If ICODE is zero, DLIN operates in a normal manner and pivot information is saved. If additional systems of equations with identical coefficient matrices are to be solved, ICODE is set to 1 and a new Y is entered and equations are solved using saved pivot information at a considerable saving in computational effort.

NAME: CHAREQ

USE: CHAREQ formulates characteristic equation of given system matrix.

CALL: CALL CHAREQ (C, N, CCEF)

C: C is the N by N system matrix

COEF: Coefficients of characteristic equation are placed in COEF in descending order, that is coefficient of s^n is placed in position 1 of COEF.

NAME: POTROT

USE: POTROT is used to set up characteristic equation for solution by POLYRT.

CALL: CALL POTROT (C, ICFL, TIME, K1, K2, K3, K4, K5, M)

C: C contains polynomial coefficients in descending order.

M: Order of polynomial.

TIME: Dummy variable.

ICFL: Output control variable.

K1, K2, K3, K4, K5: Feedback gain values.

NAME: POLYRT

USE: POLYRT finds roots of polynomials up to order 99 by Newton Raphson iteration in complex plane.

CALL: CALL POLYRT (M, C, TOL1, RX, RY, RMULT, NR, ISW, CFCTR, IDOUT, IDOUT1)

M: Order of polynomial

C: C contains polynomial coefficients in descending order.

TOL1: If distance between roots in less than TOL1, then roots are assumed to be identical.

RX: Matrix of real parts of roots.

RY: Matrix of imaginary parts of roots.

RMULT: Matrix of scaling factors.

NR: Number of non-identical roots.

ISW: If ISW = 1, the factored polynomial is re-multiplied to form a comparison polynomial.

CFCTR: Matrix of differences between coefficients of original and comparison polynomial.

IDOUT, IDOUT1: Diagnostic print variables.

POLYRT was developed by Ray Ash of the Systems Division of R.P.I. while the rest of these programs were developed by the author.

```
/JOB          4045       CASSIDY,LINES=50
C
C
C          *****      SOCODES I    *****
C
C
C
C   PROGRAM RESTRICTED TO SCALAR CONTROL
C    CERTAIN WRITE STATEMENTS ARE ENTERED AS COMMENT CARDS.  IF TROUBLE
C   DEVELOPS THE  'C' MAY BE REMOVED AND THIS DIAGNOSTIC INFORMATION MAY
C    BE PRINTED. .
      DOUBLE PRECISION ADP(20,20),COEFF(100)
      DIMENSION DS(10,10),W(10)
      DIMENSION GRADT(10,10),D(10)
      DIMENSION S(10,10),P(10,10)
      DIMENSION A(10,10),B(10,1),FBG(1,10)
      DIMENSION AA(10,10),FSA(1,10),F(10),FS(10)
      DIMENSION AIN(28,28),FHA(1,10),ED(784),EG(28,10),EH(784)
      DIMENSION EKK(28,28,10),EK(10,10),FHSA(1,10),EFS(28),FOS(28,28)
      DOUBLE PRECISION E(28,28),EE(784),EF(784),FD(28,28)
    1 FORMAT(16I5)
    2 FORMAT(4E20.5)
    3 FORMAT(1X,1P10E13.4)
    4 FORMAT(1X,10I10)
    6 FORMAT(4D20.5)
    7 FORMAT(1X,1P10D13.4)
    9 FORMAT(//2X,'ITERATION NUMBER ',I5)
   11 FORMAT(////T30,'AFTER ',I3,' ITERATIONS,THE STOPPING TOLERANCE WAS
     1 REACHED.'//)
      CALL TRAPS(0,1,100000)
 5000 CONTINUE
C   NOTE  ***     FBG=K
C   READ IN AND INITIALIZE DATA
      ISS=1
      READ(1,1) IIS,IWC,ISTAB
      WRITE(3,4) IIS,IWC,ISTAB
      APH=0.25E0
      APH=0.1E0
      APH=.001E0
      APH=1.0E0
      KOUNT=0
      READ(1,1) NS,NI,NU
      NL=NS-NU
      NP=(NS*NS+NS)/2
      WRITE(3,4) NS,NI, U,NL,NP
      READ(1,2) EPS,TOL,ESS,SWTOL
      WRITE(3,3) EPS,TOL,ESS,SWTOL
      SWTOL=SWTOL*SWTOL
      TOL=TOL*TOL
      READ(1,2) ((A(I,J),I=1,NS),J=1,NS)
```

```
      READ(1,2) (B(I,1),I=1,NS)
      WRITE(3,3) ((A(I,J),I=1,NS),J=1,NS)
      WRITE(3,3) (B(I,1),I=1,NS)
      READ(1,2) ((S(I,J),I=1,NS),J=1,NS)
      WRITE(3,3) ((S(I,J),I=1,NS),J=1,NS)
      IF(IIS-1) 320,325,320
  320 CONTINUE
      READ(1,2) ((DS(I,J),I=1,NS),J=1,NS)
      WRITE(3,3) ((DS(I,J),I=1,NS),J=1,NS)
      READ(1,2) DQQ
      WRITE(3,3) DQQ
  325 CONTINUE
      READ(1,2) Q
      WRITE(3,3) Q
C
      READ(1,2) (FBG(1,I),I=1,NS)
      WRITE(3,3) (FBG(1,I),I=1,NS)
      DO 15 I=1,NS
      F(I)=0.E0
      FSA(1,I)=0.E0
      W(I)=0.E0
      DO 15 J=1,NS
   15 AA(I,J)=A(I,J)
      IF(ISTAB-1) 334,332,334
  332 CONTINUE
      DO 336 I=1,NS
      DO 336 J=1,NS
      AA(I,J)=0.0E0
  336 AA(I,J)=A(I,J)-B(I,1)*FBG(1,J)
      WRITE(3,3) ((AA(I,J),I=1,NS),J=1,NS)
      DO 337 I=1,NU
  337 FBG(1,I)=0.0E0
  334 CONTINUE
      IF(IWC) 330,328,330
  328 READ(1,2) (W(I),I=1,NS)
      WRITE(3,3) (W(I),I=1,NS)
  330 CONTINUE
      DO 26 I=1,NP
      DO 26 J=1,NP
      DO 27 L=1,NU
   27 EKK(I,J,L)=0.E0
   26 CONTINUE
C  E  IS THE COEFFICIENT MATRIX OF THE EQUIVALENT VECTOR SYSTEM.
C    EKK = DER OF E  WR TO FBG
      DO 250 JK=1,NU
      DO 340 I=1,NS
      DO 340 J=1,NS
  340 EK(I,J)=0.E0
      DO 350 I=1,NS
  350 EK(I,JK)=-B(I,1)
```

```
C       WRITE(3,3) (FK(I,JK),I=1,NS)
        DO 250 I=1,NS
        DO 250 J=1,NS
        DO 250 KL=1,NS
        T1=NS-0.5*J
        L=(J-1)*T1+KL
        IF(KL-J) 243,244,244
  243   T1=NS-0.5*KL
        L=(KL-1)*T1+J
  244   CONTINUE
        T2=NS-0.5*I
        K=(I-1)*T2+KL
        IF(KL-I) 245,246,246
  245   T2=NS-0.5*KL
        K=(KL-1)*T2+I
  246   FAC=1.E0
        IF(I-KL) 248,247,248
  247   FAC=2.E0
  248   CONTINUE
        EKK(K,L,JK)=EKK(K,L,JK)+FAC*EK(J,I)
  250   CONTINUE
C       WRITE(3,3) (((EKK(K,L,JK),K=1,NP),L=1,NP),JK=1,NU)
 1810   CONTINUE
        NI=NI+1
        DO 9010 IOS=ISS,IIS
        DO 9000 ITTT=1,NI
        ITT=ITTT-1
        WRITE(3,9) ITT
        IF(ITT) 17,1000,17
   17   CONTINUE
C    EG=DER(FBG*C*FBG')*WR*FBG
        DO 140 I=1,NS
        DO 140 J=1,NP
  140   EG(J,I)=0.E0
C    ED=   ''P''=-EI*(S+FBG*C*FBG')
        K=0
        DO 142 J=1,NS
        DO 142 I=J,NS
        K=K+1
        EG(K,J)=FBG(1,I)*C+EG(K,J)
        EG(K,I)=FBG(1,J)*C+EG(K,I)
        ED(K)=FF(K)
  142   CONTINUE
C       WRITE(3,3) ((EG(K,I),K=1,NP),I=1,NS)
        DO 148 IK=1,NU
        DO 144 L=1,NP
        EH(L)=0.E0
        DO 144 LL=1,NP
  144   EH(L)=EH(L)-EKK(L,LL,IK)*ED(LL)
C       WRITE(3,3) (EH(L),L=1,NP)
```

```
C    ICODE=1****** DLIN HAS BEEN INITIALIZED. NO NEED TO COMPLETELY
C    RESOLVE SYSTEM OF EQUATIONS.
      ICODE=1
C  EF=-EKK*''P''-EG
C     SOLVE    E''P''=EF
      DO 145 L=1,NP
      EF(L)=EH(L)-EG(L,IK)
  145 CONTINUE
      CALL DLIN(EF,EE,NP,NEQ,EPS,IER,ICODE)
C     WRITE(3,7) (EE(K),K=1,NP)
      K=0
      DO 146 I=1,NS
      DO 146 J=I,NS
      K=K+1
      P(I,J)=EF(K)
  146 P(J,I)=EF(K)
C     WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
      DO 148 KK=1,NU
      GRAD=0.E0
      DO 147 J=1,NS
  147 GRAD=GRAD+B(J,1)*P(J,KK)/Q
      IF(IK-KK) 151,149,151
  149 GRAD=GRAD-1.0E0
  151 CONTINUE
C           GRADIENT=GRAD=QI*B'*P-I
C     WRITE(3,1) KK
C     WRITE(3,3) GRAD
      GRADT(KK,IK)=GRAD
  148 CONTINUE
      WRITE(3,3) ((GRADT(I,J),I=1,NU),J=1,NU)
   94 CONTINUE
      NEQ=1
      IER=0
   91 ESS=1.0E0
      DO 88 I=1,NU
   88 D(I)=F(I)
   89 CONTINUE
C     WRITE(3,3) (D(I),I=1,NU)
      K=0
      DO 92 I=1,NU
      EF(I)=D(I)
      DO 92 J=1,NU
      K=K+1
   92 EE(K)=GRADT(J,I)
      ICODE=0
      CALL DLIN(EF,EE,NU,NEQ,EPS,IER,ICODE)
C     WRITE(3,7) (EF(I),I=1,NU)
      DO 93 I=1,NU
      FSA(1,I)=FPG(1,I)
   93 FBG(1,I)=FPG(1,I)-ESS*APH*EF(I)
```

```
      WRITE(3,3) (FSA(1,I),I=1,NS)
      WRITE(3,3) (FBG(1,I),I=1,NS)
 1000 CONTINUE
C     A=AA-B*FBG'
      DO 10 I=1,NS
      DO 10 J=1,NS
   10 A(I,J)=AA(I,J)-FBG(1,J)*B(I,1)
C     WRITE(3,3) ((A(I,J),I=1,NS),J=1,NS)
C     FORMULATE E ***
      DO 20 K=1,NP
      DO 20 L=1,NP
   20 E(K,L)=0.D0
      DO 40 I=1,NS
      DO 40 J=1,NS
      DO 40 KL=1,NS
C      J,KL****L
      T1=NS-0.5*J
      L=(J-1)*T1+KL
      IF(KL-J) 215,220,220
  215 T1=NS-0.5*KL
      L=(KL-1)*T1+J
  220 CONTINUE
C     I,KL*****K
      T2=NS-0.5*I
      K=(I-1)*T2+KL
      IF(KL-I) 225,230,230
  225 T2=NS-0.5*KL
      K=(KL-1)*T2+I
  230 FAC=1.0
      IF(I-KL) 236,235,236
  235 FAC=2.0
  236 CONTINUE
      E(K,L)=E(K,L)+FAC*A(J,I)
   40 CONTINUE
C     WRITE(3,7) ((E(K,L),K=1,NP),L=1,NP)
      WRITE(3,7) ((E(K,L),K=1,NP),L=1,NP)
      NEQ=1
      ICODE=0
      K=0
C      EF= -''S+FBG*Q*FBG'  ''
      DO 55 I=1,NS
      DO 55 J=I,NS
      K=K+1
   55 EF(K)=-S(I,J)-FBG(1,I)*Q*FBG(1,J)
C     WRITE(3,7) (EF(K),K=1,NP)
      WRITE(3,7) (EF(K),K=1,NP)
      K=0
      DO 70 J=1,NP
      DO 70 I=1,NP
      K=K+1
```

```
      EE(K)=E(I,J)
   70 CONTINUE
      IER=0
      WRITE(3,4) IER
      CALL DLIN(FF,EE,NP,NEQ,EPS,IER,ICODE)
C     CALCULATE P
C     WRITE(3,7) (EF(K),K=1,NP)
      K=0
C      EF=  ''D''
      DO 90 I=1,NS
      DO 90 J=1,NS
      K=K+1
      P(I,J)=EF(K)
   90 P(J,I)=EF(K)
C     WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
      WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
C    F=QI*B'*P+QI*W'-FBG     ****      W='W'/2
      DO 111 I=1,NU
      FS(I)=F(I)
      F(I)=0.E0
      DO 110 J=1,NS
  110 F(I)=F(I)+B(J,I)*P(J,I)/Q
  111 F(I)=F(I)-FBG(1,I)+W(I)/Q
C     W OF PROGRAM IS 1/2  W OF THEORETICAL DEVELOPMENT.
      WRITE(3,3) (FS(I),I=1,NU)
      WRITE(3,3) (F(I),I=1,NU)
      WRITE(3,3) (W(I),I=1,NU)
C     WRITE(3,3) (FHSA(1,I),I=1,NS)
C     WRITE(3,3) (FHA(1,I),I=1,NS)
      IF(ISW-1) 131,132,131
  131 CONTINUE
      IF(ITT) 112,8000,112
 8000 IF(IWC-1) 9000,8002,9000
C      CALCULATE  W
 8002 DO 8001 I=1,NU
      W(I)=-F(I)
      F(I)=0.0E0
 8001 CONTINUE
      IWC=2
      WRITE(3,3) (W(I),I=1,NU)
      WRITE(3,3) ((S(I,J),I=1,NS),J=1,NS)
      GO TO 9001
  112 CONTINUE
  150 CONTINUE
  132 CONTINUE
C     CHECK GAINS
      DO 154 I=1,NU
      DD=F(I)
      DD=DD*DD
      IF(DD-TOL) 154,160,160
```

```
  154 CONTINUE
      WRITE(3,11) ITT
      WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
      GO TO 9001
  160 KOUNT=KOUNT+1
 9000 CONTINUE
 9001 CONTINUE
      DO 9005 I=1,NS
      DO 9005 J=1,NS
 9005 S(I,J)=S(I,J)+DS(I,J)
      WRITE(3,3) ((S(I,J),I=1,NS),J=1,NS)
      Q=Q+DCQ
      WRITE(3,3) Q
      DO 9008 I=1,NS
      DO 9008 J=1,NS
 9008 ADP(I,J)=A(I,J)
      CALL CHAREQ(ADP,NS,COEFF)
      NSS=NS+1
      WRITE(3,7) (COEFF(I),I=1,NSS)
      TIME=0.0E0
      A3=0.0E0
      A4=0.0E0
      A5=0.0E0
      A1=FBG(1,1)
      A2=FBG(1,2)
      ICFL=0
      CALL POTROT(COEFF,ICFL,TIME,A1,A2,A3,A4,A5,NS)
 9010 CONTINUE
      GO TO 5000
      END
```

```
/JOB          4045       CASSIDY,LINES=50
C
C
C        *****      SOCSEN I     *****
C
C
C
C  PROGRAM RESTRICTED TO SCALAR CONTROL
C   CERTAIN WRITE STATEMENTS ARE ENTERED AS COMMENT CARDS.  IF TROUBLE
C  DEVELOPS THE  'C' MAY BE REMOVED AND THIS DIAGNOSTIC INFORMATION MAY
C   BE PRINTED.
      DOUBLE PRECISION ADP(20,20),COEFF(100)
      DIMENSION DRAT(14,14)
      DIMENSION GRADT(14,14),D(14)
      DIMENSION A(14,14),B(14,1),FBG(1,14)
      DIMENSION AA(14,14),FSA(1,14),F(14),FS(14)
      DIMENSION W(14),P(14,14)
      DIMENSION ED(105),FG(105,14),EH(105)
      DIMENSION EKK(105,105,2)
      DIMENSION FK(14,14)
      DIMENSION AQ(7,7),S1(7,7),S2(7,7),S3(7,7),DS1(7,7),DS2(7,7)
      DIMENSION DS3(7,7)
      DOUBLE PRECISION DE(784)
      DOUBLE PRECISION E1(28,28)
      DOUBLE PRECISION E(28,28),EE(784),EF(784)
    1 FORMAT(16I5)
    2 FORMAT(4E20.5)
    3 FORMAT(1X,1P10E13.4)
    4 FORMAT(1X,10I10)
    6 FORMAT(4D20.5)
    7 FORMAT(1X,1P10D13.4)
    9 FORMAT(//2X,'ITERATION NUMBER ',I5)
   11 FORMAT(////T30,'AFTER ',I3,' ITERATIONS,THE STOPPING TOLERANCE WAS
     1 REACHED.'//)
 5000 CONTINUE
C     PROGRAM SLVES SOC SENSITIVITY PROBLEM VIA REDUCED PROBLEM
C  FORMULATION.
C    PROGRAM LIMITED TO SINGLE PARAMETER BUT STRUCTURE ALLOWS EXTENSION
C  TO MULTIPLE PARAMETERS.
C IF LOW ORDER SENSITIVITY PROBLEM IS TO BE SOLVED, THEN IT MAY BE
C  FORMULATED IN UNREDUCED FORM AND SOLVED WITH SOCDES I.
C  NS2  IS THE ORDER OF ORIGINAL SYSTEM.
C  READ IN AND INITIALIZE DATA
C  NOTE  ***     FBG=K
      ISS=1
      READ(1,1) IIS,IWC
      WRITE(3,4) IIS,IWC
      APH=0.25E0
      APH=0.1E0
      APH=.001E0
```

```
      APH=1.0E0
      KOUNT=0
      READ(1,1) NS,NI,NU
      NL=NS-NU
      NP=(NS*NS+NS)/2
      NS2=NS/2
      NS3=NS2-1
      NP2=(NS2*NS2+NS2)/2
      WRITE(3,4) NS,NI,NU,NL,NP
      READ(1,2) EPS,TOL,ESS,SWTOL
      WRITE(3,3) EPS,TOL,ESS,SWTOL
      SWTOL=SWTOL*SWTOL
      TOL=TOL*TOL
      READ(1,2) ((AA(I,J),I=1,NS2),J=1,NS2)
      READ(1,2) ((AQ(I,J),I=1,NS2),J=1,NS2)
      READ(1,2) (B(I,1),I=1,NS2)
      READ(1,2) ((S1(I,J),I=1,NS2),J=1,NS2)
      READ(1,2) ((S2(I,J),I=1,NS2),J=1,NS2)
      READ(1,2) ((S3(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((AA(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((AQ(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) (B(I,1),I=1,NS2)
      WRITE(3,3) ((S1(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((S2(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((S3(I,J),I=1,NS2),J=1,NS2)
      IF(IIS-1) 320,325,320
  320 CONTINUE
      READ(1,2) ((DS1(I,J),I=1,NS2),J=1,NS2)
      READ(1,2) ((DS2(I,J),I=1,NS2),J=1,NS2)
      READ(1,2) ((DS3(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((DS1(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((DS2(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((DS3(I,J),I=1,NS2),J=1,NS2)
  325 CONTINUE
      DO 1300 I=1,NS2
      II=I+NS2
 1300 B(II,1)=0.0E0
      READ(1,2) Q
      WRITE(3,3) Q
C
      READ(1,2) (FBG(1,I),I=1,NS)
      WRITE(3,3) (FBG(1,I),I=1,NS)
      DO 15 I=1,NS
      F(I)=0.E0
      FSA(1,I)=0.E0
      W(I)=0.E0
   15 CONTINUE
      IF(IWC) 330,328,330
  328 READ(1,2) (W(I),I=1,NS)
      WRITE(3,3) (W(I),I=1,NS)
```

```
  330 CONTINUE
      DO 26 I=1,NP
      DO 26 J=1,NP
      DO 27 L=1,NU
   27 EKK(I,J,L)=0.E0
   26 CONTINUE
C  E  IS THE COEFFICIENT MATRIX OF THE EQUIVALENT VECTOR SYSTEM.
C     EKK = DER OF E  WR TO FBG
      DO 250 JK=1,NU
      DO 340 I=1,NS
      DO 340 J=1,NS
  340 EK(I,J)=0.E0
      JKK=JK+NS2
      DO 350 I=1,NS2
      II=I+NS2
      EK(II,JKK)=-B(I,1)
  350 EK(I,JK)=EK(II,JKK)
C     WRITE(3,3) (EK(I,JK),I=1,NS)
      DO 250 I=1,NS
      DO 250 J=1,NS
      DO 250 KL=1,NS
      T1=NS-0.5*J
      L=(J-1)*T1+KL
      IF(KL-J) 243,244,244
  243 T1=NS-0.5*KL
      L=(KL-1)*T1+J
  244 CONTINUE
      T2=NS-0.5*I
      K=(I-1)*T2+KL
      IF(KL-I) 245,246,246
  245 T2=NS-0.5*KL
      K=(KL-1)*T2+I
  246 FAC=1.E0
      IF(I-KL) 248,247,248
  247 FAC=2.E0
  248 CONTINUE
      EKK(K,L,JK)=EKK(K,L,JK)+FAC*EK(J,I)
  250 CONTINUE
C     WRITE(3,3) (((EKK(K,L,JK),K=1,NP),L=1,NP),JK=1,NU)
      NI=NI+1
      DO 9010 IDS=ISS,IIS
      DO 9000 ITTT=1,NI
      ITT=ITTT-1
      WRITE(3,9) ITT
      IF(ITT) 17,1000,17
   17 CONTINUE
C  EG=  DER OF((FBG*C*FBG')BAR) WR TO FBG
      IF(INO-1) 136,135,136
  135 INO=0
      GO TO 94
```

```
  136 CONTINUE
 1500 CONTINUE
      DO 140 I=1,NS
      DO 140 J=1,NP
  140 EG(J,I)=0.E0
C   ED  =  ''P''   =-EI*''((S+(FBG*Q*FBG')BAR)''
      K=0
      DO 142 J=1,NS
      DO 142 I=J,NS
      K=K+1
      EG(K,J)=FBG(1,I)*Q+EG(K,J)
      EG(K,I)=FBG(1,J)*Q+EG(K,I)
      ED(K)=P(I,J)
  142 CONTINUE
      K=NP-NP2
      DO 480 J=1,NS2
      DO 480 I=J,NS2
      K=K+1
      EG(K,J)=FBG(1,I)*Q+EG(K,J)
  480 EG(K,I)=FBG(1,J)*Q+EG(K,I)
C     WRITE(3,3) ((EG(K,I),K=1,NP),I=1,NS)
      DO 148 IK=1,NU
      DO 144 L=1,NP
      EH(L)=0.E0
      DO 144 LL=1,NP
  144 EH(L)=EH(L)-FKK(L,LL,IK)*ED(LL)
C     WRITE(3,3) (EH(L),L=1,NP)
C   ICODE=1****** DLIN HAS BEEN INITIALIZED. NO NEED TO COMPLETELY
C    RESOLVE SYSTEM OF EQUATIONS.
      ICODE=1
C   EF=-EKK*''P''-EG
C     SOLVE     F''P''=EF
C   TO SAVE MEMORY, P IS USED IN MANY DIFFERENT WAYS.
C SINCE THERE IS ONLY ONE PARAMETER, THE UNREDUCED RICATTI MATRIX IS
C   DECOUPLED INTO THREE(3) NS2 BY NS2 BLOCKS.
      L=0
      DO 145 J=1,NS
      DO 145 I=J,NS
      L=L+1
      P(I,J)=EH(L)-EG(L,IK)
      P(J,I)=P(I,J)
  145 CONTINUE
C   SOLVE FOR P3, E ''P3''=-''(P3)''
      K=0
      II=NS2+1
      DO 1410 I=II,NS
      DO 1410 J=I,NS
      K=K+1
 1410 EF(K)=P(I,J)
      ICODE=1
```

```
      CALL DLIN(EF,EE,NP2,NEQ,EPS,IER,ICODE)
C     ONLY P1 MUST BE SAVED.
      K=0
      II=NS2+1
      DO 1420 I=II,NS
      DO 1420 J=I,NS
      K=K+1
      P(I,J)=EF(K)
 1420 P(J,I)=P(I,J)
      K=0
C  DRAT=-1/2 P2-1/2P3*AQ
      DO 1450 J=1,NS2
      DO 1450 I=1,NS2
      II=I+NS2
      K=K+1
      DRAT(I,J)=P(II,J)*0.5
      DO 1450 L=1,NS2
      LL=L+NS2
 1450 DRAT(I,J)=DRAT(I,J)-0.5*(P(II,LL)*AQ(L,J))
C SOLVE FOR THE SKEW*SYMMETRIC PORTION OF P2.
C    SOLVE  DRAT-DRAT.T
      K=0
      DO 1452 J=1,NS3
      J1=J+1
      DO 1452 I=J1,NS2
      K=K+1
 1452 EF(K)=DRAT(I,J)-DRAT(J,I)
      ICODE=1
      CALL DLON(EF,DE,NP3,NEQ,EPS,IER,ICODE)
      K=0
      P(NS,NS2)=C.0E0
      DO 1453 J=1,NS3
      J1=J+1
      JJ=J+NS2
      P(JJ,J)=C.0E0
      DO 1453 I=J1,NS2
      K=K+1
      II=I+NS2
      P(JJ,I)=-EF(K)
 1453 P(II,J)=EF(K)
C  SOLVE FOR THE SYMMETRIC PORTION OF P2.
C    SOLVE DRAT+DRAT.T
      K=0
      DO 1456 J=1,NS2
      DO 1456 I=J,NS2
      K=K+1
 1456 EF(K)=DRAT(I,J)+DRAT(J,I)
      ICODE=1
      CALL DLIN(EF,EE,NP2,NEQ,EPS,IER,ICODE)
      K=0
```

```
      DO 1457 J=1,NS2
      DO 1457 I=J,NS2
      K=K+1
      II=I+NS2
      JJ=J+NS2
      P(II,J)=EF(K)+P(II,J)
      IF(I-J) 1382,1383,1382
 1383 P(JJ,I)=P(II,J)
      GO TO 1384
 1382 P(JJ,I)=P(JJ,I)+EF(K)
 1384 CONTINUE
      P(J,II)=P(II,J)
      P(I,JJ)=P(JJ,I)
 1457 CONTINUE
      K=0
      DO 1470 J=1,NS2
      DO 1470 I=J,NS2
      K=K+1
      EF(K)=P(I,J)
      DO 1470 L=1,NS2
      LL=L+NS2
 1470 EF(K)=EF(K)-P(LL,I)*AQ(L,J)-AQ(L,I)*P(LL,J)
C SOLVE FOR P1    E''P1''=-''(P2'*AQ+AQ'*P2P+1)''
      ICODE=1
      CALL DLIN(EF,EE,NP2,NEQ,EPS,IER,ICODE)
      K=0
      DO 1480 J=1,NS2
      DO 1480 I=J,NS2
      K=K+1
      P(I,J)=EF(K)
 1480 P(J,I)=P(I,J)
C     WRITE(3,7) (EE(K),K=1,NP)
C     WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
      DO 148 KK=1,NU
      GRAD=0.E0
      DO 147 J=1,NS
  147 GRAD=GRAD+B(J,1)*P(J,KK)/Q
      IF(IK-KK) 151,149,151
  149 GRAD=GRAD-1.0E0
  151 CONTINUE
C           GRADIENT=GRAD=QI*B'*P-I
C     WRITE(3,1) KK
      WRITE(3,1) KK
C     WRITE(3,3) GRAD
      WRITE(3,3) GRAD
      GRADT(KK,IK)=GRAD
  148 CONTINUE
      WRITE(3,3) ((GRADT(I,J),I=1,NU),J=1,NU)
   94 CONTINUE
      NEQ=1
```

```
      IER=0
   91 ESS=1.0E0
      DO 88 I=1,NU
   88 D(I)=F(I)
   89 CONTINUE
C     WRITE(3,3) (D(I),I=1,NU)
      K=0
      DO 92 I=1,NU
      EF(I)=D(I)
      DO 92 J=1,NU
      K=K+1
   92 EE(K)=GRADT(J,I)
      ICODE=0
      CALL DLIN(EF,EE,NU,NEQ,EPS,IER,ICODE)
C     WRITE(3,7) (EF(I),I=1,NU)
      DO 93 I=1,NU
      FSA(1,I)=FBG(1,I)
   93 FBG(1,I)=FBG(1,I)-ESS*APH*EF(I)
      WRITE(3,3) (FSA(1,I),I=1,NS)
      WRITE(3,3) (FBG(1,I),I=1,NS)
 1000 CONTINUE
C     A=AA-B*FBG'
      DO 10 I=1,NS2
      DO 10 J=1,NS2
   10 A(I,J)=AA(I,J)-FBG(1,J)*B(I,1)
C     WRITE(3,3) ((A(I,J),I=1,NS),J=1,NS)
C     FORMULATE E ***
      DO 20 K=1,NP2
      DO 20 L=1,NP2
      E1(K,L)=0.0D0
   20 E(K,L)=0.D0
      DO 40 I=1,NS2
      DO 40 J=1,NS2
      DO 40 KL=1,NS2
C      J,KL****L
      T1=NS2-0.5*J
      FT1=1.0E0
      L=(J-1)*T1+KL
      IF(KL-J) 215,220,220
  215 T1=NS2-0.5*KL
      FT1=-FT1
      L=(KL-1)*T1+J
  220 CONTINUE
C     I,KL*****K
      T2=NS2-0.5*I
      K=(I-1)*T2+KL
      IF(KL-I) 225,230,230
  225 T2=NS2-0.5*KL
      FT1=-FT1
      K=(KL-1)*T2+I
```

```
  230 FAC=1.0
      IF(I-KL) 236,235,236
  235 FAC=2.0
  236 CONTINUE
      E(K,L)=E(K,L)+FAC*A(J,I)
      E1(K,L)=E1(K,L)+FT1*FAC*A(J,I)
   40 CONTINUE
C     WRITE(3,7) ((E(K,L),K=1,NP),L=1,NP)
C  SOLVE FOR P3  E''P3'' =-''(S+FBG*Q*FBG')''
      NEQ=1
      ICODE=0
      K=0
C      EF= -''S+FBG*Q*FBG'  ''
C    CALCULATE P3=-S3-KQKT
      DO 55 I=1,NS2
      DO 55 J=I,NS2
      K=K+1
   55 EF(K)=-S3(I,J)-FBG(1,I)*Q*FBG(1,J)
C     WRITE(3,7) (EF(K),K=1,NP)
      K=0
      DO 70 J=1,NP2
      DO 70 I=1,NP2
      K=K+1
      EE(K)=E(I,J)
   70 CONTINUE
      IER=0
      WRITE(3,4) IER
      CALL DLIN(EF,EE,NP2,NEQ,EPS,IER,ICODE)
C     CALCULATE P
C     WRITE(3,7) (EF(K),K=1,NP)
      K=0
      II=NS2+1
      DO 90 I=II,NS
      DO 90 J=I,NS
      K=K+1
      P(I,J)=EF(K)
   90 P(J,I)=EF(K)
C        FORM  -S2-P3A0  /2
      K=0
      DO 300 J=1,NS2
      DO 300 I=1,NS2
      II=I+NS2
      K=K+1
      DRAT(I,J)=-0.5*S2(I,J)
      DO 300 L=1,NS2
      LL=L+NS2
  300 DRAT(I,J)=DRAT(I,J)-0.5*(P(II,LL)*A0(L,J))
C  DRAT=-1/2 S2-1/2 P3*A0
C SOLVE FOR THE SKEW*SYMMETRIC PORTION OF P2.
C    SOLVE   DRAT-DRAT.T
```

```
C EE IS THE COEFFICIENT MATRIX FOR EQIVALENT SKEW*SYMMETRIC VECTOR SYSTEM.
      K=0
      DO 1336 J=1,NS3
      J1=J+1
      DO 1336 I=J1,NS2
      K=K+1
 1336 EF(K)=DRAT(I,J)-DRAT(J,I)
      NP3=NP2-NS2
      ICODE=0
C     FORM NEW EE FROM OLD E.
      K=0
      KJ=0
      DO 1369 J=1,NS2
      DO 1369 I=J,NS2
      KJ=KJ+1
      IF(J-I) 1361,1369,1361
 1361 KI=0
      DO 1368 JJ=1,NS2
      DO 1368 II=JJ,NS2
      KI=KI+1
      IF(JJ-II) 1364,1368,1364
 1364 K=K+1
      DE(K)=E1(KI,KJ)
 1368 CONTINUE
 1369 CONTINUE
      CALL DLIN(EF,DE,NP3,NEQ,EPS,IER,ICODE)
      K=0
      DO 1338 J=1,NS3
      JJ=J+NS2
      J1=J+1
      P(JJ,J)=0.0E0
      DO 1338 I=J1,NS2
      K=K+1
      II=I+NS2
      P(JJ,I)=-EF(K)
 1338 P(II,J)=EF(K)
C     SOLVE DRAT+DRAT.T
      K=0
      DO 1339 J=1,NS2
      DO 1339 I=J,NS2
      K=K+1
 1339 EF(K)=DRAT(I,J)+DRAT(J,I)
C  SOLVE FOR THE SYMMETRIC PORTION OF P2.
      ICODE=1
      CALL DLIN(EF,EE,NP2,NEQ,EPS,IER,ICODE)
      P(NS,NS2)=0.0E0
      K=0
C SOLVE FOR P1  E''P1''=-''(P2'*AQ+AQ'*P2+S1+FBG*Q*FBG')''
      DO 1340 J=1,NS2
      DO 1340 I=J,NS2
```

```
      K=K+1
      II=I+NS2
      JJ=J+NS2
      P(II,J)=P(II,J)+EF(K)
      IF(I-J) 1372,1373,1372
 1373 P(JJ,I)=P(II,J)
      GO TO 1374
 1372 P(JJ,I)=P(JJ,I)+EF(K)
 1374 CONTINUE
      P(I,JJ)=P(JJ,I)
 1340 P(J,II)=P(II,J)
C    CALCULATE P1,    LHS=-P2T*AQ-AQ*P2-S1-KOKT
      K=0
      DO 1350 J=1,NS2
      DO 1350 I=J,NS2
      K=K+1
      EF(K)=-S1(I,J)-FBG(1,I)*Q*FBG(1,J)
      DO 1350 L=1,NS2
      LL=L+NS2
 1350 EF(K)=EF(K)-P(LL,I)*AQ(L,J)-AQ(L,I)*P(LL,J)
      ICODE=1
      CALL DLIN(EF,EC,NP2,NEQ,EPS,IER,ICODE)
      K=0
      DO 1360 J=1,NS2
      DO 1360 I=J,NS2
      K=K+1
      P(I,J)=EF(K)
 1360 P(J,I)=P(I,J)
C     WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
      WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
C    F=QI*B'*P+QI*W'-FBG      ****        W='W'/2
      DO 111 I=1,NU
      FS(I)=F(I)
      F(I)=0.E0
      DO 110 J=1,NS
  110 F(I)=F(I)+B(J,1)*P(J,I)/Q
  111 F(I)=F(I)-FBG(1,I)+W(I)
C     'W(I)'=QINV.W(I)/2
C     W OF PROGRAM IS 1/2  W OF THEORETICAL DEVELOPMENT.
      WRITE(3,3) (FS(I),I=1,NU)
      WRITE(3,3) (F(I),I=1,NU)
      WRITE(3,3) (W(I),I=1,NU)
C     WRITE(3,3) (FHSA(1,I),I=1,NS)
C     WRITE(3,3) (FHA(1,I),I=1,NS)
      IF(ISW-1) 131,132,131
  131 CONTINUE
C     CHECK SIZE OF FHHA
      FSMAX=FUMAX
      FUMAX=0.E0
      DO 156 I=1,NU
```

```
      D3=F(I)*F(I)
      IF(D3-FUMAX) 156,156,153
  153 FUMAX=D3
  156 CONTINUE
      IF(ITT) 112,8000,112
 8000 IF(IWC-1) 9000,8002,9000
 8002 DO 8001 I=1,NU
      W(I)=-F(I)
      F(I)=0.0E0
 8001 CONTINUE
      IWC=2
      WRITE(3,3) (W(I),I=1,NU)
      WRITE(3,3) ((S1(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((S2(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((S3(I,J),I=1,NS2),J=1,NS2)
      GO TO 9001
  112 CONTINUE
  132 CONTINUE
C     CHECK GAINS
      DO 154 I=1,NU
      DD=F(I)
      DD=DD*DD
      IF(DD-TOL) 154,160,160
  154 CONTINUE
      WRITE(3,11) ITT
      WRITE(3,3) ((P(I,J),I=1,NS),J=1,NS)
      GO TO 9001
  160 KOUNT=KOUNT+1
      IF(ISW-1) 161,9000,161
  161 CONTINUE
 9000 CONTINUE
 9001 CONTINUE
      DO 9005 I=1,NS2
      DO 9005 J=1,NS2
      S1(I,J)=S1(I,J)+DS1(I,J)
      S2(I,J)=S2(I,J)+DS2(I,J)
 9005 S3(I,J)=S3(I,J)+DS3(I,J)
      WRITE(3,3) ((S1(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((S2(I,J),I=1,NS2),J=1,NS2)
      WRITE(3,3) ((S3(I,J),I=1,NS2),J=1,NS2)
      DO 9008 I=1,NS2
      DO 9008 J=1,NS2
 9008 ADP(I,J)=A(I,J)
      CALL CHAREQ(ADP,NS2,COEFF)
      NSS=NS2+1
      WRITE(3,7) (COEFF(I),I=1,NSS)
      TIME=0.0E0
      A3=0.0E0
      A4=0.0E0
      A5=0.0E0
      A1=FBG(1,1)
      A2=FBG(1,2)
      ICFL=0
      CALL PCTROT(COEFF,ICFL,TIME,A1,A2,A3,A4,A5,NS2)
 9010 CONTINUE
      GO TO 5000
      END
```

Appendix D

Derivation of the $\hat{S}$ and $\hat{W}$ Definitions for the SOC Sensitivity Problem

The rationale which governs the choice of $\hat{S}$ and $\hat{W}$ is the same as that of the ordinary SOC problem. That is, $\hat{W}$ is chosen to insure that the desired gain structure is obtained and $\hat{S}$ is chosen to simplify the structure of the equations. Recall that the general steady state or infinite time interval SOC index is of the following form

$$J = \frac{1}{2} \int_{t_o}^{\infty} (\hat{\underline{x}}^T S \hat{\underline{x}} + \hat{\underline{x}}^T \hat{S} \hat{\underline{x}} + \hat{\underline{x}}^T W \hat{\underline{u}} + \hat{\underline{x}}^T \hat{W} \hat{\underline{u}} + \hat{\underline{u}}^T Q \hat{\underline{u}})\, dt \qquad \text{(D-1)}$$

and the unreduced Ricatti equation and control law are given below.

$$\bar{A}^T P + P \bar{A} + S + \hat{S} - (P\bar{B} + \frac{W+\hat{W}}{2}) Q^{-1}(\bar{B}^T P + \frac{W+\hat{W}^T}{2}) = 0 \qquad \text{(D-2)}$$

$$\hat{\underline{u}} = - \bar{K}^T \hat{\underline{x}} \qquad \text{(D-3)}$$

$$\bar{K}^T = Q^{-1}(\bar{B}^T P + \frac{W^T+\hat{W}^T}{2}) \qquad \text{(D-4)}$$

Suppose that the last L states of the system state vector are unavailable and note that the SOC sensitivity structure requires that the first NS - L states of each sensitivity partition block in the augmented state vector have gains identical to the available state gains while the last L gains of each block are to be zero. To facilitate the discussion partition W into blocks and define the matrix $I_{I,J}$ as follows:

$$\hat{W} = \begin{bmatrix} \hat{W}_{11} & \cdots & \hat{W}_{1,\, NPA+1} \\ \vdots & & \vdots \\ \hat{W}_{NPA+1,\, 1} & \cdots & \hat{W}_{NPA+1,\, NPA+1} \end{bmatrix} \tag{D-5}$$

where $\hat{W}_{I,J}$ is a NS by NC partition block matrix and

$$I_{I,J} = \begin{bmatrix} 0 & \cdots & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & & \vdots \\ 0 & \cdots & I_{NS,NS} & & \ddots & \vdots \\ \vdots & & & \ddots & & \vdots \\ \vdots & & & & 0 & 0 \\ \vdots & & & & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & 0 \end{bmatrix}$$

where $I_{I,J}$ is NS(NPA+1) by NS(NPA+1) and $I_{NS,NS}$ is a NS by NS identity matrix and it occupies the I,J block position. Note that the expression $I_{I,J}\,\hat{W}$ isolates the $\hat{W}_{IJ}$ block of $\hat{W}$ and thus may be used to define these blocks. To separate the portions of each block corresponding to the available and unavailable states the following notation is useful.

$$I^1_{I,J} = \begin{bmatrix} 0 & \cdots & 0 & \cdots & \cdots & 0 \\ 0 & \cdots & 0 & \ddots & & \vdots \\ \vdots & & I^1_{NS,NS} & & \ddots & \vdots \\ \vdots & & & \ddots & 0 & 0 \\ \vdots & & & & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & 0 \end{bmatrix}$$

when this matrix is identical to I_{IJ} except that the last L diagonal elements of $I^1_{NS,NS}$ are zero. An additional matrix $I^2_{I,J}$ is defined as follows:

$$I^2_{I,J} = I_{I,J} - I^1_{I,J} \tag{D-6}$$

As a final notational consideration, let $[[A]]_{I,J}$ be a matrix which is equal to the I,J portion block of the matrix A.

To obtain the desired gains structure for the system states, $\hat{W}_{11}$ is chosen as follows:

$$\hat{W}_{11} = -2 \left[\left[I_{11}^{2} (P\bar{B} + \frac{W}{2})\right]\right]_{1,1} \tag{D-7}$$

To insure that these gain values are repeated for the sensitivity blocks, the remainder of $\hat{W}$ is chosen as follows. For I = J

$$\hat{W}_{I,J} = 2 \left\{ \left[\left[I_{11}^{1} (P\bar{B} + \frac{W}{2})\right]\right]_{1,1} - \left[\left[P\bar{B} + \frac{W}{2}\right]\right]_{I,J} \right\} \tag{D-8}$$

and for $I \neq J$

$$\hat{W}_{I,J} = -2 \left[\left[P\bar{B} + \frac{W}{2}\right]\right]_{I,J} \tag{D-9}$$

With these definitions the feedback gain matrix assumes this structure.

$$\bar{K}^T = \begin{bmatrix} K^T & 0 & \cdot & \cdot & 0 \\ 0 & \ddots & & & \cdot \\ \cdot & & \ddots & & 0 \\ \cdot & & & \ddots & \\ 0 & \cdot & \cdot & 0 & K^T \end{bmatrix}$$

and

$$K = [[\bar{K}]]_{1,1} = \left[\left[I_{11} (P\bar{B} + \frac{W}{2}) Q^{-1} \right]\right]_{1,1}$$

As in the unavailable state problem, $\hat{S}$ is required to be symmetric and is chosen to cancel the W and $\hat{W}$ terms. Hence,

$$\hat{S} = \frac{1}{2} ((W + \hat{W}) \bar{K}^T + \bar{K}(W^T + \hat{W}^T)) \tag{D-10}$$

With these definitions the reduced Ricatti equation becomes

$$\bar{A}_{\bar{K}}^{T} P + P \bar{A}_{\bar{K}} + S + \bar{K} Q \bar{K}^{T} = 0 \qquad \text{(D-11)}$$

and the optimal value of the index may be <u>expressed</u> as

$$J^{o} = \frac{1}{2} \int_{t_o}^{t_f} (\underline{\hat{x}}^{o^T} S \underline{\hat{x}}^{o} + \underline{\hat{u}}^{o^T} Q \underline{\hat{u}}^{o}) \, dt = \frac{1}{2} \underline{\hat{x}}^{T} P \underline{\hat{x}} \Big|_{t=t_o} \qquad \text{(D-12)}$$

Appendix E

Derivation of Equivalent System of Linear Vector Equations

Since it is often difficult to handle the Ricatti equation in its matrix form, it is convenient to formulate a vector from the elements of the Ricatti matrix and derive the equivalent vector equation. Consider the matrix equation

$$F(D, A, P) = A^T P + PA - D \tag{E-1}$$

where the matrices are NS by NS. This matrix equation is equivalent to $(NS)^2$ scalar equations. If D is symmetric and a unique solution to (E-1) is assumed, and since P and P^T satisfy (E-1), P is also symmetric. In this case the number of independent equations reduces to $NP = \frac{NS(NS + 1)}{2}$ corresponding to the diagonal and either upper and lower triangular terms.

It is clear that (E-1) is linear in P; for reasons of notation and manipulation it is convenient to formulate (E-1) in the standard format for linear equations which is denoted below. That is

$$A^T P + PA = D \tag{E-2}$$

or in terms of the Kronecker product notation

$$(A^T * I + I * A) P = D \tag{E-3}$$

The equivalent vector expression is

$$\text{"A" "P"} = \text{"D"} \tag{E-4}$$

where "P" and "D" are NP element vectors formed from P and D as follows.

$$\text{"P"}^T = (P_{1,1};\ldots;P_{NS,1};\ P_{2,2};\ldots;P_{NS,2};\ldots;P_{NS,NS})$$

$$\text{"D"}^T = (D_{1,1};\ldots;D_{NS,1};\ D_{2,2};\ldots;D_{NS,2};\ldots;D_{NS,NS})$$

and "A" = "(A^T*I + I*A)" is a NP by NP coefficient matrix formed with the elements of A. The straightforward procedure for determining this matrix is to simply write down the scalar equations and place the coefficients of the elements of P in the proper positions. For purposes of implementation on a digital computer, a more systematic approach is desirable.

To develop this approach it is helpful to derive an expression which relates the position of an element, $(P)_{I,J}$, in the matrix form to its position $("P")_K$ in the vector form, that is

$$("P")_K \triangleq (P)_{I,J}$$

This transformation is given below and may be verified by inspection.

$$K = T(I, J) \tag{E-5}$$

where

$$T(I, J) = \begin{cases} (I-1)\ \mathrm{INT}(NS - \frac{I}{2}) + J & I \leq J \\ (J-1)\ \mathrm{INT}(NS - \frac{J}{2}) + I & I > J \end{cases}$$

and INT(M) indicates the truncation of M to an integer value.

Consider the I, J^{th} scalar equation of (E-2). The notation $(PA)_{I,J}$ refers to the I, J^{th} element of the matrix PA.

$$(A^TP)_{I,J} + (PA)_{I,J} = (D)_{I,J} \tag{E-6}$$

or

$$\sum_{KL=1}^{NS} (A)_{KL,I}\,(P)_{KL,J} + (P)_{I,KL}\,(A)_{KL,J} = (D)_{I,J} \tag{E-7}$$

This expression is required to be identical to the K^{th} component of the vector equation.

$$K = T(I, J)$$

$$(D)_{I,J} = ("D")_K = \sum_{KK=1}^{NP} ("A")_{K,KK} ("P")_{KK} = ("A" \; "P")_K \qquad \text{(E-8)}$$

Thus the elements of A in (E-7) will form the K^{th} row of "A". The column position of an element of A, $(A)_{KL,J}$ in the K^{th} row of "A" depends on the element of P which multiplies it, $(P)_{I,KL}$. Hence from the terms $(P)_{I,KL}$ $(A)_{KL,J}$, $(A)_{KL,J}$ is placed in the $K = T(I, J)$ row position and the $L = T(I, KL)$ column position of "A". To generate the remaining elements of "A", the lower triangular terms of $A^T P$ and PA are considered and the elements of A are allocated to the proper position in "A". It is possible that more than one element of A is placed in the same position of "A" and in that case that coefficient is equal to the sum of all such elements.

Since P is symmetric the implementation of this scheme on a digital computer may be simplified by considering only terms in PA. Instead of checking the lower triangular terms the lower and upper triangular terms are checked with the diagonal terms considered twice.

The matrix equation $G(P, A, H) = A^T P + PA - H$ where H and P are assumed to be skew symmetric, $(H = - H^T, P = - P^T)$, may be treated in a similar manner. In this case the equations corresponding to the diagonal positions of G are trivially satisfied because of the skew symmetry. Then the equivalent vector equation system consists of $NQ = \frac{NS(NS-1)}{2}$ equations corresponding to the lower or upper off-diagonal triangular terms. Thus

$$A^T P + PA = H$$

or

$$'A' \; 'P' = 'H'$$

where

$$'P'^T = (P_{2,1};\ldots;P_{NS,1};\ P_{3,2};\ldots;P_{NS,NS-1})$$

$$'H'^T = (H_{21};\ldots;H_{NS,1};\ H_{3,2};\ldots;H_{NS,NS-1})$$

and 'A' is obtained as "A" except that only off-diagonal terms in the products A^TP and PA are considered. In addition the skew symmetry requires that some of the elements of A are multiplied by -1 before being placed in the 'A' matrix.

To illustrate this procedure for a symmetric P and D, consider a second order example.

$$\begin{bmatrix} A_{11} & A_{21} \\ A_{12} & A_{22} \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} + \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix}$$

Since $D_{12} = D_{21}$ and $P_{12} = P_{21}$ this matrix equation may be written in terms of the following set of scalar equations corresponding to the lower triangular terms.

$$\left.\begin{aligned} & 2\,A_{11}\,P_{11} + 2\,A_{21}\,P_{21} = D_{11} \\ & A_{12}\,P_{11} + (A_{22} + A_{11})\,P_{21} + A_{21}\,P_{22} = D_{21} \\ & 2\,A_{12}\,P_{21} + 2\,A_{22}\,P_{22} = D_{22} \end{aligned}\right\} \qquad \text{(E-9)}$$

Then "A" "P" = "D"

$$\begin{bmatrix} 2\,A_{11} & 2\,A_{21} & 0 \\ A_{12} & A_{22}+A_{11} & A_{21} \\ 0 & 2\,A_{12} & 2A_{22} \end{bmatrix} \begin{bmatrix} P_{11} \\ P_{21} \\ P_{22} \end{bmatrix} = \begin{bmatrix} D_{11} \\ D_{21} \\ D_{22} \end{bmatrix} \qquad \text{(E-10)}$$

This same coefficient matrix can be obtained by considering the elements of PA.

$$PA = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

$$= \begin{bmatrix} P_{11} A_{11} + P_{12} A_{21} & P_{11} A_{12} + P_{12} A_{22} \\ P_{21} A_{11} + P_{22} A_{21} & P_{21} A_{12} + P_{22} A_{22} \end{bmatrix}$$

In particular,

$$(PA)_{21} = P_{21} A_{11} + P_{22} A_{21}$$

The elements A_{11} and A_{21} will be placed in the second row of "A" since

$$K = T(2, 1) = 2$$

This column position is determined by the multiplying P element.

$$P_{21} A_{11} \longrightarrow L = T(2, 1) = 2$$

$$P_{22} A_{21} \longrightarrow L = T(2, 2) = 3$$

Thus A_{11} is placed in 2, 2 position of "A" and A_{21} is placed in 2, 3 position of "A". Note that this placement agrees with (E-10).

This systematic procedure is easily programmed for use on a digital computer as indicated below. Note that this procedure is simpler than that of referece 46, since only simple "IF" rather than logical "IF" statements are required.

Given a matrix equation

$$A^T P + PA = D$$

where all matrices are NS by NS and D is symmetric the following code generates the coefficient matrix for the equivalent vector system of equations.

E "P" = "A" "P" = "D"

where $NP = \frac{NS(NS+1)}{2}$ and E is NP by NP.

```
      DO 20 K = 1, NP
      DO 20 L = 1, NP
   20 E(K, L) = 0.0
      DO 40 I = 1, NS
      DO 40 J = 1, NS
      DO 40 KL = 1, NS
C     L = T(J, KL)
      T1 = NS - 0.5*J
      L = (J-1)*T1 + KL
      IF(KL-J) 22, 24, 24
   22 T1 = NS - 0.5*KL
      L = (KL-1)*T1 + J
   24 CONTINUE
C     K = T(I, KL)
      T2 = NS - 0.5*I
      K = (I-1)*T2 + KL
      IF(KL-I) 26, 28, 28
   26 T2 = NS - 0.5*KL
      K = (KL-1)*T2 + I
C     DIAGONAL TERMS MUST BE CONSIDERED TWICE
   28 FAC = 1.0
      IF(I-KL) 32, 30, 32
   30 FAC = 2.0
   32 CONTINUE
      E(K, L) = E(k, L) + FAC*A(J, I)
   40 CONTINUE
```

Appendix F

Time Varying Model

An eleven state time varying model was used to evaluate the proposed control laws.

$$\dot{\underline{x}} = A\,\underline{x} + \underline{b}\,\beta_c + \underline{v}\,; \qquad \underline{x}(t_o) = \underline{c} \tag{F-1}$$

where

$$\underline{x} = \begin{bmatrix} \phi \\ \dot{\phi} \\ \alpha \\ \eta_1 \\ \dot{\eta}_1 \\ \eta_2 \\ \dot{\eta}_2 \\ \eta_3 \\ \dot{\eta}_3 \\ \beta \\ \dot{\beta} \end{bmatrix} \quad ; \quad \underline{y} = \begin{bmatrix} \phi_D \\ \dot{\phi}_R \end{bmatrix}$$

$$
A = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & A(2,3) & 0 & 0 & 0 & 0 & 0 & 0 & A(2,10) & 0 \\
A(3,7) & 1 & A(3,3) & 0 & 0 & 0 & 0 & 0 & 0 & A(3,10) & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & A(5,4) & A(5,5) & 0 & 0 & 0 & 0 & A(5,10) & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & A(7,6) & A(7,7) & 0 & 0 & A(7,10) & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & A(9,8) & A(9,9) & A(9,10) & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -50 & -10
\end{bmatrix}
$$

$$
\underline{b} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 50 \end{bmatrix} \quad ; \quad \underline{v} = \begin{bmatrix} 0 \\ 0 \\ \frac{\dot{v}}{v}\alpha_w + \dot{\alpha}_w \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
$$

$$c = \begin{bmatrix} 0 & 0 & 0 & c(1,4) & 0 & c(1,6) & 0 & c(1,8) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c(2,5) & 0 & c(2,7) & 0 & c(2,9) & 0 & 0 \end{bmatrix}$$

The following pages contain these parameter values at four second intervals for the duration of the trajectory. Linear interpolation was used to obtain the coefficient values for values of time not given in the table.

TIME	A(2,3)	A(3,1)	A(3,3)	A(2,10)	A(3,10)
0.00	-0.00000E-01	-3.89717E 01	-1.00000E 01	-8.72556E-01	-3.92183E 01
4.00	2.34665E-04	-9.51120E-01	-2.63210E-01	-8.74762E-01	-9.71986E-01
8.00	1.01622E-03	-4.53424E-01	-1.38907E-01	-8.78044E-01	-4.73305E-01
12.00	2.43095E-03	-2.88416E-01	-9.79386E-02	-8.82565E-01	-3.07714E-01
16.00	4.57627E-03	-2.06503E-01	-7.81349E-02	-8.87882E-01	-2.25471E-01
20.00	7.60375E-03	-1.57548E-01	-6.69383E-02	-8.94789E-01	-1.76404E-01
24.00	1.14311E-02	-1.24987E-01	-6.01712E-02	-9.02719E-01	-1.43915E-01
28.00	1.62591E-02	-1.01944E-01	-5.58516E-02	-9.11922E-01	-1.20837E-01
32.00	2.15078E-02	-8.46425E-02	-5.30970E-02	-9.22882E-01	-1.03582E-01
36.00	2.64925E-02	-7.11557E-02	-5.14360E-02	-9.35202E-01	-9.01960E-02
40.00	3.22049E-02	-6.02938E-02	-5.05306E-02	-9.49016E-01	-7.95108E-02
44.00	3.62459E-02	-5.14219E-02	-4.99302E-02	-9.64189E-01	-7.08068E-02
48.00	3.51612E-02	-4.40860E-02	-4.96738E-02	-9.81029E-01	-6.36027E-02
52.00	2.11236E-02	-3.78825E-02	-4.96559E-02	-9.99619E-01	-5.75869E-02
56.00	-1.38871E-02	-3.26468E-02	-5.00528E-02	-1.01913E 00	-5.25287E-02
60.00	-8.21618E-03	-2.79962E-02	-5.04673E-02	-1.03976E 00	-4.82414E-02
64.00	3.69351E-02	-2.43069E-02	-4.94450E-02	-1.06234E 00	-4.45763E-02
68.00	1.12482E-01	-2.10898E-02	-4.79278E-02	-1.08613E 00	-4.13469E-02
72.00	1.62098E-01	-1.82865E-02	-4.60360E-02	-1.11129E 00	-3.84289E-02
76.00	1.86803E-01	-1.57953E-02	-4.27376E-02	-1.13864E 00	-3.57815E-02
80.00	2.03047E-01	-1.36615E-02	-4.06825E-02	-1.16693E 00	-3.33820E-02
84.00	2.39399E-01	-1.18564E-02	-3.96747E-02	-1.19690E 00	-3.12185E-02
88.00	2.08929E-01	-1.02619E-02	-3.57209E-02	-1.22966E 00	-2.92647E-02
92.00	1.85940E-01	-8.86826E-03	-3.31805E-02	-1.26529E 00	-2.75001E-02
96.00	1.57643E-01	-7.70149E-03	-3.09094E-02	-1.30489E 00	-2.59155E-02
100.00	1.31395E-01	-6.22695E-03	-2.93635E-02	-1.34873E 00	-2.44987E-02
104.00	1.08228E-01	-5.52359E-03	-2.73510E-02	-1.40055E 00	-2.31645E-02
108.00	8.61555E-02	-5.01682E-03	-2.56583E-02	-1.45446E 00	-2.21193E-02
112.00	6.73236E-02	-4.78110E-03	-2.38922E-02	-1.52292E 00	-2.10694E-02
116.00	5.00602E-02	-4.68305E-03	-2.23819E-02	-1.59491E 00	-2.02576E-02
120.00	3.89616E-02	-4.04329E-03	-2.16080E-02	-1.68970E 00	-1.94422E-02
124.00	2.85525E-02	-3.60106E-03	-2.09329E-02	-1.79097E 00	-1.88293E-02
120.00	2.32690E-02	-2.85572E-03	-2.06856E-02	-1.92912E 00	-1.82079E-02
132.00	1.78008E-02	-2.33237E-03	-2.04568E-02	-2.08038E 00	-1.77667E-02
136.00	1.64635E-02	-2.27325E-03	-1.98447E-02	-2.29853E 00	-1.73135E-02
140.00	1.43081E-02	-2.35353E-03	-1.93036E-02	-2.54732E 00	-1.70297E-02

TIME	A(5,4)	A(5,5)	A(5,10)	C(1,4)	C(2,5)
0.00	-3.89433E 01	-1.24809E-01	1.99244E 02	1.50000E-02	7.00000E-03
4.00	-3.92419E 01	-1.25287E-01	2.00990E 02	1.50000E-02	7.00000E-03
8.00	-3.95337E 01	-1.25752E-01	2.02669E 02	1.50000E-02	7.00000E-03
12.00	-3.98190E 01	-1.26205E-01	2.04929E 02	1.50000E-02	7.00000E-03
16.00	-4.01126E 01	-1.26669E-01	2.07314E 02	1.50000E-02	7.00000E-03
20.00	-4.04555E 01	-1.27209E-01	2.08774E 02	1.50000E-02	7.00000E-03
24.00	-4.07920E 01	-1.27737E-01	2.11242E 02	1.50000E-02	7.00000E-03
28.00	-4.10170E 01	-1.28089E-01	2.13627E 02	1.50000E-02	7.00000E-03
32.00	-4.13153E 01	-1.28554E-01	2.16306E 02	1.50000E-02	7.00000E-03
36.00	-4.16228E 01	-1.29031E-01	2.18875E 02	1.50000E-02	7.00000E-03
40.00	-4.19477E 01	-1.29534E-01	2.21923E 02	1.50000E-02	7.00000E-03
44.00	-4.22249E 01	-1.29961E-01	2.24808E 02	1.50000E-02	7.00000E-03
48.00	-4.25358E 01	-1.30439E-01	2.27871E 02	1.50000E-02	7.00000E-03
52.00	-4.27984E 01	-1.30841E-01	2.30702E 02	1.50000E-02	7.00000E-03
56.00	-4.30866E 01	-1.31281E-01	2.34065E 02	1.50000E-02	7.00000E-03
60.00	-4.33510E 01	-1.31683E-01	2.37979E 02	1.50000E-02	7.00000E-03
64.00	-4.36078E 01	-1.32073E-01	2.41442E 02	1.50000E-02	7.00000E-03
68.00	-4.38572E 01	-1.32450E-01	2.44845E 02	1.50000E-02	7.00000E-03
72.00	-4.41072E 01	-1.32827E-01	2.47996E 02	1.50000E-02	7.00000E-03
76.00	-4.43580E 01	-1.33204E-01	2.51663E 02	1.50000E-02	7.00000E-03
80.00	-4.46681E 01	-1.33668E-01	2.54610E 02	1.50000E-02	7.00000E-03
84.00	-4.49289E 01	-1.34058E-01	2.58204E 02	1.50000E-02	7.00000E-03
88.00	-4.51988E 01	-1.34460E-01	2.61315E 02	1.50000E-02	7.00000E-03
92.00	-4.54526E 01	-1.34837E-01	2.65393E 02	1.50000E-02	7.00000E-03
96.00	-4.56902E 01	-1.35189E-01	2.69548E 02	1.50000E-02	7.00000E-03
100.00	-4.59368E 01	-1.35553E-01	2.73145E 02	1.50000E-02	7.00000E-03
104.00	-4.62183E 01	-1.35968E-01	2.78995E 02	1.50000E-02	7.00000E-03
108.00	-4.64921E 01	-1.36370E-01	2.84057E 02	1.50000E-02	7.00000E-03
112.00	-4.68182E 01	-1.36848E-01	2.91524E 02	1.50000E-02	7.00000E-03
116.00	-4.71542E 01	-1.37338E-01	2.99606E 02	1.50000E-02	7.00000E-03
120.00	-4.74827E 01	-1.37815E-01	3.10272E 02	1.50000E-02	7.00000E-03
124.00	-4.79427E 01	-1.38481E-01	3.19662E 02	1.50000E-02	7.00000E-03
128.00	-4.83701E 01	-1.39097E-01	3.42572E 02	1.50000E-02	7.00000E-03
132.00	-4.89926E 01	-1.39989E-01	3.74430E 02	1.50000E-02	7.00000E-03
136.00	-4.97874E 01	-1.41120E-01	4.02145E 02	1.50000E-02	7.00000E-03
140.00	-5.07675E 01	-1.42503E-01	4.34076E 02	1.50000E-02	7.00000E-03

TIME	A(7,6)	A(7,7)	A(7,10)	C(1,6)	C(2,7)
0.00	-1.22483E 02	-2.21344E-01	-3.23933E 02	4.69000E-03	-3.36000E-03
4.00	-1.23207E 02	-2.21997E-01	-3.27875E 02	4.61000E-03	-3.42000E-03
8.00	-1.23948E 02	-2.22664E-01	-3.32040E 02	4.53000E-03	-3.48000E-03
12.00	-1.24760E 02	-2.23392E-01	-3.36842E 02	4.43000E-03	-3.54800E-03
16.00	-1.25674E 02	-2.24209E-01	-3.42340E 02	4.31000E-03	-3.62400E-03
20.00	-1.26578E 02	-2.25013E-01	-3.48188E 02	4.19000E-03	-3.70000E-03
24.00	-1.27683E 02	-2.25994E-01	-3.54858E 02	4.04600E-03	-3.78400E-03
28.00	-1.28793E 02	-2.26974E-01	-3.61956E 02	3.90200E-03	-3.86800E-03
32.00	-1.29993E 02	-2.28029E-01	-3.69626E 02	3.74600E-03	-3.95600E-03
36.00	-1.31301E 02	-2.29173E-01	-3.77912E 02	3.57800E-03	-4.04800E-03
40.00	-1.32614E 02	-2.30316E-01	-3.86685E 02	3.41000E-03	-4.14000E-03
44.00	-1.34109E 02	-2.31611E-01	-3.95892E 02	3.21400E-03	-4.24000E-03
48.00	-1.35612E 02	-2.32905E-01	-4.05584E 02	3.01800E-03	-4.34000E-03
52.00	-1.37212E 02	-2.34275E-01	-4.15422E 02	2.81600E-03	-4.44000E-03
56.00	-1.38910E 02	-2.35720E-01	-4.25364E 02	2.60800E-03	-4.54000E-03
60.00	-1.40618E 02	-2.37165E-01	-4.35646E 02	2.40000E-03	-4.64000E-03
64.00	-1.42442E 02	-2.38698E-01	-4.44946E 02	2.17600E-03	-4.74400E-03
68.00	-1.44293E 02	-2.40244E-01	-4.54430E 02	1.95200E-03	-4.84800E-03
72.00	-1.46171E 02	-2.41802E-01	-4.63115E 02	1.72600E-03	-4.94600E-03
76.00	-1.48107E 02	-2.43398E-01	-4.70877E 02	1.49800E-03	-5.03800E-03
80.00	-1.50055E 02	-2.44994E-01	-4.78571E 02	1.27000E-03	-5.13000E-03
84.00	-1.51985E 02	-2.46565E-01	-4.83702E 02	1.04600E-03	-5.21800E-03
88.00	-1.53928E 02	-2.48136E-01	-4.88564E 02	8.22000E-04	-5.30600E-03
92.00	-1.55820E 02	-2.49656E-01	-4.91951E 02	6.04000E-04	-5.38400E-03
96.00	-1.57693E 02	-2.51152E-01	-4.93844E 02	3.92000E-04	-5.45200E-03
100.00	-1.59560E 02	-2.52634E-01	-4.95491E 02	1.80000E-04	-5.52000E-03
104.00	-1.61263E 02	-2.53979E-01	-4.95429E 02	-2.00000E-05	-5.57200E-03
108.00	-1.62975E 02	-2.55324E-01	-4.95336E 02	-2.20000E-04	-5.62400E-03
112.00	-1.64583E 02	-2.56580E-01	-4.95516E 02	-4.10000E-04	-5.67600E-03
116.00	-1.66119E 02	-2.57774E-01	-4.96251E 02	-5.90000E-04	-5.72800E-03
120.00	-1.67661E 02	-2.58968E-01	-4.96867E 02	-7.70000E-04	-5.78000E-03
124.00	-1.69047E 02	-2.60036E-01	-5.02431E 02	-9.13000E-04	-5.81200E-03
128.00	-1.70438E 02	-2.61104E-01	-5.08067E 02	-1.06600E-03	-5.84400E-03
132.00	-1.71869E 02	-2.62197E-01	-5.21646E 02	-1.20800E-03	-5.86800E-03
136.00	-1.73321E 02	-2.63303E-01	-5.43285E 02	-1.34400E-03	-5.88400E-03
140.00	-1.74780E 02	-2.64409E-01	-5.65285E 02	-1.48000E-03	-5.90000E-03

TIME	A(9,8)	A(9,9)	A(9,10)	C(1,8)	C(2,9)
0.00	-2.57615E 02	-3.21008E-01	3.16400E 02	-8.75000E-03	-4.52000E-03
4.00	-2.57736E 02	-3.21083E-01	3.17442E 02	-8.74200E-03	-4.51600E-03
8.00	-2.57837E 02	-3.21146E-01	3.18714E 02	-8.73400E-03	-4.51200E-03
12.00	-2.58019E 02	-3.21259E-01	3.21105E 02	-8.73800E-03	-4.50400E-03
16.00	-2.58281E 02	-3.21422E-01	3.24621E 02	-8.75400E-03	-4.49200E-03
20.00	-2.58523E 02	-3.21573E-01	3.28407E 02	-8.77000E-03	-4.48000E-03
24.00	-2.59009E 02	-3.21875E-01	3.34164E 02	-8.80200E-03	-4.45600E-03
28.00	-2.59494E 02	-3.22177E-01	3.40201E 02	-8.83400E-03	-4.43200E-03
32.00	-2.60143E 02	-3.22579E-01	3.47436E 02	-8.88000E-03	-4.39800E-03
36.00	-2.60974E 02	-3.23094E-01	3.55841E 02	-8.94000E-03	-4.35400E-03
40.00	-2.61807E 02	-3.23609E-01	3.64488E 02	-9.00000E-03	-4.31000E-03
44.00	-2.63089E 02	-3.24401E-01	3.75254E 02	-9.08800E-03	-4.24200E-03
48.00	-2.64375E 02	-3.25192E-01	3.86157E 02	-9.17600E-03	-4.17400E-03
52.00	-2.65991E 02	-3.26185E-01	3.98055E 02	-9.28400E-03	-4.08800E-03
56.00	-2.67922E 02	-3.27366E-01	4.10859E 02	-9.41200E-03	-3.98400E-03
60.00	-2.69859E 02	-3.28548E-01	4.23577E 02	-9.54000E-03	-3.88000E-03
64.00	-2.72611E 02	-3.30219E-01	4.37679E 02	-9.70800E-03	-3.72400E-03
68.00	-2.75378E 02	-3.31890E-01	4.51507E 02	-9.87600E-03	-3.56800E-03
72.00	-2.78682E 02	-3.33876E-01	4.65440E 02	-1.00700E-02	-3.38200E-03
76.00	-2.82492E 02	-3.36150E-01	4.79386E 02	-1.02900E-02	-3.16600E-03
80.00	-2.86328E 02	-3.38425E-01	4.92799E 02	-1.05100E-02	-2.95000E-03
84.00	-2.91347E 02	-3.41378E-01	5.04290E 02	-1.07780E-02	-2.62600E-03
88.00	-2.96410E 02	-3.44331E-01	5.15275E 02	-1.10460E-02	-2.30200E-03
92.00	-3.02061E 02	-3.47598E-01	5.23000E 02	-1.13300E-02	-1.92000E-03
96.00	-3.08273E 02	-3.51155E-01	5.27618E 02	-1.16300E-02	-1.48000E-03
100.00	-3.14549E 02	-3.54711E-01	5.31960E 02	-1.19300E-02	-1.04000E-03
104.00	-3.21248E 02	-3.58468E-01	5.24521E 02	-1.22300E-02	-5.24000E-04
108.00	-3.28018E 02	-3.62225E-01	5.16982E 02	-1.25300E-02	-8.00110E-06
112.00	-3.34467E 02	-3.65769E-01	5.02933E 02	-1.27920E-02	4.90000E-04
116.00	-3.40632E 02	-3.69124E-01	4.82598E 02	-1.30160E-02	9.70000E-04
120.00	-3.46853E 02	-3.72480E-01	4.61991E 02	-1.32400E-02	1.45000E-03
124.00	-3.51549E 02	-3.74993E-01	4.37139E 02	-1.33960E-02	1.83800E-03
128.00	-3.56301E 02	-3.77519E-01	4.12472E 02	-1.35520E-02	2.22600E-03
132.00	-3.60321E 02	-3.79642E-01	3.91770E 02	-1.36780E-02	3.01000E-03
136.00	-3.63596E 02	-3.81364E-01	3.75070E 02	-1.37740E-02	4.19000E-03
140.00	-3.66886E 02	-3.83086E-01	3.58701E 02	-1.38700E-02	5.37000E-03

TIME (SEC)	MASS (KG)	XCG (M)	XCP (M)	LCP (M)	IXX (KG-M)
0.00	2.76205E 06	2.73900E 01	3.62103E 01	-8.82031E 00	8.50078E 08
4.00	2.70854E 06	2.73900E 01	3.61097E 01	-8.71973E 00	8.48227E 08
8.00	2.65511E 06	2.74000E 01	3.60091E 01	-8.60915E 00	8.46271E 08
12.00	2.60150E 06	2.74200E 01	3.58080E 01	-8.38797E 00	8.44119E 08
16.00	2.54799E 06	2.74400E 01	3.56068E 01	-8.16679E 00	8.41937E 08
20.00	2.49447E 06	2.74800E 01	3.55062E 01	-8.02621E 00	8.39623E 08
24.00	2.44095E 06	2.75200E 01	3.53051E 01	-7.78505E 00	8.37146E 08
28.00	2.38743E 06	2.75700E 01	3.51039E 01	-7.53387E 00	8.34599E 08
32.00	2.33392E 06	2.76400E 01	3.48021E 01	-7.16212E 00	8.31848E 08
36.00	2.28040E 06	2.77200E 01	3.42992E 01	-6.57921E 00	8.28952E 08
40.00	2.22688E 06	2.78100E 01	3.38969E 01	-6.08687E 00	8.25766E 08
44.00	2.17337E 06	2.79100E 01	3.32934E 01	-5.38336E 00	8.22359E 08
48.00	2.11985E 06	2.80300E 01	3.21869E 01	-4.15694E 00	8.18694E 08
52.00	2.06633E 06	2.81700E 01	3.01753E 01	-2.00526E 00	8.14624E 08
56.00	2.01282E 06	2.83200E 01	2.72583E 01	1.06169E 00	8.10440E 08
60.00	1.95930E 06	2.84800E 01	2.79624E 01	5.17593E-01	8.05866E 08
64.00	1.90578E 06	2.86700E 01	3.07788E 01	-2.10876E 00	8.00737E 08
68.00	1.85227E 06	2.88800E 01	3.50033E 01	-6.12329E 00	7.95254E 08
72.00	1.79875E 06	2.91100E 01	3.78197E 01	-8.70966E 00	7.89244E 08
76.00	1.74523E 06	2.93800E 01	4.02337E 01	-1.08537E 01	7.82626E 08
80.00	1.69172E 06	2.96600E 01	4.16419E 01	-1.19819E 01	7.75421E 08
84.00	1.63820E 06	2.99700E 01	4.33518E 01	-1.33818E 01	7.67645E 08
88.00	1.58468E 06	3.03200E 01	4.45588E 01	-1.42388E 01	7.58893E 08
92.00	1.53117E 06	3.07100E 01	4.51623E 01	-1.44523E 01	7.49251E 08
96.00	1.47765E 06	3.11500E 01	4.52629E 01	-1.41129E 01	7.38585E 08
100.00	1.42413E 06	3.16300E 01	4.53635E 01	-1.37335E 01	7.26793E 08
104.00	1.37061E 06	3.21900E 01	4.55143E 01	-1.33243E 01	7.13040E 08
108.00	1.31710E 06	3.27500E 01	4.56652E 01	-1.29152E 01	6.99287E 08
112.00	1.26358E 06	3.34400E 01	4.56149E 01	-1.21749E 01	6.82292E 08
116.00	1.21006E 06	3.41300E 01	4.55646E 01	-1.14346E 01	6.65296E 08
120.00	1.15655E 06	3.49950E 01	4.57155E 01	-1.07205E 01	6.44063E 08
124.00	1.10303E 06	3.58600E 01	4.58664E 01	-1.00064E 01	6.22830E 08
128.00	1.04951E 06	3.69500E 01	4.66208E 01	-9.67078E 00	5.95878E 08
132.00	9.95996E 05	3.80400E 01	4.73752E 01	-9.33516E 00	5.68926E 08
136.00	9.42480E 05	3.94400E 01	4.91354E 01	-9.69537E 00	5.33912E 08
140.00	8.88963E 05	4.08400E 01	5.08956E 01	-1.00556E 01	4.98897E 08

TIME (SEC)	THRUST (N)	DRAG (N)	VELOCITY (M/SEC)	Q (N/M)	CNALPHA
0.00	3.38509E 07	3.53040E 04	0.00000E-01	0.00000E-01	4.60000E 00
4.00	3.38626E 07	4.09890E 04	1.02900E 01	6.25000E 01	4.60000E 00
8.00	3.38989E 07	5.97750E 04	2.15800E 01	2.73500E 02	4.6C000E 00
12.00	3.39620E 07	9.35160E 04	3.39400E 01	6.69800E 02	4.60000E 00
16.00	3.40534E 07	1.33627E 05	4.74200E 01	1.28890E 03	4.61000E 00
20.00	3.41742E 07	1.78357E 05	6.21300E 01	2.16840E 03	4.62000E 00
24.00	3.43254E 07	2.24660E 05	7.81700E 01	3.34370E 03	4.63000E 00
28.00	3.45071E 07	2.59467E 05	9.56900E 01	4.84720E 03	4.68000E 00
32.00	3.47186E 07	2.82588E 05	1.14890E 02	6.70820E 03	4.69000E 0C
36.00	3.49584E 07	2.84786E 05	1.35970E 02	8.94460E 03	4.70000E 00
40.00	3.52241E 07	3.16057E 05	1.59150E 02	1.15600E 04	4.76000E 00
44.00	3.55119E 07	3.83726E 05	1.84610E 02	1.45280E 04	4.80000E 00
48.00	3.58171E 07	5.02170E 05	2.12520E 02	1.77990E 04	4.90000E 00
52.00	3.61339E 07	7.73218E 05	2.42930E 02	2.12750E 04	5.08000E 00
56.00	3.64560E 07	1.14050E 06	2.75840E 02	2.48160E 04	5.38000E 00
60.00	3.67763E 07	1.71583E 06	3.11270E 02	2.82650E 04	5.70000E 00
64.00	3.70682E 07	2.02296E 06	3.49260E 02	3.14300E 04	5.62000E 00
68.00	3.73853E 07	2.06992E 06	3.90520E 02	3.41980E 04	5.38000E 00
72.00	3.76622E 07	1.91976E 06	4.35880E 02	3.64170E 04	5.08000E 00
76.00	3.79139E 07	1.76683E 06	4.85710E 02	3.76990E 04	4.50000E 00
80.00	3.81346E 07	1.61020E 06	5.40220E 02	3.76130E 04	4.40000E 00
84.00	3.83214E 07	1.44991E 06	5.99450E 02	3.58840E 04	4.82000E 00
88.00	3.84723E 07	1.24739E 06	6.63670E 02	3.26150E 04	4.30000E 00
92.00	3.85877E 07	1.02566E 06	7.33130E 02	2.83000E 04	4.29000E 00
96.00	3.86746E 07	8.23414E 05	8.07950E 02	2.42770E 04	4.28000E 00
100.00	3.87589E 07	6.70182E 05	8.88270E 02	2.05580E 04	4.26000E 00
104.00	3.87795E 07	5.43185E 05	9.77135E 02	1.74090E 04	4.19000E 00
108.00	3.88201E 07	4.16187E 05	1.06600E 03	1.42600E 04	4.12000E 00
112.00	3.88411E 07	3.26879E 05	1.16715E 03	1.17617E 04	4.04000E 00
116.00	3.88621E 07	2.37570E 05	1.26830E 03	9.26339E 03	3.96000E 00
120.00	3.88724E 07	1.63636E 05	1.38300E 03	7.50129E 03	3.93000E 00
124.00	3.88828E 07	1.29702E 05	1.49770E 03	5.73920E 03	3.90000E 00
128.00	3.88877E 07	9.87330E 04	1.62800E 03	4.59475E 03	3.93000E 00
132.00	3.88927E 07	6.77640E 04	1.75830E 03	3.45030E 03	3.96000E 00
136.00	3.88950E 07	5.08205E 04	1.90690E 03	2.76475E 03	4.13000E 00
140.00	3.88973E 07	3.38770E 04	2.05550E 03	2.07920E 03	4.30000E 00