MASSACHUSETTS INSTITUTE OF TECHNOLOGY

TE-26

A NEW MODEL PERFORMANCE INDEX FOR THE ENGINEERING DESIGN OF CONTROL SYSTEMS

by
Herman A. Rediess

December 9, 1968

EXPERIMENTAL ASTRONOMY LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE 39, MASSACHUSETTS

TE-26

A NEW MODEL PERFORMANCE INDEX FOR THE ENGINEERING DESIGN OF CONTROL SYSTEMS

by
Herman A. Rediess

December 9, 1968

Degree of Doctor of Philosophy

TE-26

A NEW MODEL PERFORMANCE INDEX FOR THE ENGINEERING DESIGN OF CONTROL SYSTEMS

by

Herman A. Rediess

S. B., University of California at Berkeley
(1959)

S. M., University of Southern California
(1964)

Submitted in Partial Fulfillment
of the Requirements for the
Degree of Doctor of Philosophy
at the
Massachusetts Institute of Technology
February 1969

Signature of Author: Herman A. Rediess
Department of Aeronautics and Astronautics
December 9, 1968

Certified by: H. Philip Whitaker
Thesis Advisor and Chairman, Thesis Committee

Thesis Advisor

James E. Potter
Thesis Advisor

Accepted by:
Chairman, Departmental Graduate Committee

This thesis was prepared under DSR Project 70343, sponsored by the National Aeronautics and Space Administration, Electronics Research Center, Cambridge, Massachusetts, through NASA Grant Number NGR 22-009-229.

A NEW MODEL PERFORMANCE INDEX
FOR THE
ENGINEERING DESIGN OF CONTROL SYSTEMS

by

Herman A. Rediess

Submitted to the Department of Aeronautics and Astronautics on December 9, 1968, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

ABSTRACT

The theory and application of a new performance index, the Model PI, that brings engineering design specifications into the analytical design process is developed. A parameter optimization design procedure is established that starts with practical engineering specifications and uses the Model PI as a synthesis tool to obtain a satisfactory design. Although the techniques apply to linear, time invariant, deterministic control systems in general, the thesis is developed in the context of flight control systems in order to emphasize the relationship of realistic design requirements to the synthesis process. The Model PI represents a new criterion for approximating one dynamical system by another, based on a novel geometrical representation of linear autonomous systems. It is shown to be an effective performance index in designing practical systems and is shown to be substantially more efficient to use than a comparable model-referenced integral squared error performance index. A general digital computer program for control system design using the Model PI is developed. Its usefulness is demonstrated by three practical flight control system design examples.

Some interesting developments in linear optimal control resulting from the Model PI theory are presented. The Model PI is shown to provide a means of interpreting the state vector weighting matrix in terms of a model which the optimal system will approach in a limiting case. An interestingly simple solution of the linear optimal control synthesis procedure using one root square locus is presented.

Thesis Supervisor: H. Philip Whitaker, S. M.
Title: Professor of Aeronautics and Astronautics

Thesis Supervisor: Yao Tzu Li, Sc. D.
Title: Professor of Aeronautics and Astronautics

Thesis Supervisor: James E. Potter, Ph. D
Title: Associate Professor of Aeronautics and Astronautics

ACKNOWLEDGEMENT

The author wishes to thank the members of his doctoral thesis committee, Professor H. P. Whitaker, Y. T. Li, and J. E. Potter, for their helpful discussions, comments, and constructive criticisms during the course of this thesis effort. In addition, a special thanks is extended to Professor Whitaker for his continual interest and encouragement, as well as his many hours of effort spent in reviewing the early drafts of this document.

The M. I. T. Instrumentation Laboratory under the direction of Dr. C. Stark Draper and the M. I. T. Experimental Astronomy Laboratory under the direction of Professor W. R. Markey provided the environment and support for the research. Thanks go to Ralph Trueblood, Dr. William Bryant, and Marvin Todd of the M. I. T. Instrumentation Laboratory TAC Group for their helpful discussions on VTOL flight control systems and to the TAC Group as a whole for their general support. The author is indebted to the M. I. T. Experimental Astronomy Laboratory, Professor Markey, and Kenneth Britting, for their direct support of this research.

The computational work of this thesis was done at the Information Processing Services Center at M. I. T., Cambridge, Massachusetts. The excellent illustrations of the three-dimensional trajectories in Chapter 3 are the work of Dustin Thomas of the M. I. T. Instrumentation Laboratory, Technical Publications Group.

The author would also like to express his gratitude to the National Aeronautics and Space Administration, in particular the Flight Research Center, Edwards, California, under the direction of Paul F. Bikle, for providing the Graduate Study Leave that made the initial commitment here possible.

The author's deepest gratitude is given to his wife, Sharon, and children, Sharilyn and Nicholas, for their patience and understanding throughout this effort. A further thanks is due his wife for doing tedious data reduction, typing all the rough drafts and proof reading of this manuscript and finally the excellent job of typing the final manuscript, which was a demanding and arduous task. The production of this document was truly a husband-wife team effort.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

TABLE OF CONTENTS (cont)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (cont)

LIST OF ILLUSTRATIONS (cont)

LIST OF ILLUSTRATIONS (cont)

LIST OF TABLES

LIST OF PRINCIPAL NOMENCLATURE

Symbol	Definition
A	sensitivity matrix of $\underline{a}$ with respect to $\underline{p}$
B	sensitivity matrix of $\underline{x}_0$ with respect to $\underline{p}$
D(s)	characteristic polynomial of an open-loop system
F, F_s	system coefficient matrix
F_m	model coefficient matrix
H	a projection matrix (Ch. 3), output matrix (Ch. 7)
$\mathcal{H}$	Hamiltonian
I	identity matrix, or performance index
J	a quadratic cost functional
M	transformation matrix from $\underline{x}$ to $\underline{\tilde{x}}$
N(s)	transfer function numerator polynomial
O	null matrix
P	Lagrange multiplier matrix (Ch. 4) or solution to a matrix algebraic equation (Ch. 7)
PI	Model Performance Index
Q	general weighting matrix in cost functional
$\tilde{Q}$	Model PI weighting matrix $\dfrac{\underline{\tilde{\alpha}}\,\underline{\tilde{\alpha}}'}{\Vert \underline{\tilde{\alpha}} \Vert^2}$
W	pseudo initial condition weighting matrix
$\tilde{W}$	extended pseudo initial condition weighting matrix
X	integral of $\underline{x}\,\underline{x}'$ from 0 to ∞
$\underline{a}$	system coefficient vector
$\underline{\tilde{a}}$	system extended coefficient vector

Symbol	Definition
a_i	system characteristic polynomial coefficient of s^i
b_i	system transfer function numerator polynomial coefficient of s^i
c	a constant scalar
e	error between system and model output or state variable
$\underline{g}$	control coefficient vector (Ch. 7)
k	free design parameter, or number of model zeros
ℓ	order of model
m	number of system zeros
n	order of system
$\underline{0}$	null vector
$\underline{p}$	free design parameter vector
q_{ij}	general element of Q
r	pseudo initial condition weighting factor or control effort weighting factor (Ch. 7)
s	Laplace transform variable
t	time
u	input variable or control variable (Ch. 7)
x	system transient response variable
$\underline{x}$	system state (phase) vector
$\underline{\tilde{x}}$	system extended state vector or trajectory
$\underline{\hat{x}}$	projection of system trajectory into model's extended state space
y	system output variable
$\underline{y}$	system output vector

LIST OF PRINCIPAL NOMENCLATURE (cont)

Symbol	Definition
$\underline{\alpha}$	model coefficient vector
$\underline{\hat{\alpha}}$	model extended coefficient vector
$\underline{\tilde{\alpha}}$	model extended coefficient vector in systems extended state space
α_i	model characteristic polynomial coefficient of s^i
β_i	model transfer function numerator polynomial coefficient of s^i
δ	variational symbol (with another symbol)
ϵ_n	(n×1) vector with a one in the nth row, the rest zeros
$\underline{\lambda}$	Lagrange multiplier vector, costate vector
τ	time constant

Special symbols used in examples are defined locally.

GENERAL SUBSCRIPT NOTATION

d	diagonal matrix
m	model
0	initial time
s	system

SPECIAL NOTATION

Vectors are denoted by lower case letters with an underline.
Matrices are upper case letters.

SPECIAL NOTATION (cont)

A dot over a variable means the time derivation of that variable.

$(\ )^{(n)}$ means the nth derivative of the variable ( ).

( )' means the transpose of ( ).

$(\underset{\sim}{\ })$ means a vector in the systems extended state space $(n+1) \times 1$.

$(\hat{\underline{\ }})$ means a vector in the model's extended state space $(\ell+1) \times 1$.

$\|\underline{v}\|$ means $\sqrt{\underline{v}'\underline{v}}$ which is the length of $\underline{v}$.

$\|\underline{v}\|^2_M$ means $\underline{v}'M\underline{v}$.

CHAPTER 1

INTRODUCTION

The reliance on automatic flight control systems in aircraft has continued to increase since that first autopilot developed by Dr. E. A. Sperry only eleven years after the historic flight of the Kitty Hawk (1). The advances in autopilot technology were rather slow up to the Second World War. By the end of World War II sophisticated autopilots were available that adequately met the requirements of the operational aircraft at that time.

The development of high-performance jet aircraft brought a whole new dimension to flight control system requirements. Up to that time autopilots were designed mainly for attitude and heading control. The desired stability and control characteristics had always been designed into the basic airframe. It was possible to provide such heavy basic damping that the classic textbook by Perkins and Hage (2) states that the short period modes are of very little consequence in the flying qualities of airplanes. They could be designed almost entirely on their static and long period characteristics. With the advent of jet transports flying at high subsonic speeds and jet fighters flying at subsonic, transonic and supersonic speeds, the "short period" pitch dynamics and the "Dutch roll" dynamics became dominant factors in establishing adequate aircraft handling qualities. The aerodynamicists could no longer design sufficient damping into the basic airframe if they were to meet their primary objective of providing high aerodynamic efficiency at the cruise flight conditions. Consequently, most high-performance aircraft had marginal to unacceptable handling qualities in some portion of their flight envelope without a stability augmentation system.

A simple, fixed-gain pitch or yaw damper may be sufficient to provide satisfactory handling qualities in many situations. But, for several reasons, stability augmentation and control systems have necessarily become more complex. Mission requirements have become more severe, such as a re-entry from space (3). The vehicle response modes for some configurations like lifting bodies have become highly coupled (4). In some cases the pilot control task has become extremely difficult, as in landing a VTOL at steep approach angles under IFR (5). For some VTOL aircraft, going from hover through transition to cruise flight requires a complete change in the type of control commands (6, 7). And certainly many other examples could be added to this list.

In addition to providing satisfactory dynamics for a pilot to maneuver the high-performance aircraft proficiently, the requirements for completely automatic flight control under various situations have increased. The desire to operate commercial airports under all-weather conditions has motivated the development of all-weather automatic approach and landing systems (8). The flight control system necessary to couple with a landing aid presents a complex multivariable design problem. There are similar design problems with automatic carrier landing systems and automatic fire control systems.

The design process for flight control systems has become correspondingly more complex, lengthy and laborious by the conventional linear servo theory techniques (9, 10). Even with the aid of digital computers for root locus and Bode diagram computations and automatic root locus plotters, these techniques are rather slow processes when synthesizing complex multiloop and multiple input/output systems. This is mainly due to their "trial-and-error" nature and treating only one of several design parameters at a time. The iterations between the effects of several parameters and between inner and outer loop closures can become quite tedious. This is not to say that an experienced designer can not obtain excellent results with conventional techniques. The many complex and yet extremely effective flight control systems currently in operational use attest to the fact that they can. On the other hand it is easy to understand why control system engineers desire more systematic design techniques and thus have been pursuing various analytical design theories.

The primary objective of this dissertation is to develop an analytical design technique based on a new "Model Performance Index" (Model PI) that is applicable to linear flight control systems. It is recognized that the goal of most real design problems is to produce the simplest design that meets the performance specifications within acceptable tolerances. Therefore the analytical design process must start not with a mathematical performance index but with the general design requirements. The Model PI is used to bridge the gap between the engineering specifications and the purely mathematical optimization process. In addition, the Model PI provides a new interpretation of quadratic cost functionals used in optimal control theory that offers a physical basis for selection of the state vector weighting matrix.

1.1 Historical Background of Analytical Design Methods

Contemporary control system design theory is placing an increasing emphasis on optimization of mathematical functionals, sometimes called performance indices, that give a measure of the system's performance relative to some reference. This concept must be traced back ultimately to the independent, concurrent works of Wiener (11) in the United States and Kolmogoroff (12) in the U. S. S. R. which occurred in the late 1930's and early 1940's. These were the first formulations of filter design as an optimization problem. The Wiener-Kolmogoroff problem is referred to as the "free-configuration" problem in that it selects the one filter from the class of all possible linear filters that minimizes the mean squared error between the actual and desired signals. Subsequently, Hall (13) and Phillips (14) formulated the "fixed-configuration" or parameter optimization problem using a mean squared error performance index. In this approach the designer selects the filter or control system configuration leaving characteristic frequencies, time constants and gains unspecified, to be determined by the optimization process. Phillips derived a procedure for evaluating the mean squared error over an infinite time interval as an explicit, nonlinear function of the free design parameters. The design process is then to select the free parameter values that correspond to a minimum point of this function.

A problem encountered in using the Hall-Phillips method was that the resulting design could allow excessive signal magnitudes within the system that may exceed the range of assumed linearity or may even saturate. Newton, Gould and Kaiser (NGK) (15) proposed constraining any signal magnitude by adjoining the mean squared value of that signal to the original performance index by a Lagrange multiplier. The augmented performance index can be evaluated as a function of the free design parameters and the Lagrange multiplier using tabulated integrals. The design selected is the one in which the values of the free parameters minimize this augmented performance index while requiring the signal magnitude constraint to be satisfied.

The servomechanism design problem was formulated by Phillips and NGK in terms of the error between the desired and actual responses. The performance index penalizes large deviations of the actual response from the desired, so that minimizing it tends to force the system's response to be similar to the desired response. Thus the performance index is a measure of the performance relative to some desired response that the designer would specify for each problem. This type is sometimes called a model-referenced performance index. An alternate philosophy was adopted by several other researchers in which the performance index was to represent an absolute criterion in itself. Numerous performance indices were proposed to supposedly represent optimum transient response for a step input. Several proposed were the integral squared error (ISE), integral of the absolute error (IAE), integral of time-multiplied absolute error (ITAE), and integral of time-multiplied squared-error

(ITSE) where in each case the error refers to the difference between the instantaneous output and the steady state value for a step input. These and other criteria were compared and evaluated by Graham and Lathrop (16) with the conclusion that ITAE was clearly superior.

This may well be true for many servomechanisms, although there is some disagreement on that point (17). But in many applications defining a performance index relative to some model provides a necessary flexibility that is missing in all of the so called optimum transient response criteria.

At this point, Aizerman (18) proposed a new concept for representing the desired system response within a performance index. Rather than using the squared error between the desired and actual responses, Aizerman used a linear combination of the squares of the actual transient response and its derivatives. The relative weighting of these squared variables in the performance index was chosen so that the absolute minimum value of the performance index would correspond to a system design with a transient response identical to the desired response. In general, the absolute minimum value can only be obtained if one has complete freedom in selecting the closed-loop system design. Since the feedback configuration and design parameter values are usually constrained due to practical requirements, it is not usually possible to obtain the desired response identically. However, minimizing Aizerman's performance index would tend to force the system's response to be similar to the desired response, at least for a certain class of systems and types of desired response. The objective is the same as for a model-referenced performance index, but the relative effectiveness of these two types in synthesizing control systems has not been assessed to the author's knowledge. Aizerman's concept could provide a significant computational advantage in the optimization process over model-referenced performance indices in that the model's response never enters the computational problem. This point has not been verified or even suggested in the literature thus far. Although rather limited in application, Aizerman's performance index represented a distinctly different philosophy in analytical design.

Rekasius (19) recognized the potential of Aizerman's concept but noticed that it seriously restricted the type of models that could be used for the desired response. He proposed a somewhat improved performance index that lifted some of the restrictions but not others. Several serious limitations remained that restricted its application to essentially academic examples which possibly accounts for the apparent lack of wide spread knowledge of these performance indices.

Once a performance index is selected and a means for evaluating it established, the problem of minimizing it with respect to the free design parameters remains. The parameter optimization process is generally not easy and may be a formidable task. It is seldom possible to obtain a direct analytical solution except for simple academic examples. Several numerical optimization techniques suitable for digital computer application were reviewed by Spang (20) and Paiewonsky (21). An attractive feature of digital computation is that general algorithms can be written that apply to a wide variety of design problems. However, computational efficiency becomes an important factor because the computational times required for designing high order systems with several parameters can be excessive. Analog computer mechanizations of essentially gradient techniques have been used successfully by Roberts (22), Bingulac and Koktovic (23), Whitaker and Potter (24), and Whitaker, et. al. (4). The latter two works are specific applications to flight control system design problems.

In parallel with these developments in fixed-configuration design techniques was the rapidly developing theory of optimal control. The background and status of this popular new field were recently reviewed by Paiewonsky (21) and Athans (25). Of particular importance to flight control system synthesis is the special class of problems known as linear optimal control (26). It has been applied to several flight control system design problems (27 - 32). The performance index is taken to be some quadratic functional of the system's state vector and the control effort. This forces the solution to produce a linear feedback control law. Unfortunately there has not been any direct procedure for selecting the weighting matrices used in the performance index so they have been used as arbitrary constants that are adjusted until a satisfactory design is obtained. The resulting design is quite often more complex than necessary to satisfy the design specifications. Also it assumes availability of all the system state variables for feedback signals, which is generally not true in practice. Therefore a simplified approximation is usually sought using the theoretical optimal solution as a guide.

Modern control theory has thus produced several new design tools based on minimizing a variety of performance indices. The most effective performance indices in the synthesis of linear flight control systems are quadratic functionals that include, in some way, a model of the desired closed loop response. Significant contributions can be made in the areas of relating performance indices to the engineering design specifications, improving the computational efficiency of analytical design processes, and providing general digital computer program packages for linear control system synthesis.

1.2 Thesis Scope and Organization

The theory and application of a new performance index that brings engineering design specifications into the analytical design process is developed. A design procedure is established that starts with practical engineering specifications and uses this Model PI as a synthesis tool rather than an absolute criterion in itself. Although the techniques apply to linear, time invariant, deterministic control systems in general, the thesis is developed in the context of flight control systems in order to emphasize the relationship of realistic design requirements to the synthesis process. The Model PI method is compared to a model-referenced integral squared error design method in terms of the resulting design and the relative computational efficiency. A general digital computer program for control system design via parameter optimization is developed. Its usefulness is demonstrated by application to flight control system design examples.

The thesis is organized so that practical design problems are first introduced (Chapter 2), then the theory developed (Chapters 3 and 4), the general design procedure established (Chapter 5), and applied to flight control system design (Chapter 6). Some related topics on linear optimal control theory are treated in Chapter 7.

In Chapter 2 the requirements of flight control systems are discussed in terms of the inner loop aircraft handling qualities and outer loops such as autopilots. Th overall preliminary design process is reviewed in terms of the necessary steps to produce an acceptable design. Then the role of analytical design techniques in synthesizing a system to meet engineering specifications is discussed.

The Model PI theory is developed in Chapter 3 using an interesting geometrical representation of linear autonomous systems. A simple form is derived first for systems without zeros, and then is extended to systems with zeros and multivariable systems. The Model PI is shown to be related to the works of Aizerman (18) and Rekasius (19) which are critically reviewed. A summary of the important results of Chapter 3 is presented at the end of the chapter.

Chapter 4 presents a numerical method for minimizing the Model PI or any general quadratic functional. The procedure developed for evaluating the performance index and its gradient can alternatively be used with some other numerical optimization algorithm the designer may prefer.

In Chapter 5 the design procedures for using the Model PI are established and demonstrated for various forms of engineering specifications. This illustrates the use of a general digital computer program, described in Appendix B, for control system design via parameter optimization. By formulating a model-referenced integral squared error design method in state-space form the same computer program can be used for that method also. A comparison is made of these two design techniques to evaluate their relative effectiveness and computational efficiency. Methods for including parameter constraints in either method are presented.

Chapter 6 presents several flight control system applications of the Model PI design procedure. A simple pitch damper system is designed to provide satisfactory longitudinal handling qualities. A complex lateral-directional stability augmentation system is designed to illustrate the multivariable design methods. Finally, a pitch axis control system is designed for a VTOL aircraft. In each case the emphasis is on meeting realistic type design specifications for flight control systems.

Some interesting developments in linear optimal control theory resulting from the Model PI theory are presented in Chapter 7. Only the single control regulator problem is considered. The Model PI is shown to provide an interestingly simple solution to the linear optimal control synthesis procedure using root square locus. A procedure is presented for computing an equivalent Model PI for a general quadratic functional. This allows one to interpret the state vector weighting matrix in terms of a model response which the system will approach in a limiting case.

As a result of the research for this dissertation a general digital computer program for linear control system design has been developed. A description of the program, its operational format, and a complete listing are presented in Appendix B.

CHAPTER 2

FLIGHT CONTROL SYSTEM DESIGN

2.1 Flight Control System Requirements and Design Specifications

Flight control systems, in the most general sense, consist of several types of feedback loops for accomplishing various mission and operational requirements. Figure 2-1 depicts the functional aspects of a general flight control system for a piloted flight vehicle. The primary feedback loops of the flight control system are combined in figure 2-1 into the two categories; stabilization loops and auto-guidance loops. The stabilization loops include pitch, roll and yaw dampers, automatic Mach trim, structural mode suppression, etc. Such systems as autopilots and automatic approach and landing systems are indicated by the auto-guidance loops. The feedback loops contain vehicle motion sensors, noise filters, and some compensation networks. A major component of each loop is the control surface actuator system, which may be a complex feedback control system itself. Most high-performance aircraft require an artificial feel system for the pilot controls because of heavily boosted or completely irreversible control surface actuation systems. In the maneuvering mode, the pilot uses the control stick force as a primary measure of his maneuvering command inputs. This action closes a loop around the feel system. The pilot also closes the loop around the entire flight control system and aircraft using various motion cues, external visual cues and pilot instrument display as feedback sensors. Thus the effect of the pilot as an element in the maneuvering loop plays a dominant role in establishing the control system requirements.

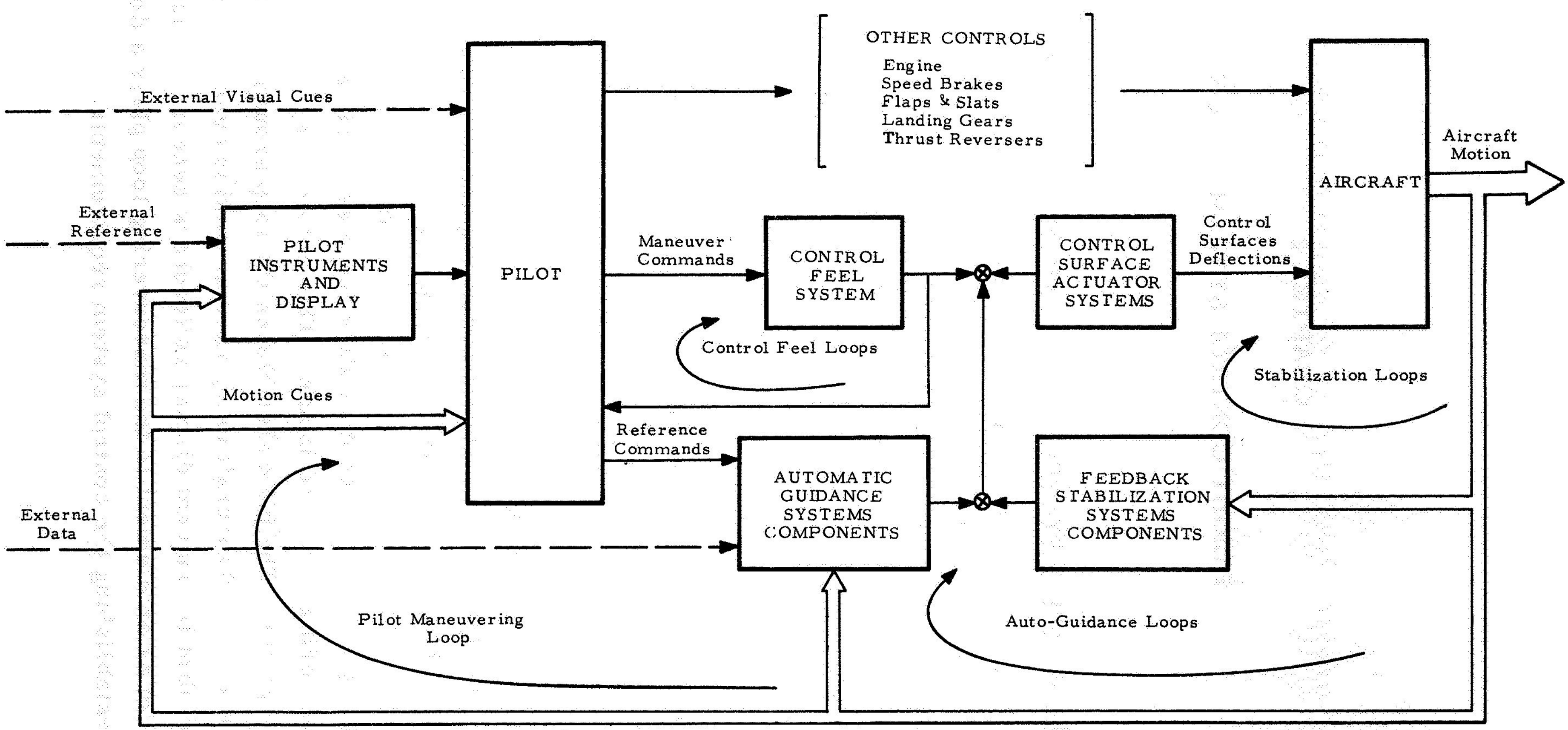


Figure 2-1 Functional Block Diagram of a General Flight Control System

The design of a flight control system presents several design problems, each with different requirements and correspondingly different specifications. These can be grossly categorized into design of the pilot maneuvering loop, the auto-guidance loops, and the control surface actuator systems. The requirements for the maneuvering loop are specified as aircraft handling qualities. These form the primary basis of the design specifications for the stabilization loops and control feel system. In auto-guidance modes of operation, the pilot is not in the active control loop, but monitors its operation and makes changes in the reference guidance commands, such as a desired heading change or reference holding altitude. The design specifications for auto-guidance loops, therefore, are largely independent of the handling qualities requirements and can be given in several forms more common to standard regulator or servo design problems. Although the control surface actuator systems are part of the other loops, they are generally designed separately to their own specifications and treated in the design of the outer loops as fixed dynamic elements. These are fairly standard servomechanism type design problems.

In designing flight control systems the engineer is faced with the various common forms of specifications as well as a special form based on aircraft handling qualities. It is important to have some knowledge of the types of design specifications involved in order to treat this subject from a realistic viewpoint. Some examples of aircraft handling qualities criteria are presented in the following section and discussed in relationship to the maneuvering loop design problem. Then in section 2.1.2 several common forms of engineering specifications that may arise in flight control system design problems are presented. These are discussed mainly in relationship to the design of auto-guidance loops and control surface actuator systems.

Flight control systems, like all physical control systems, are only linear over a limited amplitude range at best. At the preliminary design stage, it is usually quite adequate to treat them as linear systems unless the dominant characteristic is a nonlinear element. This thesis only considers linear or linearized control system design, hence specifications on nonlinear characteristics, such as saturation, hysteresis, and limit cycles, are not discussed. It should be recognized, however, that these are important and must be considered at some point in the design process. One aspect of nonlinearity can be treated in the linear analysis. That is, the amplitude range for which the linearization is valid can be specified as a constraint on the design. Similar constraints arise naturally from the aircraft limitations such as a normal acceleration limit or maximum roll rate limit. These constraints are included as part of the design specification.

2.1.1 Some Aircraft Handling Qualities Criteria

Handling qualities criteria are requirements on the vehicle's dynamic characteristics and control feel characteristics necessary to produce specified levels of closed-loop pilot-vehicle performance for a specific task or mission. Extensive analytical and experimental research has been and continues to be expended on developing adequate criteria for the various types of flight vehicles. The intent here is not to survey the field, but merely to present examples of typical criteria. For more specific details and bibliographies the interested reader is referred to references 33-36.

2.1.1.1 Longitudinal Handling Qualities

Longitudinal handling qualities for fixed wing aircraft are typically given as functions of the short period mode natural frequency, ω_{sp}, and damping, $\zeta_{sp}\omega_{sp}$. Figure 2-2 is a representative example. It shows boundaries of decreasing levels of pilot-vehicle performance, from Minimum Satisfactory to Minimum Flyable for combinations of ω_{sp} and $\zeta_{sp}\omega_{sp}$. The words "Satisfactory", "Acceptable", "Flyable" have specific connotations relative to pilot opinion ratings as indicated in table 2-1. These data are from reference 34 and were established from fixed-base and in-flight simulations of a piloted re-entry vehicle. The region indicated as "Good" handling qualities is from an earlier in-flight simulation study (37).

The primary design specification for the longitudinal stabilization loop can be given in terms of a desired region in the ω_{sp} -

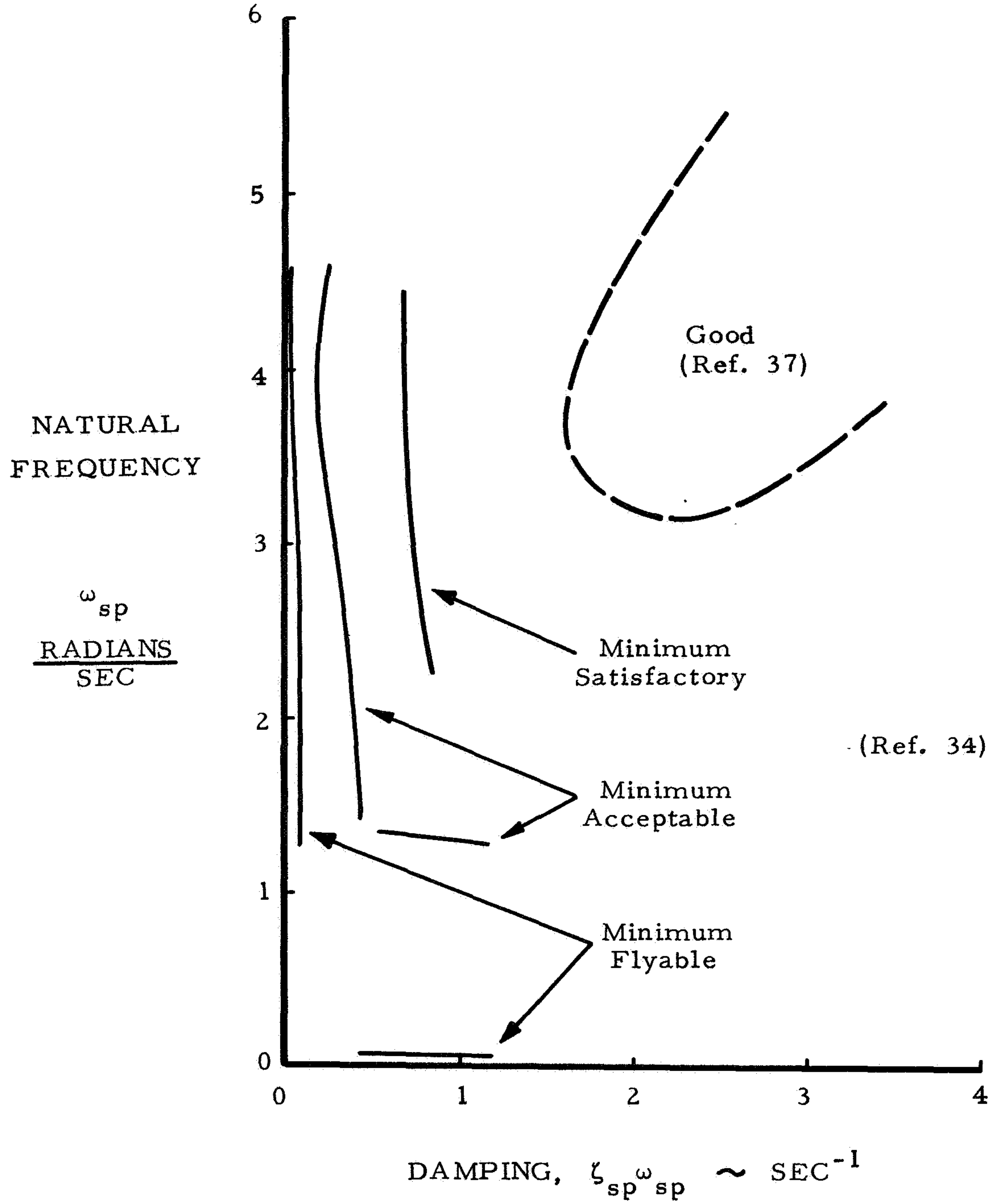


Figure 2-2 Typical Longitudinal Handling Qualities Criteria

TABLE 2-1

TYPICAL PILOT'S RATING SCALE*
FOR
EVALUATING AIRCRAFT HANDLING QUALITIES

NUMERICAL RATING	HANDLING QUALITIES CATEGORY	ADJECTIVE DESCRIPTION WITHIN CATEGORY
1		Excellent
2	Satisfactory	Good
3		Fair
4	Acceptable	Fair
5	but	Poor
6	Unsatisfactory	Bad
7	Unacceptable	Bad
8	but	Very Bad
9	Flyable	Dangerous
10	Unflyable	

*Modified Cooper pilot's rating scale, references 38, 34, 39.

$\zeta_{sp}\omega_{sp}$- plane for the closed-loop short period mode poles. It may be totally unrealistic to specify the "Good" region for all flight conditions because of unnecessarily severe demands it might place on the control system. Specifying a minimum Satisfactory level of performance would be more realistic. The designer would still attempt to obtain the best handling qualities possible (closest to the "Good" region) within the practical limitation of the control system. In other words, although a minimum specification is set as a region of allowable designs the relative performance within the region is still important to the designer. On the other hand there is clearly no single best or optimum design.

Criteria such as figure 2-2 do not completely determine the longitudinal handling qualities. Chalk (39) has shown that the short period mode zeros and static sensitivities have an important but lesser effect. Other factors are the phugoid mode characteristics, pitch-up characteristic, stick force per "g", and other control feel characteristics. However these usually are treated separately from the stabilization loop design problem.

2.1.1.2 Lateral-Directional Handling Qualities

Lateral-directional handling qualities for fixed wing aircraft are more complicated because of potentially strong coupling between the Dutch-roll, roll-subsidence, and spiral modes. Control of bank-angle with the ailerons is a primary task in the lateral-directional maneuvering loop. The aileron, δ_a, to bank angle, ϕ, transfer function in most cases is of the form

$$\frac{\phi(s)}{\delta_a(s)} = \frac{L_{\delta_a}(s^2 + 2\zeta_\phi\omega_\phi s + \omega_\phi^2)}{\left(s + \frac{1}{\tau_s}\right)\left(s + \frac{1}{\tau_R}\right)\left(s^2 + 2\zeta_d\omega_d s + \omega_d^2\right)} \qquad (2\text{-}1)$$

The parameters of primary importance to the lateral-directional handling qualities are

$L_{\delta_a} \delta_{a_{max}}$	roll control power
τ_R	roll-subsidence mode time constant
τ_s	spiral mode time constant
$\zeta_d \omega_d$	Dutch-roll mode damping
$\frac{\omega_\phi}{\omega_d}$	ratio of numerator frequency to Dutch-roll frequency
$\frac{\lvert\phi\rvert}{\lvert\beta\rvert}$	bank angle to side slip angle ratio

Extensive parametric studies have been made to relate pilot opinion to these parameters. Ashkenas (33) recently compiled and compared the results of many of these studies. Figure 2-3 is a composite of several criteria in a form that might be used as the design specifications for the lateral-directional stabilization loop. It presents criteria for the Dutch-roll mode, roll-subsidence mode and spiral mode. The data for the Dutch-roll and spiral modes are based on summary figures and tabulated data in reference 33. The roll-subsidence mode data were obtained from reference 40 for one value of roll control power. This figure should not be taken as a general criterion, but as a representative example of the type of specifications that might be given for a specific design problem.

The criteria shown in figure 2-3 for the Dutch-roll mode consists of two parts. The locations for poles corresponding to various levels of handling qualities are indicated by the nearly vertical boundaries to the right. And the ranges of ω_ϕ/ω_d for Acceptable, Satisfactory and Good handling qualities are tabulated for a given range of $\lvert \phi/\beta \rvert$. Near the origin is shown the spiral mode criterion for Satisfactory handling qualities. The roll-subsidence mode criterion is given as ranges of the roll mode pole locations corresponding

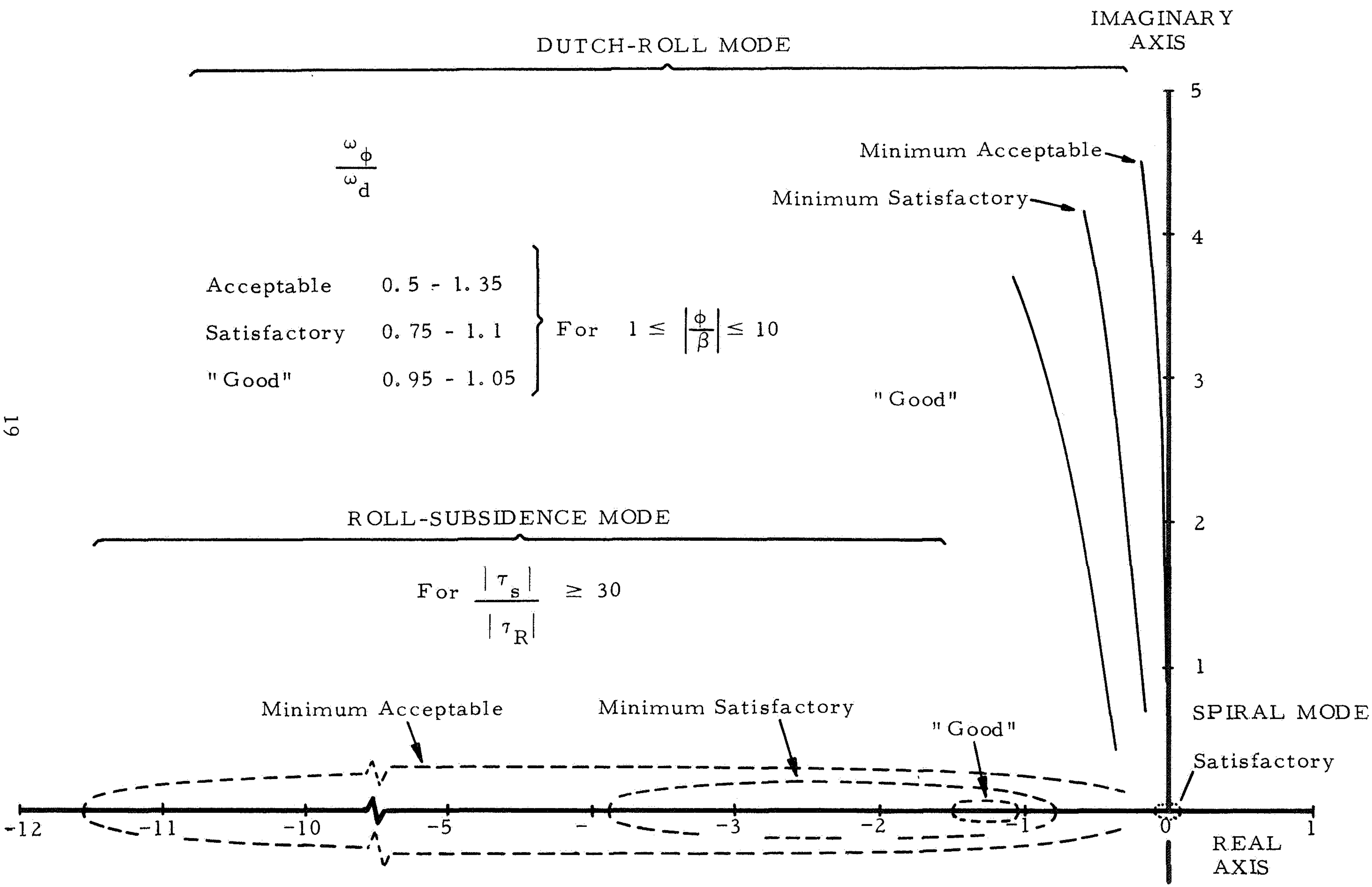


Figure 2-3 Composite of Typical Lateral-Directional Handling Qualities Criteria

to three levels of handling qualities. This is valid for the given range of spiral to roll mode time constants, which is usually more than satisfied. All these criteria must be met for the lateral-directional handling qualities to be at the desired level.

A realistic design specification might require Minimum Satisfactory lateral-directional handling qualities. Then by figure 2-3 the Dutch-roll mode poles must lie to the left of the Minimum Satisfactory boundary, the ω_ϕ/ω_d ratio must be within the Satisfactory range; and, the roll and spiral mode poles must lie within their respective boundaries on the real axis for Satisfactory handling qualities. Again here, as in the longitudinal case, the designer would try to make the closed loop system characteristics be as close to the "Good" range as possible within reasonable demands on the control system.

2.1.1.3 VTOL Handling Qualities - Pitch Axis

The two primary handling qualities parameters for pitch control of VTOL aircraft at low speeds and hover are control power and damping. Handling qualities requirements have been established in terms of these parameters using various piloted flight simulators. A typical example from reference 35 is presented in figure 2-4. Once the pitch control power is established the design specification for the stabilization loop could be given as an allowable range of damping for Satisfactory handling qualities. The designer would still try to obtain the best damping not just that corresponding to the minimum Satisfactory handling qualities.

2.1.2 Other Design Specifications

Design specifications for automatic control systems in general occur in numerous forms using such terms as rise time, peak overshoot, bandwidth, peak frequency, gain margin, etc. The various forms developed from practical engineering requirements and conventional control system analysis and design techniques. An extensive study of most types of specifications for linear automatic control systems was performed by Gibson, et. al. (17) in an attempt to establish a standard set. Their recommendations were summarized in

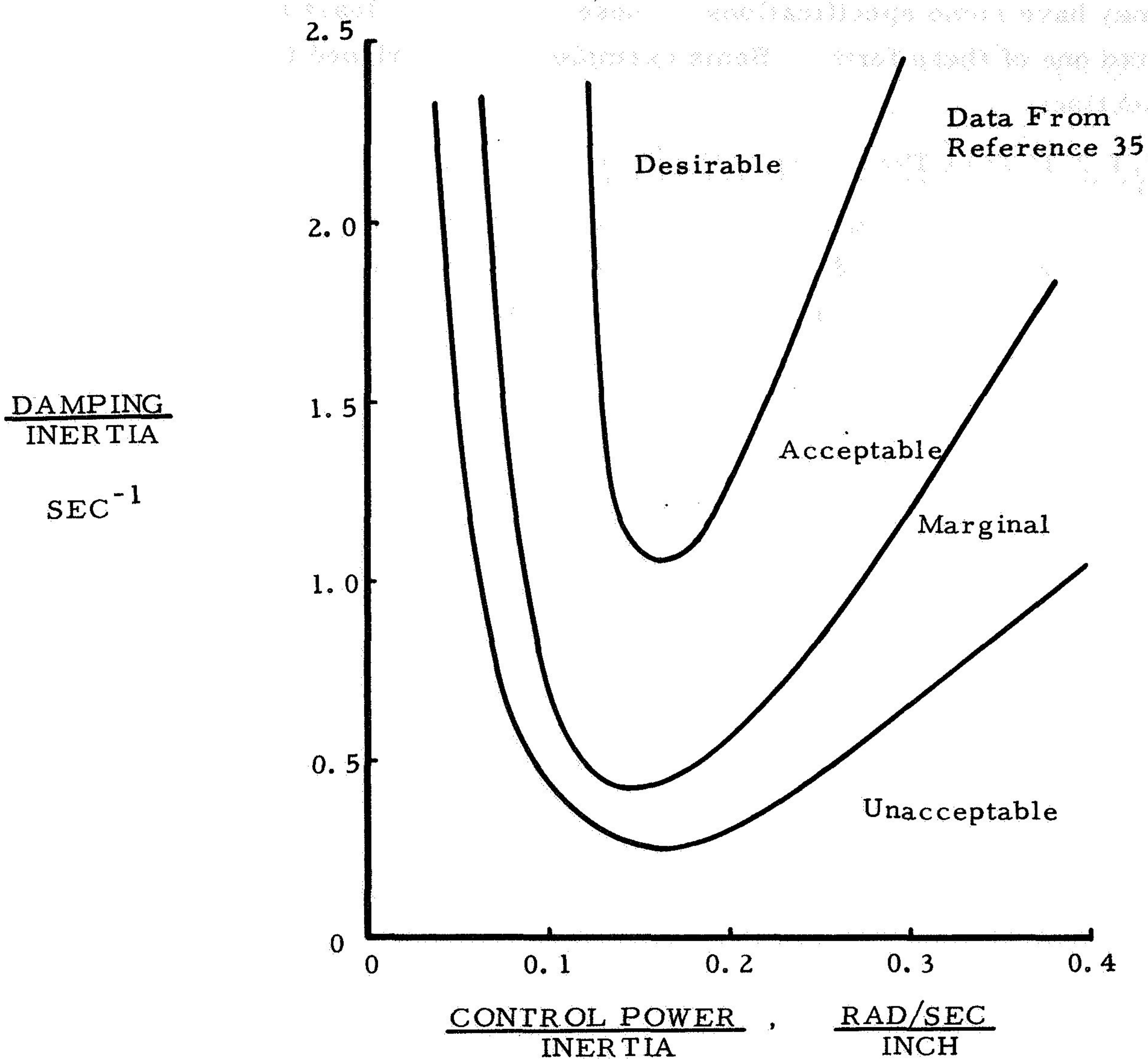


Figure 2-4 Typical VTOL Handling Qualities Criteria for Pitch Axis Control

standard graphical representations of specifications in the time domain and frequency domain. Flight control system design problems may have some specifications in these forms or at least transformable into one of these forms. Some examples are mentioned in the following sections

2.1.2.1 Time Domain Specifications

The standard time domain specifications recommended in reference 17 is based on the step response of the closed loop system. Figure 2-5 shows the recommended graphical representation and its relationship to the common time domain specification terms. If just the common terms are given together with allowable tolerances such a diagram is easily drawn. This type of specification implies that any design with a step response lying within the given envelope is acceptable and, further, that all such designs are equally acceptable. It is possible to use two or more diagrams similar to figure 2-5 for indicating different levels of acceptability.

Portions of the design specifications for an auto-guidance loop in a flight control system may be given in the form of figure 2-5. For example, the specifications for capturing the ILS beam in an automatic approach system could be put into an equivalent step response specification. Also the response of an autopilot to a step change in the reference altitude of an altitude hold mode, or a step heading change in the VOR tracking mode could be specified similarly.

2.1.2.2 Frequency Domain Specifications

The standard frequency domain specifications recommended in reference 17 is based on the amplitude ratio of the closed loop frequency response. Figure 2-6 shows the graphical form and its relationship to some common terms in frequency domain specifications. This is only illustrative since the specific shape of the tolerance "box" would depend on the specific application. Again, as mentioned in the previous section, it is possible to use two or more diagrams similar to figure 2-6 to specify different levels of acceptable frequency response.

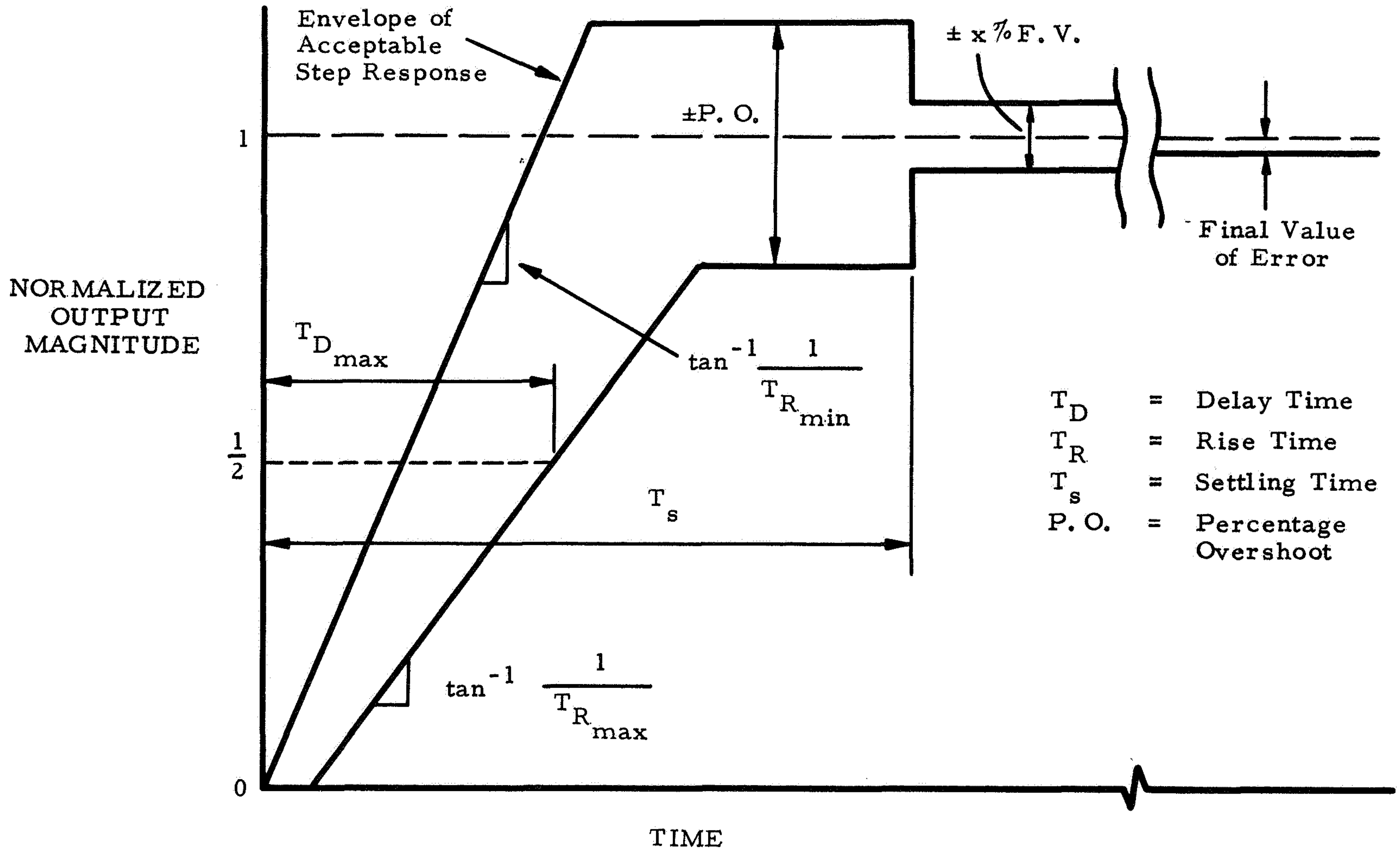


Figure 2-5 Graphical Form of Time Domain Specifications Recommended in Reference 17

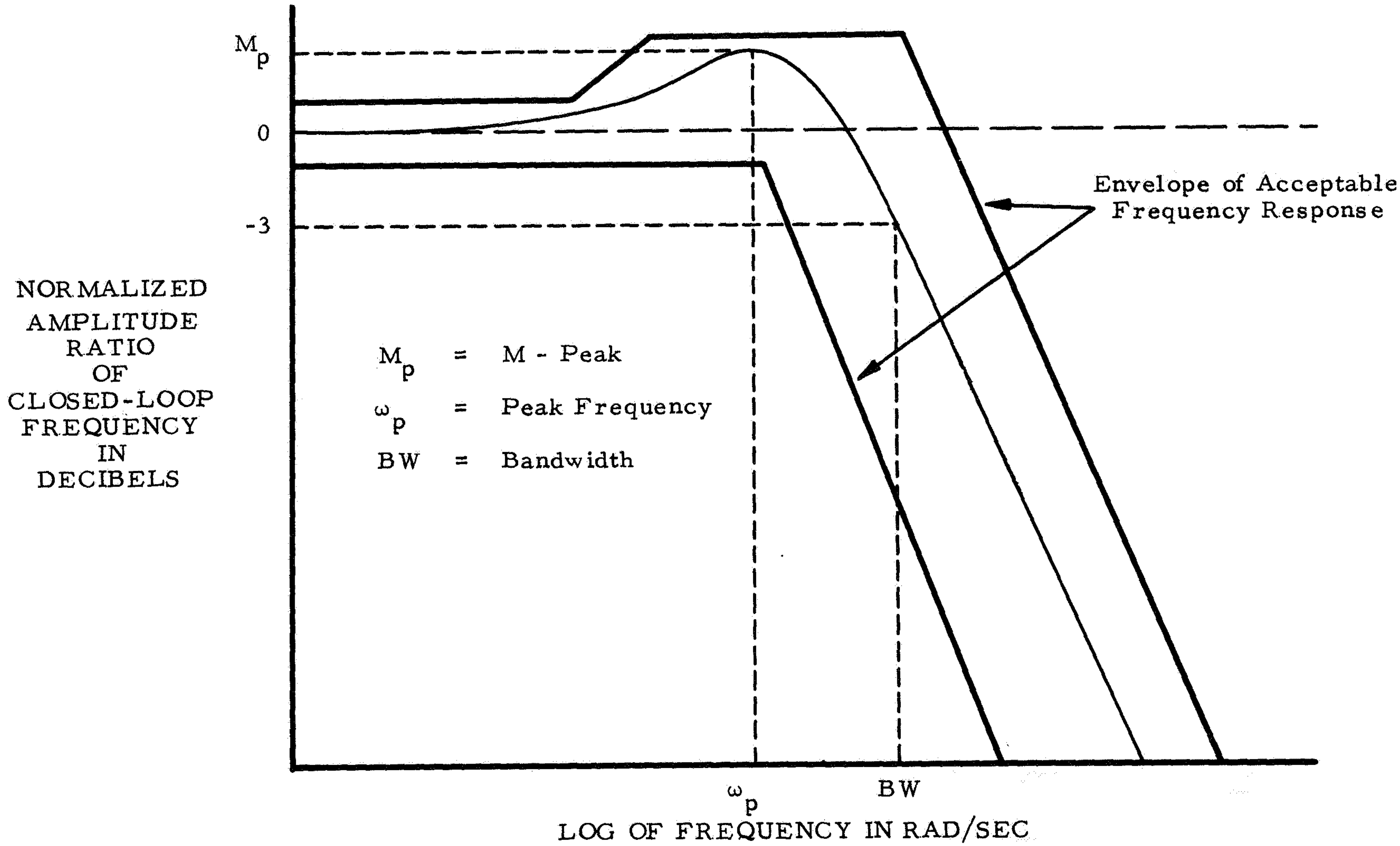


Figure 2-6 Graphical Form of Frequency Domain Specifications Recommended in Reference 17

The design specifications for the control surface actuator systems are typically presented in the frequency domain and quite likely in the form of figure 2-6. A common configuration is to use a small electro-hydraulic servo to drive the valve of a larger hydraulic actuator that actually moves the control surface. The design specification on the frequency response from the electrical input at the servo to the actuator ram displacement may be given in the form of figure 2-6.

2.2 Preliminary Design Process

Designing a flight control system involves, at the preliminary stage, a combination of technical engineering and management analyses, evaluations, and decisions. The goal is to establish a conceptual design that has a high probability of satisfying the overall acceptance criteria when implemented, has satisfactory interface with other flight systems, and can be produced at a competitive, or at least acceptable, cost. Obtaining this goal is a multiloop iterative process consisting of design, analysis, redesign, simulation, evaluation, and finally a decision.

One of the first steps of the preliminary design team is to transform the general mission and operating requirements into design specifications. These must reflect not only the direct flight control system requirements of the previous section but also the interaction with the aerodynamic, structural and subsystem requirements. Often there are incomplete and even conflicting requirements which must be reduced through experience and simplified analyses to one set of compatible, quantitative specifications. In addition there are ambiguous requirements of design simplicity, maintainability, reliability, flight safety, etc. and indirect factors such as time schedules and cost limitations.

Practical performance specifications seldom dictate a unique design solution. Rather they allow an acceptable range or envelope of solutions as indicated in the previous section. The role of the control system engineer is to establish, as it were, candidate designs within this acceptable range. These candidates are filtered down to a final preliminary design based on the more subjective criteria of simplicity, cost, maintainability, etc. These two functions are not distinct, sequential operations. The subjective criteria must be kept in mind by the designer as he works to satisfy the quantitative specifications.

This thesis considers the problem of synthesizing candidate control system designs by analytical methods, once the quantitative specifications have been established. It is thus recognized as only the middle portion, an inner loop, of the total preliminary design process. Real time simulation of a proposed flight control system is a very important part of the preliminary design process, particularly if a human pilot is to take part in closing the loop. There is invariably some redesign during this phase which may be done directly on the simulator or with the original analytical method. Simulation per se is not considered here, other than this brief recognition of its important role in the preliminary design process.

2.3 The Role of Analytical Design Techniques

The control system design engineer has many different synthesis techniques available and should always select a procedure commensurate with the scope of the problem and the desired results. In many cases conventional "cut and try" methods using root locus, Nyquist or Bode diagrams are by far the most appropriate to use. Analytical techniques based on optimization theory attempt to provide a more direct synthesis procedure than the conventional "cut and try" methods, and for complicated design problems are potentially the most effective. Many flight control system design problems involve multiloop or multiple input-output design which is difficult or at least tedious and time consuming to do by conventional techniques. Analytical methods can reduce the number of design iterations by considering the effect of several if not all design parameters on the system performance simultaneously.

Parameter optimization and linear optimal control are two fundamentally different analytical design techniques based on minimizing a mathematical performance index. The most effective performance indices in the synthesis of flight control systems are quadratic functionals that include, in some way, a model of the desired closed loop response. The one most widely used in parameter optimization problems, is the integral squared error between the model and system outputs for a specific input. A more general form is used in the model-following approach of linear optimal control theory. It is a quadratic functional of the error between several output variables of the model and system plus a quadratic penalty on the control effort (26). An alternate procedure, termed model-in-the-performance-index is also used in linear optimal control (26). The object of including a model in each of these is to represent the design specifications for the closed-loop response, which may be given in one of the forms discussed in section 2. 1. The model itself is not the specification.

Optimization theory does not provide an automatic method for designing control systems as is sometimes implied in the literature. Because of uncertainties in representing typical engineering specifications by a mathematical performance index, these techniques must be considered as part of a design iteration loop. Figure 2-7 illustrates in a simplified manner the roles of an optimization procedure in designing a practical flight control system.

Assuming that the design specifications have been established, the first item is to select a suitable linear model to represent the dynamic response portion of the specifications in a performance index. This is not an insignificant matter in itself. The mathematical representation of the flight vehicle is used in some cases to establish the form of a suitable model (27). However, there is, as yet, no rigid set of rules for selecting a model, and indeed, there is no guarantee that a design based on any specific model will meet the actual design specifications. The problem of model selection is treated in part in reference 41 and also in later chapters of this thesis.

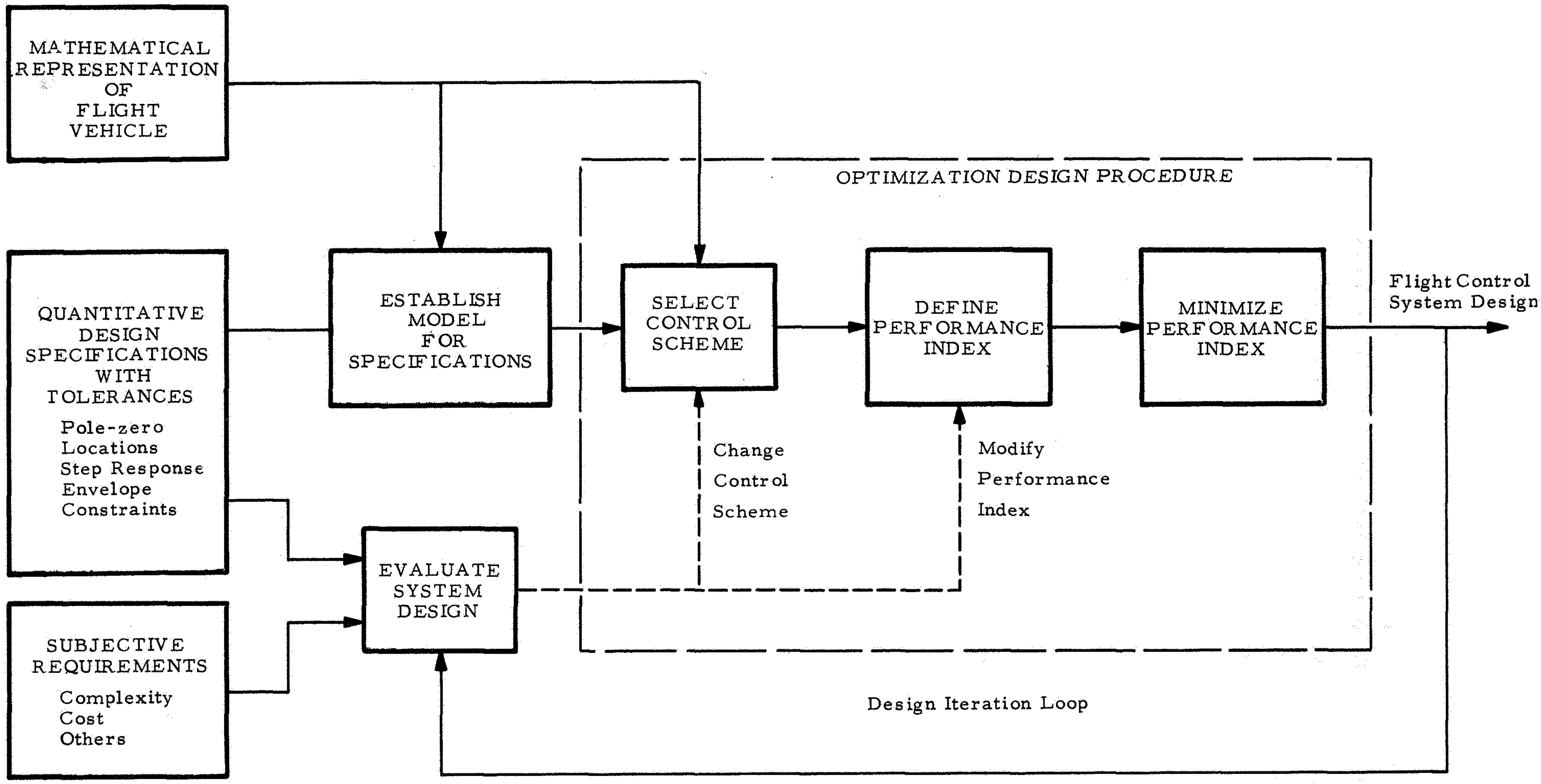


Figure 2-7 A Functional Representation of the Role of Optimization Procedures in Flight Control System Design

Once a model has been selected, an optimization design problem is formulated. Figure 2-7 shows as the first element of the optimization design procedure, the selection of a control scheme. In a parameter optimization problem, this represents selection of the feedback and compensation configuration. In an optimal control problem, the control configuration in theory is not selected a priori but in application is of a known form.

The next step is to define a performance index that includes the model in some way, and this varies depending on the procedure used. Constraints on parameters, signals, or control effort may also be represented in the performance index, or retained as auxiliary constraint equations as are the differential equations representing the flight vehicle. A control system design is then determined that minimizes the performance index subject to the auxiliary constraints. Optimization theory is well established for a large class of idealized system design problem (e. g. references 20 and 25). The design thus obtained is optimum with respect to the idealized mathematical performance criterion and constraints imposed. However it must be evaluated against the original quantitative specifications and the more subjective requirements to be an acceptable design. If these are not satisfied, the optimization procedure is repeated with a different control scheme and/or a modified performance index. In a parameter optimization problem this might correspond to changing the form of compensation used. In optimal control theory it might require a change in the relative weighting of variables in the performance index. Thus when practical engineering specifications must be met, optimization design procedures form part of the design iteration loop. The aim of developing the practical application of optimization techniques to synthesizing control systems is to provide rapid convergence of the iteration to a satisfactory design.

The viewpoint taken in this thesis regarding the role of optimization and optimal control theory in designing control systems is fundamentally different from that used to derive the theory. The theory assumes that it is possible to define mathematically the design objective and, therefore, an optimum system design may exist (or at least it may under certain well defined conditions). To the extent a performance index truely represents the physical objective of a problem, the theory applies directly, and it is reasonable to refer to an optimal design. However, when the design goal is to satisfy typical engineering specifications that allow some tolerances, a performance index becomes a guide in the selection of one of the acceptable designs, rather than a criterion itself. The flight control system design problems considered here fall into the latter category, and therefore the term "optimal", and hence "sub-optimal", control system becomes rather meaningless and is used in this thesis only to the extent necessary to discuss related work in the literature.

CHAPTER 3

MODEL PERFORMANCE INDEX THEORY

In this chapter a performance index is developed that includes a model in an entirely different manner than those discussed in the previous chapter. It is based on a geometrical criterion for approximating one dynamical system (the model) by another (the actual system). The basic form of the resulting "Model Performance Index" (Model PI) is the same as that of quadratic functionals frequently appearing in modern control theory. The important point, however, is the ability to interpret the performance index directly in terms of a model of the desired system response.

3.1 Transient Response of Linear Invariant Systems

Development of the Model PI theory requires a clear understanding of some fundamental properties of linear invariant systems. For the time being the discussion is restricted to single input/output systems that can be described, in general, by the nth order differential equation

$$y^{(n)}(t) + a_{n-1}y^{(n-1)}(t) + \cdots + a_2\ddot{y}(t) + a_1\dot{y}(t) + a_0y(t) = b_mu^{(m)}(t) + \cdots + b_1\dot{u}(t) + b_0u(t) \tag{3-1}$$

where y is the output, u is the input and $m \leq n-1$. The system's initial conditions are assumed to be zero. The corresponding transfer function is

$$\frac{y(s)}{u(s)} = \frac{b_m s^m + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0} \tag{3-2}$$

There are many alternate ways to represent mathematically the transfer characteristics of a system. The representation considered here is the transient portion of the time response of the system for a unit step input. Assuming that a finite steady-state value of the output, y_{ss}, exists for a step input, the transient portion of the response is defined as

$$x(t) = y(t) - y_{ss} \tag{3-3}$$

The transform of the transient response is easily obtained from (3-2) as

$$x(s) = \frac{b_m s^{m-1} + \cdots + b_3 s^2 + b_2 s + b_1}{s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0} - \frac{\frac{b_0}{a_0}\left(s^{n-1} + a_{n-1}s^{n-2} + \cdots + a_2 s + a_1\right)}{s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0} \tag{3-4}$$

A more convenient form can be obtained by defining a specific set of hypothetical or <u>pseudo initial conditions</u> for x(t) that produces a response identical to the unit step response. That is, it is possible to replace the step input with an appropriate set of pseudo IC's such that the transient response is the same. The reason for doing this is that the effect of the system's closed-loop transfer function zeros can be included in a performance index developed subsequently by these pseudo IC's. The transient response is then described by the homogeneous differential equation

$$x^{(n)}(t) + a_{n-1}x^{(n-1)}(t) + \cdots + a_1\dot{x}(t) + a_0x(t) = 0 \qquad (3\text{-}5)$$

with pseudo IC's denoted by

$$\begin{aligned} x(0) &= x_0 \\ \dot{x}(0) &= \dot{x}_0 \\ &\vdots \\ x^{(n-1)}(0) &= x_0^{(n-1)} \end{aligned} \qquad (3\text{-}6)$$

where the values of x_0, $\dot{x}_0$, etc. are yet to be determined to give the desired equivalence. These can be established easily by taking the Laplace transform of (3-5) and comparing the result with (3-4) i. e.

$$\begin{aligned} (s^n + a_{n-1}s^{n-1} + \cdots + a_2s^2 + a_1s + a_0)x(s) \\ = \quad & (\dot{x}_0)s^{n-2} \\ + \quad & (\ddot{x}_0 + a_{n-1}\dot{x}_0)s^{n-3} \\ & \vdots \\ + \quad & (x_0^{(n-3)} + a_{n-1}x_0^{(n-4)} + \cdots + a_4\dot{x}_0)s^2 \\ + \quad & (x_0^{(n-2)} + a_{n-1}x_0^{(n-3)} + \cdots + a_4\ddot{x}_0 + a_3\dot{x}_0)s \\ + \quad & (x_0^{(n-1)} + a_{n-1}x_0^{(n-2)} + \cdots + a_3\ddot{x}_0 + a_2\dot{x}_0) \\ + \quad & x_0(s^{n-1} + a_{n-1}s^{n-2} + \cdots + a_2s + a_1) \end{aligned} \qquad (3\text{-}7)$$

Note that x_0 occurs in (3-7) only as a factor of the last term. Comparing the last term of (3-7) to the numerator of (3-4) it is clear that

$x_0 = -b_0/a_0$. Then, equating the coefficients of like powers of s of the remaining terms on the right hand side of (3-7) with those of the numerator of the first term in (3-4), results in the relationships

$$x_0 = -\frac{b_0}{a_0}$$

$$x_0^{(n-i)} = \begin{cases} 0 & \text{for } i > m \\ b_i - \sum_{j=n-m}^{n-i-1} a_{j+i} x_0^{(j)} & \text{for } i = 1, 2, \cdots m \end{cases} \tag{3-8}$$

Thus if one uses the initial condition given by (3-8) with the homogeneous equation (3-5), the time response would be identical to the transient response of the original system equation (3-1) for a unit step input. An alternate interpretation of the pseudo IC's is that they are the values of the transient response variable, x(t), and its (n-1)th derivatives at an infinitely small time increment after the application of the unit step input.

The work that follows is based on representing the forced dynamical system by an autonomous system with a specific set of pseudo initial conditions (3-8). The initial conditions will always be referred to as pseudo in order to emphasize that they are not the actual initial conditions of the original system equation (3-1). These pseudo IC's contain the effect of system zeros and are also functions of the characteristic equation coefficients. This is shown later to be an important point in the development of the Model PI, a point not treated properly in previous literature.

The following state space form of the transient response provides a convenient and useful compact notation:

Define a state vector* $\underline{x}$,

---

* The time argument is suppressed for notational convenience. The transpose of a vector or matrix is denoted by ( )'.

$$\underline{x}' = [\, x \;\; \dot{x} \;\; \ddot{x} \; \cdots \; x^{(n-2)} x^{(n-1)} \,] \tag{3-9}$$

then equation (3-5) is given by

$$\dot{\underline{x}} = F \, \underline{x} \tag{3-10}$$

where

$$F = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-2} & -a_{n-1} \end{bmatrix} \tag{3-11}$$

This is known as the phase variable canonical form. A more compact notation for F is

$$F = \left[\begin{array}{c|c} \underline{0} & I \\ \hline \multicolumn{2}{c}{-\underline{a}'} \end{array} \right] \tag{3-12}$$

where

$$\underline{a}' = [\, a_0 \; a_1 \; a_2 \cdots\cdots a_{n-2} \; a_{n-1} \,] \tag{3-13}$$

The pseudo initial condition for (3-10) is

$$\underline{x}'(0) = \underline{x}_0' = [\, x_0 \; \dot{x}_0 \; \ddot{x}_0 \cdots\cdots x_0^{(n-2)} \; x_0^{(n-1)} \,] \tag{3-14}$$

where x_0, $\dot{x}_0$, $\cdots$, $x_0^{(n-1)}$ are given by (3-8).

3.1.1 A Geometrical Property of Autonomous Systems

A geometrical property of autonomous systems is considered in this section that presents a novel characterization and leads to a new criterion for approximating one system by another. For the general nth order autonomous system (3-5) define (n+1)th order partitioned vectors

$$\underline{\tilde{x}}' = [\ \underline{x}' \ \vdots\ x^{(n)}\ ] = [\ \underline{x}' \ \vdots\ -\underline{x}'\underline{a}\ ] \tag{3-15}$$

$$\underline{\tilde{a}}' = [\ \underline{a}' \ \vdots\ 1\ ] \tag{3-16}$$

where $\underline{x}$ and $\underline{a}$ are defined by (3-9) and (3-13) respectively. The (n+1)th order space is referred to here as the extended state space. Then the autonomous system (3-5) can be written as

$$\underline{\tilde{x}}'\underline{\tilde{a}} = 0 \tag{3-17}$$

with pseudo IC

$$\underline{\tilde{x}}_0' = [\ \underline{x}_0' \ \vdots\ -x_0'\underline{a}\ ] \tag{3-18}$$

Equation (3-17) defines a hyper-plane in the extended state space normal to the constant vector $\underline{\tilde{a}}$. The trajectory of $\underline{\tilde{x}}$ as a function of time (i. e. the transient response and its first n derivatives) must lie within the hyper-plane. Any other autonomous system producing a trajectory that lies within this plane can only differ from the first by its pseudo IC. Thus the hyper-plane contains the trajectories of all possible systems with the same characteristic equation, and is referred to here as the characteristic plane or simply the $\underline{\tilde{a}}$-plane. A linear, invariant system can therefore be completely described by its characteristic plane and pseudo IC's.

This geometrical interpretation is illustrated for two simple second order systems with the same characteristic plane in figure 3-1. The systems considered are given by

$$\underline{\tilde{x}}'\underline{\tilde{a}} = \ddot{x} + \sqrt{2}\ \dot{x} + x = 0 \tag{3-19}$$

so that

$$\underline{\bar{x}}' = [x \quad \dot{x} \quad \ddot{x}]$$
$$\underline{\bar{a}}' = [1 \quad \sqrt{2} \quad 1] \qquad (3\text{-}20)$$

with pseudo IC's of

$$\underline{\bar{x}}'_{1_0} = [-1 \quad 0 \quad 1] \qquad (3\text{-}21a)$$

$$\underline{\bar{x}}'_{2_0} = [1 \quad -0.25 \quad -0.75] \qquad (3\text{-}21b)$$

for figure 3-1, parts a and b respectively. The trajectories (time responses) for both are seen to lie within the $\underline{\bar{a}}$-plane. Progression of time, in seconds, is indicated along the trajectories.

Each pseudo IC (3-21) together with (3-19) correspond to a different system transfer function. By reference to the relationship (3-8) between the transfer function coefficients and the pseudo IC's one readily sees that they represent

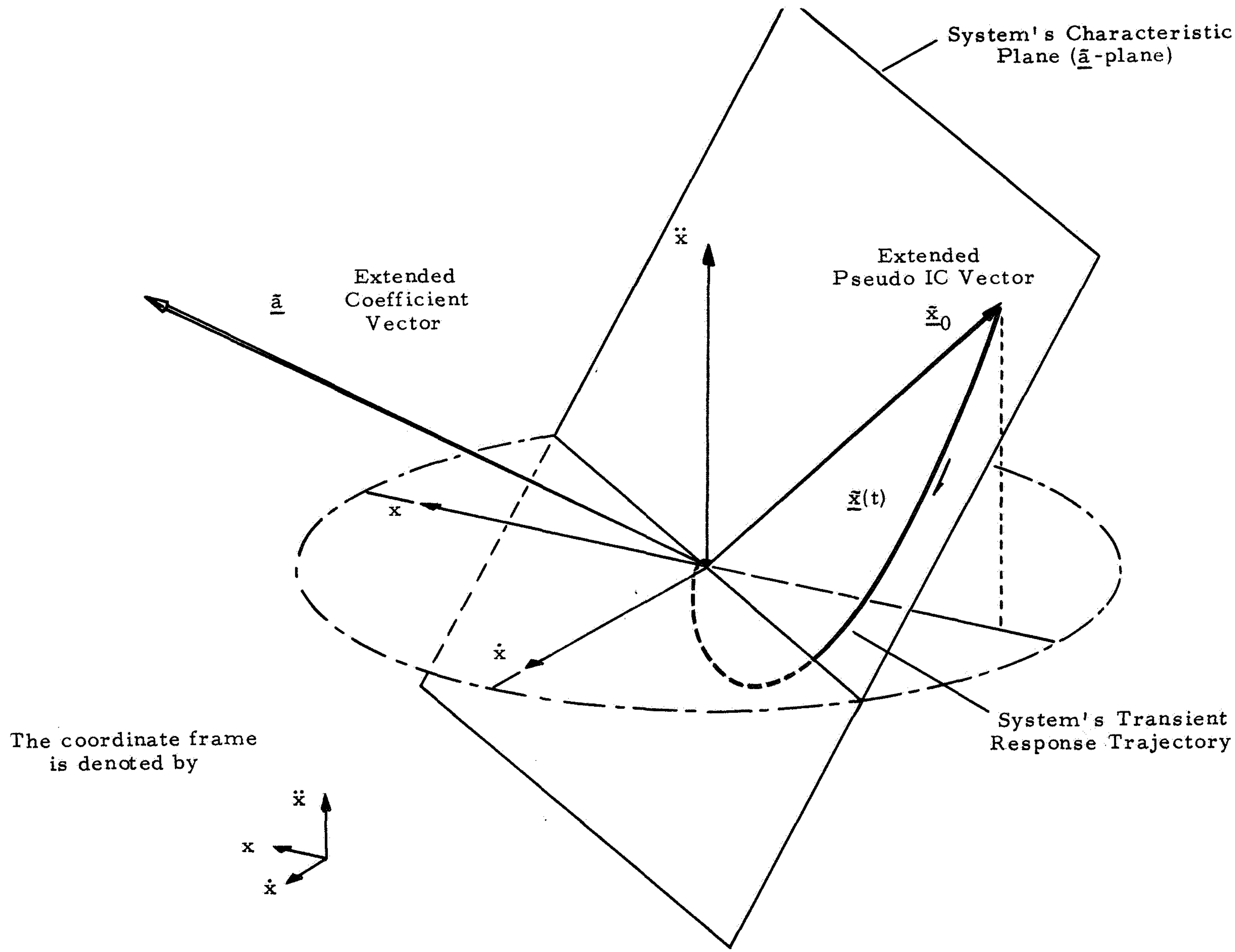
$$\frac{1}{s^2 + \sqrt{2}\,s + 1} \qquad (3\text{-}22a)$$

and

$$\frac{-0.25\,(s + 4)}{s^2 + \sqrt{2}\,s + 1} \qquad (3\text{-}22b)$$

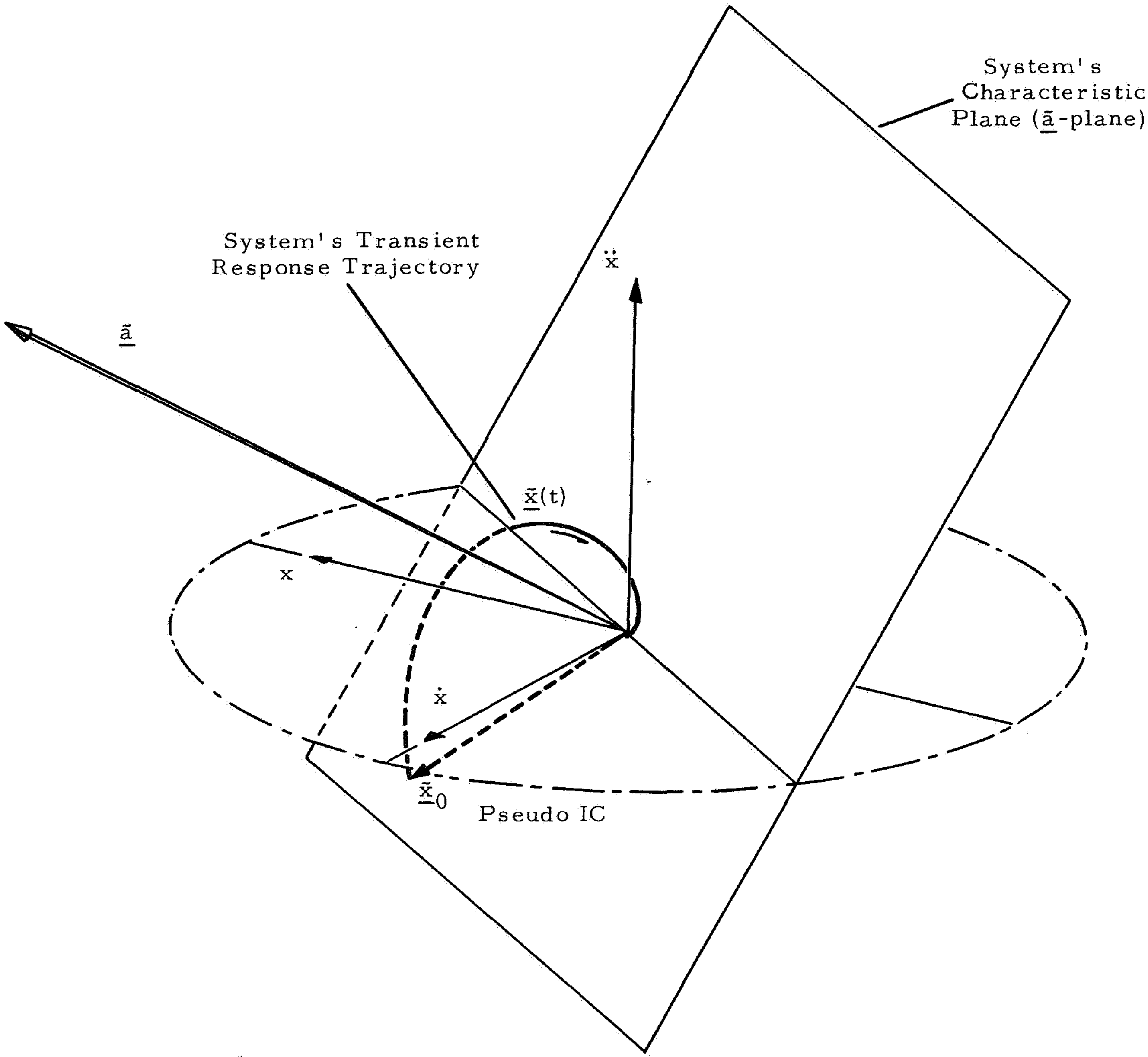
respectively for (3-21a) and (3-21b).

Representing a system by its characteristic plane and pseudo IC's does not set forth any new mathematical information. Stating that the trajectory of any nth order system lies within a plane in the extended state space of (n+1)th order, merely relates the fact that the nth derivative is a linear combination of the state variables, as in equation (3-5). However it provides a useful way of visualizing the process of approximating one system by another using the performance index developed in the next section.



(a) Pseudo IC Vector $\underline{\tilde{x}}_0' = [\,-1 \quad 0 \quad 1\,]$

Figure 3-1 Geometrical Representation of Two Linear Autonomous Systems with the Same Characteristic Plane



(b) Pseudo IC Vector $\tilde{x}_0' = [\,1 \quad -.25 \quad -.75\,]$

Figure 3-1 Concluded

3.2 Model Performance Index (Model PI)

The term "model" as used here relates specifically to dynamic response specifications of some system design problem. Usually it refers to a mathematical model of the desired, or at least satisfactory, response characteristics, so that the design objective would be to closely approximate the model. For the most part, it is assumed in this section that the model is already given, although some factors affecting the choice of the model structure, i. e. number of poles and zeros, are discussed. Methods for selecting an appropriate model for various types of practical engineering design specifications are discussed subsequently.

As pointed out in the previous section a linear, invariant system can be represented geometrically by its characteristic plane and pseudo IC. If both the model and the system to be designed are represented in this fashion, one can establish criteria for approximating the model by the system in terms of their characteristic planes and pseudo IC's. The Model PI is one such criterion. The basic form of the Model PI can be thoughtof as a generalized measure of the distance between the system's time response trajectory and the model's characteristic plane. This concept can be derived and explained most clearly for the situation in which the closed-loop system to be designed has no zeros, as in the following section. It is extended to the more general case in section 3. 2. 2.

3.2.1 Systems Without Zeros

Consider a general closed-loop control system given by the transfer function

$$\frac{y(s)}{u(s)} = \frac{a_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0} \qquad (3\text{-}23)$$

This is the special case of (3-2) with no zeros and unity static sensitivity. Its transient response for a unit step input can correspondingly be represented by its characteristic plane (the $\tilde{\underline{a}}$-plane) defined in the (n+1)-dimensional space* by

$$\tilde{\underline{x}}'(t)\,\tilde{\underline{a}} = 0 \tag{3-24}$$

and pseudo IC vector

$$\tilde{\underline{x}}_0' = [\; -1 \;\vdots\; \underline{0}' \;\vdots\; a_0 \;] \tag{3-25}$$

where $\underline{0}$ is an (n-1)-dimension null vector.

The coefficient vector $\underline{a}$ is generally a nonlinear function of the free design parameters. For convenience, define a vector, $\underline{p}$, whose elements are the free parameters, then this functional dependence can be emphasized by writing (3-17) as

$$\tilde{\underline{x}}'(t)\,\tilde{\underline{a}}(\underline{p}) = 0 \tag{3-26}$$

Varying the free parameter vector changes $\tilde{\underline{a}}(\underline{p})$ and hence the orientation of the system's characteristic plane. From the geometrical viewpoint, selecting the free design parameters corresponds to selecting an orientation of the system's characteristic plane in the (n+1)-dimensional space. The significance of this will become apparent subsequently.

Assume for the moment that the model of the desired closed-loop system response is of the same order as the system and is given by

$$\frac{y_m(s)}{u(s)} = \frac{\alpha_0}{s^{\ell} + \alpha_{\ell-1}s^{\ell-1} + \cdots + \alpha_2 s^2 + \alpha_1 s + \alpha_0} \tag{3-27}$$

where $\ell = n$ in this case. The model can also be represented in the (n+1)-dimensional space by its characteristic plane (the $\tilde{\underline{\alpha}}$-plane),

---

* The ˜ denotes a vector in the (n+1)-dimensional extended state space of the control system.

defined by

$$\underline{\tilde{x}}'_m(t)\underline{\tilde{\alpha}} = 0 \qquad (3\text{-}28)$$

and pseudo IC vector

$$\underline{\tilde{x}}'_{m_0} = [\ -1 \ \vdots \ \underline{0}' \ \vdots \ \alpha_0\ ] \qquad (3\text{-}29)$$

where $\underline{\tilde{x}}_m(t)$ and $\underline{\tilde{\alpha}}$ correspond to (3-15) and (3-16) respectively for the model, and $\underline{0}$ is an (n-1)-dimension null vector.

Now both the system and model are represented geometrically in the (n+1)-dimensional space by their respective characteristic planes and pseudo IC vectors. The time response trajectory for each of these lies within the corresponding plane. Ideally the designer would like to make these trajectories coincide, which means perfect model matching. That is, the time histories of the system's and model's response to a step input would be identical. What conditions, in terms of this geometrical representation, are both sufficient and necessary for this to occur? The answer to this is fairly obvious but will be established rigorously.

A sufficient condition is considered first. If the orientation of the system's characteristic plane is selected, by means of the free design parameters, so that it coincides with the model's characteristic plane, then the two trajectories must at least lie within the same plane. And they can only differ if the system and model pseudo IC vectors are not equal. The first n elements of both pseudo IC vectors, (3-25) and (3-29), are equal. The last elements of each, the (n+1) elements, must also be equal for the two pseudo IC vectors to lie within the common characteristic plane. Therefore a sufficient condition for the trajectories of a system and model of the same order to be coincident is for the two characteristic planes to be coincident.

To show that this condition is also necessary, assume that the trajectories coincide but that the two characteristic planes are not coincident. Since the common trajectory must lie in both planes it must lie along the intersection of the two planes. The common pseudo

IC vector must also lie along this intersection so that the trajectory and pseudo IC vector must be colinear. This means that the time response trajectory, $\underline{\tilde{x}}(t)$, must equal the pseudo IC vector (3-25) multiplied by some scalar function of time, f(t), i. e.

$$\underline{\tilde{x}}'(t) = \underline{\tilde{x}}_0' f(t) = [\ -1 \ \vdots \ \underline{0}' \ \vdots \ a_0 \ ] f(t)$$

or (3-30)

$$[x(t) \quad \dot{x}(t) \quad \cdots \quad x^{(n-1)}(t) \qquad x^{(n)}(t)\ ] =$$
$$[\ -f(t) \quad 0 \quad 0 \ \cdots \ 0 \quad a_0 f(t)\ ]$$

However such a solution is not possible in general because it would require the first (n-1) derivatives of the transient response variable to be identically zero while the variable itself, x(t), and its nth derivative, $x^{(n)}(t)$, are non-constant functions of time. A first order system presents a special case in which the trajectory is colinear with the pseudo IC vector, i. e.

$$\begin{bmatrix} x(t) \\ \\ \dot{x}(t) \end{bmatrix} = \begin{bmatrix} -1 \\ \\ a_0 \end{bmatrix} e^{-a_0 t} = \underline{\tilde{x}}_0 \, e^{-a_0 t} \qquad (3\text{-}31)$$

But in that case the "characteristic planes" are actually lines and must coincide if the trajectories, which lie along these lines, coincide. Therefore it is not possible to have system and model trajectories that coincide when their characteristic planes do not. It follows that their characteristic planes must coincide if their trajectories coincide.

The complete answer to the question posed earlier is that <u>the time response trajectories of a system and model of the same order coincide if and only if the system and model characteristic planes coincide.</u>* To meet this condition requires selecting the free parameter

* If the system and model transfer functions have zeros, this statement is not necessarily true (see section 3. 2. 2).

vector such that $\underline{\tilde{a}}$ is colinear with $\underline{\tilde{\alpha}}$, i. e.

$$\underline{\tilde{a}}(\underline{p}) = c\underline{\tilde{\alpha}} \qquad (3\text{-}32)$$

where c is some constant scalar. Any criterion established for approximating the model by the actual system should give this as a solution if sufficient freedom is allowed in determining $\underline{\tilde{a}}$. However the functional relationship of $\underline{\tilde{a}}$ to $\underline{p}$ in most design problems precludes satisfying (3-32), which means that the two characteristic planes ($\underline{\tilde{a}}$-plane and $\underline{\tilde{\alpha}}$-plane) would intersect at some angle for all allowable values of $\underline{p}$ and the model's trajectory could not be matched identically. It was shown above that if the orientation of the $\underline{\tilde{a}}$-plane could be selected so that the system's trajectory would lie within the $\underline{\tilde{\alpha}}$-plane, then the two trajectories would coincide. It is reasonable to expect that if the system's trajectory could be made to lie close to the $\underline{\tilde{\alpha}}$-plane it would be close to the model's trajectory. One can certainly state that it must be close to the $\underline{\tilde{\alpha}}$-plane in order for the error between the two trajectories to be small since the model's trajectory lies in the $\underline{\tilde{\alpha}}$-plane. A criterion for approximating the model by the system can be established based on minimizing some generalized measure of the distance between the system's trajectory, $\underline{\tilde{x}}(t)$, and the $\underline{\tilde{\alpha}}$-plane.

The instantaneous distance from $\underline{\tilde{x}}(t)$ to the $\underline{\tilde{\alpha}}$-plane is given by*

$$\frac{\| \underline{\tilde{x}}'(t)\underline{\tilde{\alpha}} \|}{\| \underline{\tilde{\alpha}} \|} \qquad (3\text{-}33)$$

As mentioned previously, it is generally not possible to select the free parameter vector, $\underline{p}$, so that $\underline{\tilde{x}}(t)$ lies in the $\underline{\tilde{\alpha}}$-plane thus making (3-33) zero at each instant of time. A generalized measure of the distance from $\underline{\tilde{x}}(t)$ to the $\underline{\tilde{\alpha}}$-plane over the time interval 0 to ∞ is the following linear function space norm:

---

* The notation $\| \underline{v} \|$ is a compact notation for $\sqrt{\underline{v}'\underline{v}}$ which is the length of $\underline{v}$.

$$\left[\int_0^{\infty} \left(\frac{\| \tilde{\underline{x}}'(t)\, \underline{\tilde{\alpha}} \|}{\| \underline{\tilde{\alpha}} \|} \right)^2 dt \right]^{\frac{1}{2}} \qquad (3\text{-}34)$$

Selecting $\underline{p}$ to minimize (3-34) selects the system trajectory that lies the closest to the $\underline{\tilde{\alpha}}$-plane in the root-integral-square sense It is equivalent to use the square of (3-34) as a criterion, which is taken as the basis for the " Model Performance Index" The Model PI is defined as

$$PI = \int_0^{\infty} \left(\frac{\| \tilde{\underline{x}}'(t)\underline{\tilde{\alpha}} \|}{\| \underline{\tilde{\alpha}} \|} \right)^2 dt$$

$$= \int_0^{\infty} \left(\frac{\tilde{\underline{x}}'(t)\underline{\tilde{\alpha}}\underline{\tilde{\alpha}}'\tilde{\underline{x}}(t)}{\| \underline{\tilde{\alpha}} \| \| \underline{\tilde{\alpha}} \|} \right) dt$$

which can be written in the compact form*

$$PI = \int_0^{\infty} \| \tilde{\underline{x}}(t) \|_{\tilde{Q}}^2 \, dt \qquad (3\text{-}35)$$

where

$$\tilde{Q} = \frac{\underline{\tilde{\alpha}}\underline{\tilde{\alpha}}'}{\| \underline{\tilde{\alpha}} \|^2} \qquad (3\text{-}36)$$

It is termed the Model PI because the state vector weighting matrix, $\tilde{Q}$, is used to represent the model, or more correctly the model's characteristic plane, in the performance index. It possesses all the desired properties of a performance index, that is, it is a

---

* The notation $\| \underline{v} \|_M^2$, where M is a square matrix, means $\underline{v}' M \underline{v}$.

non-negative number that decreases as the system better approximates the model (in the sense defined earlier), goes to zero if the system matches the model exactly, and is of a convenient mathematical form. The weighting matrix, $\tilde{Q}$, is always a positive semi-definite matrix by its definition (3-36) and can be written down directly from the model's characteristic equation.

The derivation to this point was for models of the same order as the system ($\ell = n$). It will now be shown that the Model PI (3-35) is valid for any lower order model as well. The method is also valid for higher order models ($\ell > n$), but that case is not of much interest in the type of design problems considered.

The model is still assumed to be given by equation (3-27) except now $\ell < n$. Define the characteristic plane of an ℓth order model in its $(\ell+1)$-dimensional extended state space* as

$$\underline{\hat{x}}'_m(t)\,\underline{\hat{\alpha}} = 0 \tag{3-37}$$

and the model's pseudo IC vector

$$\underline{\hat{x}}'_{m_0} = [\ -1 \ \vdots \ \underline{0}' \ \vdots \ \alpha_0\ ] \tag{3-38}$$

which correspond to definitions (3-28) and (3-29) respectively. The model's time response trajectory lies in its characteristic plane ($\underline{\hat{\alpha}}$-plane) in the $(\ell+1)$-dimension space. Since the system's trajectory lies in the $\underline{\bar{a}}$-plane in the (n+1)-dimension space, only its projection into the model's extended state space can be considered for approximating the model's trajectory.

Let $\underline{\hat{x}}(t)$ be the projection of the system's trajectory in the $(\ell+1)$-space, then

$$\underline{\hat{x}}(t) = H\underline{\bar{x}}(t) \tag{3-39}$$

where H is an $(\ell+1) \times (n+1)$ matrix defined as

---

* The ^ denotes a vector in the $(\ell+1)$-dimensional extended state space of the model when $\ell < n$.

$$H = [\, I \;\vdots\; O \,] \tag{3-40}$$

I and O are appropriately dimensioned identity and null matrices respectively. From the geometrical viewpoint the design problem in this case is to select the orientation of the $\tilde{\underline{a}}$-plane, by means of $\underline{p}$, such that the projection of the system's trajectory, $\hat{\underline{x}}(t)$, approximates the model's trajectory, $\hat{\underline{x}}_m(t)$. Mathematically, it would be possible to match the model's trajectory, except for an arbitrarily small region around $\hat{\underline{x}}_{m_0}$, by $\hat{\underline{x}}(t)$ if sufficient freedom is allowed in selecting $\underline{p}$ to make certain elements of $\tilde{\underline{a}}$ go to infinity. This statement will be verified later. It should be clear that $\hat{\underline{x}}(t)$ would have to lie within the $\hat{\underline{\alpha}}$-plane in order to match $\hat{\underline{x}}_m(t)$ identically. It will also be shown subsequently that if $\hat{\underline{x}}(t)$ lies in the $\hat{\underline{\alpha}}$-plane for all time greater than zero, it must coincide with $\hat{\underline{x}}_m(t)$ except for an arbitrarily small region around $\hat{\underline{x}}_{m_0}$. Assuming these statements to be true it would again be reasonable to expect that if the projection of the system's trajectory in the $(\ell+1)$-space could be made to lie close to the $\hat{\underline{\alpha}}$-plane it would be close to the model's trajectory. It then follows that the criterion used to establish the Model PI for the case where $\ell = n$ is also valid here.

The instantaneous distance from $\hat{\underline{x}}(t)$ to the $\hat{\underline{\alpha}}$-plane is given by

$$\frac{\| \hat{\underline{x}}'(t)\hat{\underline{\alpha}} \|}{\| \hat{\underline{\alpha}} \|} \tag{3-41}$$

One could use the square of (3-41) to form a performance index similar to (3-35), but by substituting (3-39) in (3-41) and making a new, yet consistent, definition of $\tilde{\underline{\alpha}}$, the same Model PI (3-35) results.

$$\frac{\| \hat{\underline{x}}'(t)\hat{\underline{\alpha}} \|}{\| \hat{\underline{\alpha}} \|} = \frac{\| \underline{x}'(t)H'\hat{\underline{\alpha}} \|}{\| \hat{\underline{\alpha}} \|} \tag{3-42}$$

Define an (n+1)-dimensional vector

$$\tilde{\underline{\alpha}} = H'\hat{\underline{\alpha}} \tag{3-43}$$

Using (3-40) in the above gives

$$\underline{\tilde{\alpha}}' = [\underline{\hat{\alpha}}' \;\vdots\; \underline{0}'\;]$$

or

$$\underline{\tilde{\alpha}}' = [\underline{\alpha}' \;\vdots\; 1 \;\vdots\; \underline{0}'\;] \tag{3-44}$$

where $\underline{0}$ is an $(n-\ell)$-dimension null vector. This definition is consistent with the earlier use of $\underline{\tilde{\alpha}}$ when $\ell = n$. The null vector in (3-44) adds nothing to the length of $\underline{\hat{\alpha}}$ so that

$$\|\underline{\hat{\alpha}}\| = \|\underline{\tilde{\alpha}}\| \tag{3-45}$$

Using this new definition for $\underline{\tilde{\alpha}}$ (3-43) in (3-42) gives

$$\frac{\|\underline{\hat{x}}'(t)\underline{\hat{\alpha}}\|}{\|\underline{\hat{\alpha}}\|} = \frac{\|\underline{\tilde{x}}'(t)\underline{\tilde{\alpha}}\|}{\|\underline{\tilde{\alpha}}\|} \tag{3-46}$$

which is exactly (3-33). Therefore (3-33) is the instantaneous distance from the projection of the system's trajectory into the $(\ell+1)$-space to the model's characteristic plane for $\ell \leq n$. The Model PI (3-35) with $\underline{\tilde{\alpha}}$ defined by (3-44) is valid for models of order less than or equal to the system's order.

It still remains to verify the two statements assumed to be true in the foregoing development. To establish these points, it is useful to introduce the notion of extending the $\underline{\hat{\alpha}}$-plane, which was defined in the $(\ell+1)$-space, into the $(n+1)$-space. The new definition for $\underline{\tilde{\alpha}}$ (3-44) provides a direct means for doing this.

Assume $\ell < n$, then the expression

$$\underline{\tilde{x}}'_m(t)\underline{\tilde{\alpha}} = 0 \tag{3-47}$$

defines a plane ($\underline{\tilde{\alpha}}$-plane) in the $(n+1)$-space that is coincident with the $\underline{\hat{\alpha}}$-plane in the $(\ell+1)$-space. A plane is defined as being coincident with a second plane in a lower dimensional space if the projection of an arbitrary vector in the first plane into the lower dimensional space lies in the second plane. This is easily verified for the $\underline{\tilde{\alpha}}$-plane and the

$\underline{\hat{\alpha}}$-plane. Let $\underline{\tilde{v}}$ be an arbitrary vector in the $\underline{\tilde{\alpha}}$-plane, then

$$\underline{\tilde{v}}'\underline{\tilde{\alpha}} = 0 \tag{3-48}$$

Using the definition of $\underline{\tilde{\alpha}}$ in (3-48) gives

$$\underline{\tilde{v}}'H\underline{\hat{\alpha}} = 0 \tag{3-49}$$

which means that the vector $H\underline{\tilde{v}}$, which is the projection of $\underline{\tilde{v}}$ into the $(\ell+1)$-space, lies in the $\underline{\hat{\alpha}}$-plane.

The statements in question can be verified by showing that $\underline{\hat{x}}(t)$ coincides in the limit with $\underline{\hat{x}}_m(t)$, except for an arbitrarily small region near $\underline{\hat{x}}_{m_0}$, if and only if the $\underline{\tilde{a}}$-plane coincides in the limit with the $\underline{\tilde{\alpha}}$-plane as certain elements of $\underline{\tilde{a}}$ go to infinity. When $\ell < n$ the $\underline{\tilde{a}}$-plane can not coincide with $\underline{\tilde{\alpha}}$-plane for any finite $\underline{\tilde{a}}$ because $\underline{\tilde{a}}$ has to be co-linear with $\underline{\tilde{\alpha}}$ for the planes to coincide, i. e.

$$\underline{\tilde{a}} = c\underline{\tilde{\alpha}}$$

or

$$\frac{1}{c}\left[\underline{\hat{a}}' \;\vdots\; a_{\ell+1} \;\; a_{\ell+2} \cdots a_{n-1} \;\; 1\right] = \left[\underline{\hat{\alpha}}' \;\vdots\; \underline{0}'\right] \tag{3-50}$$

and in order to satisfy (3-50) c must go to infinity, the elements of $\underline{\hat{a}}$ must go to infinity in such a way that

$$\lim_{c\to\infty}\left[\frac{1}{c}\underline{\hat{a}}\right] = \underline{\hat{\alpha}} \tag{3-51}$$

and the remaining elements of $\underline{\tilde{a}}$ must satisfy

$$\lim_{c\to\infty}\left[\frac{1}{c}a_i\right] = 0 \qquad \text{for } i = \ell+1,\ \ell+2,\ \cdots,\ n-1 \tag{3-52}$$

Then $\underline{\tilde{a}}$ must satisfy

$$\lim_{c\to\infty}\left[\frac{1}{c}\underline{\tilde{a}}\right] = \underline{\dot{\alpha}} \tag{3-53}$$

The elements of $\hat{\underline{a}}$ are the first $(\ell+1)$ elements of $\tilde{\underline{a}}$, which are functions of $\underline{p}$. Whenever necessary in the following proof it is assumed that sufficient freedom is allowed in selecting $\tilde{\underline{a}}$, via $\underline{p}$, to either make the $\tilde{\underline{a}}$-plane arbitrarily close to the $\tilde{\underline{\alpha}}$-plane or coincide in the limit with the $\tilde{\underline{\alpha}}$-plane.

The sufficiency condition will be considered first. As the $\tilde{\underline{a}}$-plane is made arbitrarily close to the $\tilde{\underline{\alpha}}$-plane the system's trajectory, $\tilde{\underline{x}}(t)$, becomes arbitrarily close to the $\tilde{\underline{\alpha}}$-plane. Since the $\tilde{\underline{\alpha}}$-plane coincides with the $\hat{\underline{\alpha}}$-plane in $(\ell+1)$-space the projection of $\tilde{\underline{x}}(t)$ into $(\ell+1)$-space, $\hat{\underline{x}}(t)$, becomes arbitrarily close to the $\hat{\underline{\alpha}}$-plane, except for $\hat{\underline{x}}(\tau)$ where $0 \leq \tau < \epsilon$ and ϵ is some small positive number. The projection of the system's pseudo IC vector, $\tilde{\underline{x}}_0$, into $(\ell+1)$-space, $\hat{\underline{x}}_0$, is generally not close to the $\hat{\underline{\alpha}}$-plane. This is easily shown by evaluating $\hat{\underline{x}}_0$, i. e.

$$\hat{\underline{x}}'_0 = \tilde{\underline{x}}'_0 H' = \left[-1 \;\vdots\; \underline{0}' \;\vdots\; a_0 \right] \begin{bmatrix} I \\ \hline O \end{bmatrix} = \left[-1 \;\vdots\; \underline{0}' \right] \tag{3-54}$$

where the null vectors are appropriately dimensioned. Then

$$\hat{\underline{x}}'_0 \hat{\underline{\alpha}} = -\alpha_0 \neq 0 \tag{3-55}$$

so that $\hat{\underline{x}}_0$ is only close to the $\hat{\underline{\alpha}}$-plane if α_0 is very small. Therefore the $\hat{\underline{x}}(t)$ has to travel from $\hat{\underline{x}}_0$ to a region arbitrarily close to the $\hat{\underline{\alpha}}$-plane during the time interval $[0, \epsilon)$.

Before proceeding with the proof it is helpful to illustrate several of the points made thus far for the simple case of a second order system (n = 2) and a first order model (ℓ = 1). Figure 3-2 shows a sequence of sketches in which the system's characteristic plane, $\tilde{\underline{a}}$-plane, becomes successively closer, going from part a to part c, to the extended characteristic plane of the model, $\tilde{\underline{\alpha}}$-plane. In this case the model's "characteristic plane" is actually a line in the two-dimensional space. Its trajectory, $\hat{\underline{x}}_m(t)$, lies along this line starting at $\hat{\underline{x}}_{m_0}$ and receding into the origin with time. The $\tilde{\underline{\alpha}}$-plane is clearly shown to coincide with the $\hat{\underline{\alpha}}$-plane in the two-dimensional space. The

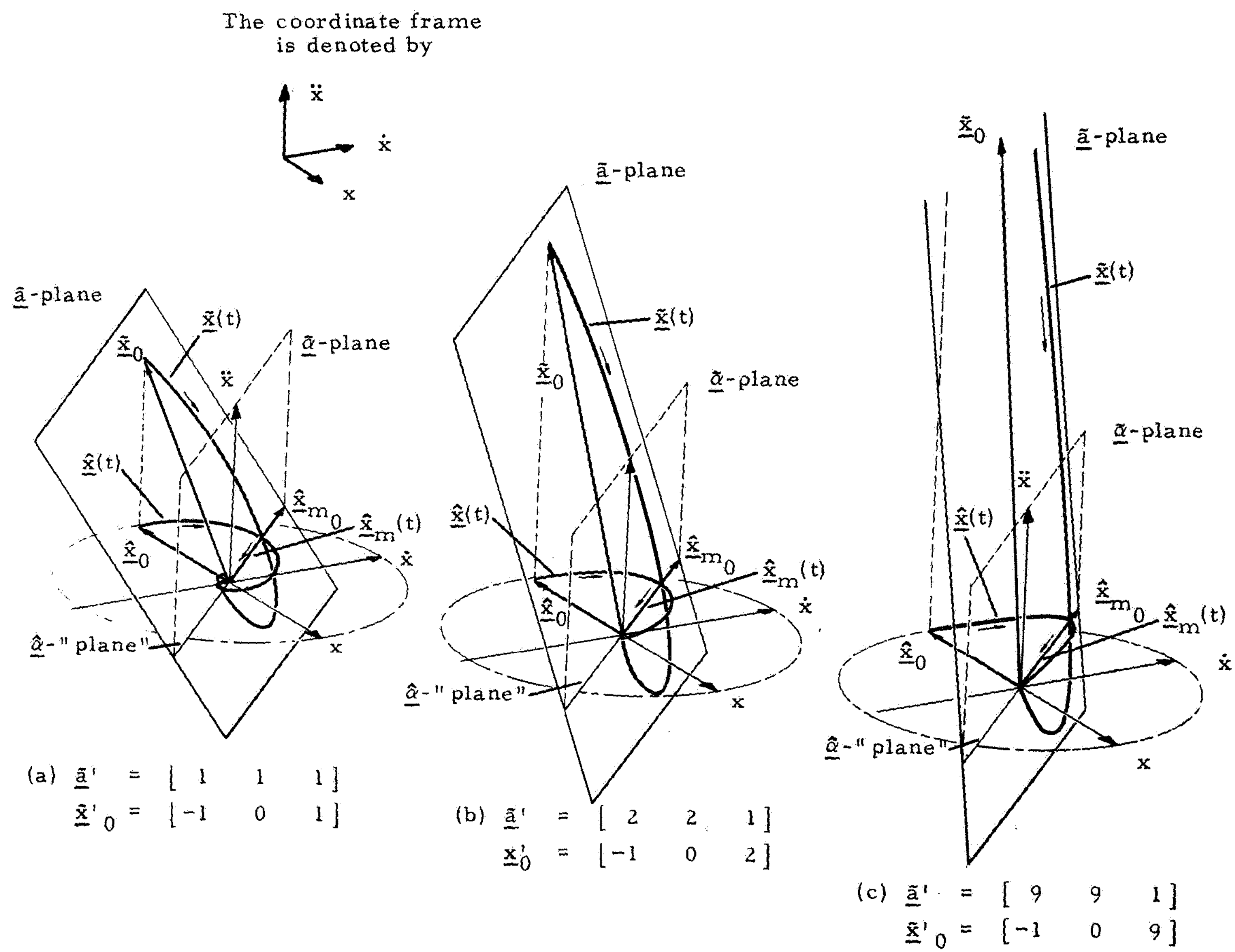


Figure 3-2 Geometrical Representation of a System Approximating a Lower Order Model, $\underline{\hat{\alpha}}' = [1 \quad 1]$, $\underline{\hat{x}}'_{m_0} = [-1 \quad 1]$

system's trajectory, $\underline{\tilde{x}}(t)$, is shown lying in the $\underline{\tilde{a}}$-plane in the three-dimensional space. Its projection into the two-dimensional space, $\underline{\hat{x}}(t)$, starts at $\underline{\hat{x}}_0$ and travels towards the $\underline{\hat{\alpha}}$-plane (line). Comparing $\underline{\hat{x}}(t)$ in parts a, b, and c shows that as the $\underline{\tilde{a}}$-plane becomes closer to the $\underline{\tilde{\alpha}}$-plane the $\underline{\hat{x}}(t)$ becomes closer to the $\underline{\hat{\alpha}}$-plane except for the portion required to go from $\underline{\hat{x}}_0$ to the $\underline{\hat{\alpha}}$-plane. It is in the region of the $\underline{\hat{\alpha}}$-plane near $\underline{\hat{x}}_{m_0}$ that $\underline{\hat{x}}(t)$ is not close to the $\underline{\hat{\alpha}}$-plane. One can also see that the time interval required to go from $\underline{\hat{x}}_0$ to the $\underline{\hat{\alpha}}$-plane decreases as the $\underline{\tilde{a}}$-plane approaches the $\underline{\tilde{\alpha}}$-plane.

Now proceeding with the proof, consider the limiting case as the $\underline{\tilde{a}}$-plane coincides with the $\underline{\tilde{\alpha}}$-plane. In the limit, $\underline{\tilde{x}}(t)$ then lies in the $\underline{\tilde{\alpha}}$-plane and $\underline{\hat{x}}(t)$ must lie in the $\underline{\hat{\alpha}}$-plane. But $\underline{\hat{x}}_0$ is not in the $\underline{\hat{\alpha}}$-plane, so $\underline{\hat{x}}(t)$ is discontinuous at $t = 0$ ($\epsilon \to 0$). By considering the system's trajectory in the limit, one can show that only the last element of $\underline{\hat{x}}(t)$ is discontinuous. Consider the partitioned form of the system's trajectory equation

$$\underline{\hat{x}}'(t)\underline{\hat{a}} = - a_{\ell+1}x^{(\ell+1)}(t) - a_{\ell+2}x^{(\ell+2)}(t) - \cdots - x^{(n)}(t) \tag{3-56}$$

Multiplying (3-56) by $\frac{1}{c}$, taking the limit as $c \to \infty$, and using (3-51) and (3-52), gives on the left hand side

$$\lim_{c \to \infty} \left[\frac{1}{c}\,\underline{\hat{x}}'(t)\,\underline{\hat{a}}\right] = \underline{\hat{x}}'(t)\,\underline{\hat{\alpha}} \tag{3-57}$$

and zero on the right hand side. Thus in the limit, the system's trajectory equation reduces to

$$\underline{\hat{x}}'(t)\underline{\hat{\alpha}} = 0 \tag{3-58}$$

or

$$x^{(\ell)}(t) = - \alpha_0 x(t) - \alpha_1 \dot{x}(t) - \cdots - \alpha_{\ell-2}x^{(\ell-2)}(t) - \alpha_{\ell-1}x^{(\ell-1)}(t) \tag{3-58a}$$

Using the first ℓ initial conditions in (3-54) for the variables on the right hand side of (3-58a) gives $x_0^{(\ell)} = \alpha_0$. But $x_0^{(\ell)} = 0$ in (3-54) so that $x^{(\ell)}(t)$ is discontinuous at t = 0. Let $\underline{\hat{x}}_{0^+}$ denote the value of $\underline{\hat{x}}(t)$ at the discontinuity when approached from the positive time direction. Then

$$\underline{\hat{x}}'_{0^+} = [-1 \mid \underline{0}' \mid \alpha_0] \tag{3-59}$$

and $\underline{\hat{x}}(t)$, given by (3-58) and (3-59), is continuous over the time interval $(0, \infty)$ and lies entirely in the $\underline{\hat{\alpha}}$-plane.

The model's trajectory, given by (3-37) and (3-38), is continuous at t = 0 so that

$$\underline{\hat{x}}'_{m_{0^+}} = \underline{\hat{x}}'_{m_0} = [-1 \mid \underline{0}' \mid \alpha_0] \tag{3-60}$$

Since $\underline{\hat{x}}(t)$ and $\underline{\hat{x}}_m(t)$ both lie in the $\underline{\hat{\alpha}}$-plane for $t > 0$ and are equal at $t = 0^+$, they must coincide over the time interval $(0, \infty)$. However there is an arbitrarily small region near $\underline{\hat{x}}_{m_0}$ in which $\underline{\hat{x}}(t)$ does not coincide with $\underline{\hat{x}}_m(t)$ due to the discontinuity in $\underline{\hat{x}}(t)$ at t = 0. This verified that $\underline{\hat{x}}(t)$ coincides in the limit with $\underline{\hat{x}}_m(t)$, except for an arbitrarily small region near $\underline{\hat{x}}_{m_0}$, if the $\underline{\bar{a}}$-plane coincides in the limit with the $\underline{\tilde{\alpha}}$-plane.

Proof of the necessary condition is simpler. Assume that $\underline{\hat{x}}(t)$ and $\underline{\hat{x}}_m(t)$ coincide, except for an arbitrarily small region near $\underline{\hat{x}}_{m_0}$, i.e. they coincide over the time interval $(0, \infty)$. Then $\underline{\hat{x}}(t)$ must lie in the $\underline{\hat{\alpha}}$-plane for $t > 0$. Since the $\underline{\tilde{\alpha}}$-plane is coincident with the $\underline{\hat{\alpha}}$-plane in $(\ell+1)$-space, $\underline{\bar{x}}(t)$ must lie in the $\underline{\tilde{\alpha}}$-plane in order for its projection into the $(\ell+1)$-space to lie in the $\underline{\hat{\alpha}}$-plane. This can be proven by contradiction. Assume for the moment that $\underline{\bar{x}}(t)$ has a component normal to the $\underline{\tilde{\alpha}}$-plane, i.e.

$$\underline{\bar{x}}(t) = \underline{\bar{x}}_t(t) + \underline{\bar{x}}_n(t) \tag{3-61}$$

where $\underline{\tilde{x}}_t(t)$ lies in the $\underline{\tilde{\alpha}}$-plane and $\underline{\tilde{x}}_n(t)$ is normal to the $\underline{\tilde{\alpha}}$-plane. The projection of $\underline{\tilde{x}}(t)$ into the $(\ell+1)$-space would be

$$\underline{\hat{x}}(t) = H\underline{\tilde{x}}_t(t) + H\underline{\tilde{x}}_n(t) \tag{3-62}$$

so that

$$\begin{aligned} \underline{\hat{x}}'(t)\,\underline{\hat{\alpha}} &= \underline{\tilde{x}}_t(t)H'\,\underline{\hat{\alpha}} + \underline{\tilde{x}}_n'(t)H'\,\underline{\hat{\alpha}} \\ &= \underline{\tilde{x}}_n'(t)H'\,\underline{\hat{\alpha}} \\ &= \underline{\tilde{x}}_n'(t)\,\underline{\tilde{\alpha}} \end{aligned} \tag{3-63}$$

Since $\underline{\tilde{x}}_n(t)$ is normal to the $\underline{\tilde{\alpha}}$-plane it must be colinear with $\underline{\tilde{\alpha}}$, i. e.

$$\underline{\tilde{x}}_n(t) = f(t)\,\underline{\tilde{\alpha}} \tag{3-64}$$

where f(t) is some nonzero function of time. Using (3-64) in (3-63) gives

$$\underline{\hat{x}}'(t)\,\underline{\hat{\alpha}} = f(t)\,\|\underline{\tilde{\alpha}}\|^2 \neq 0 \tag{3-65}$$

But (3-65) contradicts the statement that $\underline{\hat{x}}(t)$ lies in the $\underline{\hat{\alpha}}$-plane so that the assumption that $\underline{\tilde{x}}(t)$ has a component normal to the $\underline{\tilde{\alpha}}$-plane has to be false. Thus $\hat{x}(t)$ can lie in the $\underline{\hat{\alpha}}$-plane only if $\underline{\tilde{x}}(t)$ lies in the $\underline{\tilde{\alpha}}$-plane; so $\underline{\tilde{x}}(t)$ must lie in the $\underline{\tilde{\alpha}}$-plane. It follows that the $\underline{\tilde{a}}$-plane must either coincide with the $\underline{\tilde{\alpha}}$-plane or $\underline{\tilde{x}}(t)$ lies along the intersection of the $\underline{\tilde{a}}$-plane and the $\underline{\tilde{\alpha}}$-plane. From this point the proof is the same as that presented earlier for the necessary condition in the case where $\ell = n$ (see page 41). The pseudo IC vector to use in this case is $\underline{\tilde{x}}_{0^+}$ where the first $(\ell+1)$ elements are the same as $\underline{\hat{x}}_{0^+}$ (3-59), because only the time interval $(0, \infty)$ is considered. The remaining details will not be repeated here. This verifies that $\underline{\hat{x}}(t)$ coincides in the limit with $\underline{\hat{x}}_m(t)$, except for an arbitrarily small region near $\underline{\hat{x}}_{m_0}$, <u>only if</u> the $\underline{\tilde{a}}$-plane coincides in the limit with the $\underline{\tilde{\alpha}}$-plane.

Q. E. D.

The geometrical approach used here to motivate the Model PI may not be appealing to pragmatic control system engineers, particularly since it becomes rather abstract for systems of more than second order. It is difficult to visualize hyperplanes in four-dimensional or higher dimensional spaces and to relate the system's trajectory to one of these representing the model. One would like a more physical interpretation of the Model PI. Two alternate interpretations are given below that provide additional insight to the Model PI.

The vector error between the system and model time responses in the model's extended state space is given by

$$\underline{\hat{e}}(t) = \underline{\hat{x}}(t) - \underline{\hat{x}}_m(t) \qquad (3\text{-}66)$$

A general quadratic performance index similar to ones often appearing in modern control theory is

$$J = \int_0^{\infty} \| \underline{\hat{e}}(t) \|_Q^2 \, dt \qquad (3\text{-}67)$$

where Q is taken to be diagonal for lack of any reason to choose otherwise. However if Q is chosen to be

$$Q = H'\tilde{Q}H \qquad (3\text{-}68)$$

where $\tilde{Q}$ is given by (3-36) then this performance index is equivalent to the Model PI (3-35). This can be shown by writing out the integrand of (3-67) as

$$\begin{aligned}
\| \underline{\hat{e}}(t) \|_{H'\tilde{Q}H}^2 &= \underline{\hat{e}}'(t)H'\tilde{Q}H\underline{\hat{e}}(t) = \underline{\hat{e}}'(t) \left(\frac{H'\underline{\tilde{\alpha}}\underline{\tilde{\alpha}}'H}{\|\underline{\tilde{\alpha}}\|^2} \right) \underline{\hat{e}}(t) \\
&= \underline{\hat{e}}'(t) \left(\frac{\underline{\hat{\alpha}}\underline{\hat{\alpha}}'}{\|\underline{\hat{\alpha}}\|^2} \right) \underline{\hat{e}}(t) = \left(\frac{\underline{\hat{e}}'(t)\underline{\hat{\alpha}}}{\|\underline{\hat{\alpha}}\|} \right)^2 \qquad (3\text{-}69) \\
&= \left(\frac{[\underline{\hat{x}}(t) - \underline{\hat{x}}_m(t)]'\underline{\hat{\alpha}}}{\|\underline{\hat{\alpha}}\|} \right)^2 = \left(\frac{\underline{\hat{x}}'(t)\underline{\hat{\alpha}}}{\|\underline{\hat{\alpha}}\|} - \frac{\underline{\hat{x}}_m'(t)\underline{\hat{\alpha}}}{\|\underline{\hat{\alpha}}\|} \right)^2
\end{aligned}$$

but $\underline{\hat{x}}'_m(t)\,\underline{\hat{\alpha}} = 0$, so that

$$\|\underline{\hat{e}}(t)\|^2_{H'\tilde{Q}H} = \left(\frac{\underline{\hat{x}}'(t)\,\underline{\hat{\alpha}}}{\|\underline{\hat{\alpha}}\|}\right)^2 = \left(\frac{\underline{\tilde{x}}'(t)H'\,\underline{\hat{\alpha}}}{\|\underline{\hat{\alpha}}\|}\right)^2$$

$$= \left(\frac{\underline{\tilde{x}}'(t)\,\underline{\tilde{\alpha}}}{\|\underline{\tilde{\alpha}}\|}\right)^2 = \|\underline{\tilde{x}}(t)\|^2_{\tilde{Q}} \qquad \text{(3-69a)}$$

Therefore one can interpret the Model PI as a weighted integral quadratic error performance index for a special weighting matrix. Minimizing the Model PI corresponds to minimizing a certain weighted combination of the error and the first ℓ of its derivatives (ℓ is the order of the model).

The above development also has a special connotation in the geometrical approach. The term

$$\frac{\underline{\hat{e}}'(t)\,\underline{\hat{\alpha}}}{\|\underline{\hat{\alpha}}\|}$$

in equation (3-69) is the component of $\underline{\hat{e}}(t)$ normal to the model's characteristic plane, which is the same as the distance from the projection of the system's trajectory into the model's extended state space to the $\underline{\hat{\alpha}}$-plane, which is also the same as the distance from the system's trajectory to the $\underline{\tilde{\alpha}}$-plane. Minimizing the Model PI can then be viewed as minimizing the error normal to the $\underline{\hat{\alpha}}$-plane.

The most direct physical explanation of the Model PI is not rigorous but quite pragmatic. The model, written as an autonomous system

$$x_m^{(\ell)} + \alpha_{\ell-1}x_m^{(\ell-1)} + \cdots + \alpha_2\ddot{x}_m + \alpha_0\dot{x}_m + \alpha_0 x_m = 0 \qquad \text{(3-70)}$$

with appropriate pseudo IC, can be interpreted as representing the relationship one would like to exist among the first $\ell+1$ state variable of the control system, that is, that this sum of terms be zero. Assuming this view, a logical performance index to use for trying to force the control system state variables into this linear relationship is

$$PI = \int_0^{\infty} [x^{(\ell)} + \alpha_{\ell-1} x^{(\ell-1)} + \cdots + \alpha_2 \ddot{x} + \alpha_1 \dot{x} + \alpha_0 x]^2 dt \tag{3-71}$$

Note that the integrand is the square of (3-70) with the system's state variables replacing those of the model. The control system state variables must of course satisfy the system equation, which is repeated here

$$x^{(n)} + a_{n-1} x^{(n-1)} + \cdots + a_2 \ddot{x} + a_1 \dot{x} + a_0 x = 0 \tag{3-5}$$

but the coefficients are functions of the free parameters, $\underline{p}$. If $\underline{p}$ is chosen to minimize (3-71) subject to the constraint (3-5), then the system state variables can be said to respond in the best linear relationship to approximate the model, in the integral squared sense. The performance index (3-42) differs from the Model PI only by a constant factor $\|\underline{\tilde{\alpha}}\|^2$ and thus is effectively the same.

Mathematically, the Model PI (3-35) is the square of a linear function space norm of the distance from the system's trajectory to the model's characteristic plane. Physically, it represents a desire to force the control system's output and its derivatives to respond to a step input with the same linear relationship as the model. Historically, it is a quadratic cost functional similar to ones used extensively in modern control theory without any previous physical interpretation or solid justification. The nice mathematical characteristics of positive definite or semi-definite quadratic cost functionals have been the prime motivation for many theorists. Some interesting properties of the Model PI are developed in Chapter 7 that allow one to determine an equivalent Model PI for general quadratic cost functionals. It is then possible to give a physical interpretation to some of the strictly mathematical examples that have appeared in modern control literature. An example of this is also presented in Chapter 7.

The next point to consider briefly is whether the Model PI gives satisfactory results for at least simple academic examples. It is possible to obtain an analytical solution to the Model PI design of a general second order system with no zeros for zero, first, and second order models. The detail formulation and solution are presented in Appendix A. The results for two cases are shown here and interpreted in the geometrical sense as well as by root locus and step response.

Example 3-1

Consider the simple feedback control system in figure 3-3, where k is a free design parameter. The object is to select the gain, k, that makes the closed loop response to a unit step input approximate that of the first order model shown.

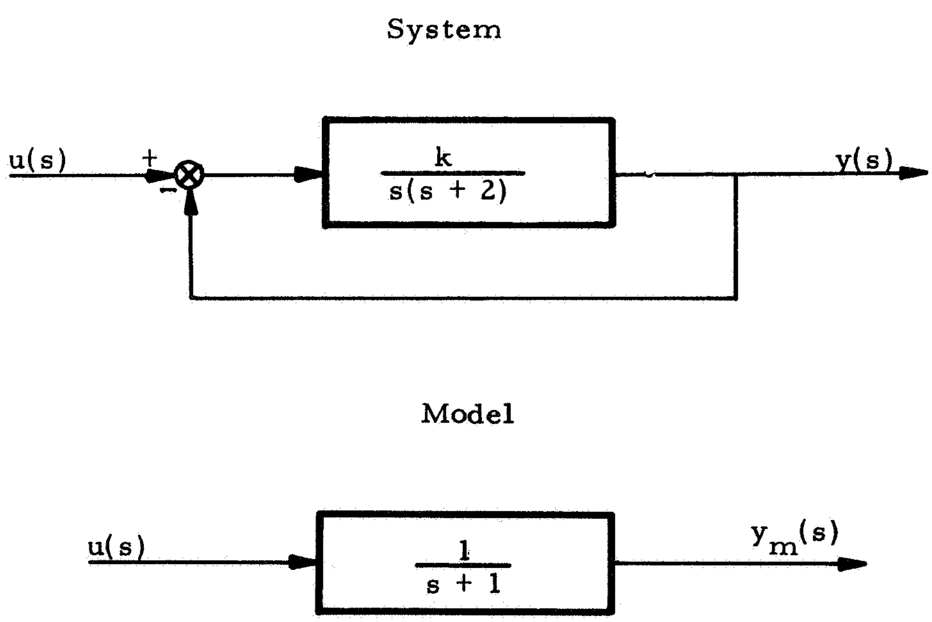


Figure 3-3 Block Diagram of System and Model for Example 3-1

The autonomous system representation of the closed loop control system transient response is

$$\ddot{x} + 2\dot{x} + kx = 0 \tag{3-72}$$

with $x_0 = -1$, $\dot{x}_0 = 0$. The Model PI corresponding to the first order model is

$$PI = \int_0^\infty \left(\frac{\underline{\bar{x}}'\underline{\bar{\alpha}}}{\|\underline{\bar{\alpha}}\|} \right)^2 dt = \frac{1}{2}\int_0^\infty \left([x\ \dot{x}\ \ddot{x}] \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \right)^2 dt$$

or

$$PI = \frac{1}{2}\int_0^\infty (x + \dot{x})^2\, dt \tag{3-73}$$

This example is of the same form as Case 1b in Appendix A and the value of k that minimizes (3-73) can be seen from equation (A-10) to be k = 2. 0.

The geometrical interpretation of this solution is shown in figure 3-4. The extension of the model's characteristic planes is normal to

$$\underline{\bar{\alpha}}' = [\,1 \quad 1 \quad 0\,] \tag{3-74}$$

and hence perpendicular to the $x\dot{x}$-plane. The system's characteristic plane is normal to

$$\underline{\bar{a}}' = [\,2 \quad 2 \quad 1\,] \tag{3-75}$$

The general $\underline{\bar{a}}$ for this example is

$$\underline{\bar{a}}' = [\,k \quad 2 \quad 1\,] \tag{3-76}$$

so that only the first element was free to be selected. Since the model is of lower order than the system, the projection of the system's trajectory on the $x\dot{x}$-plane should be compared to the model's trajectory as mentioned earlier.

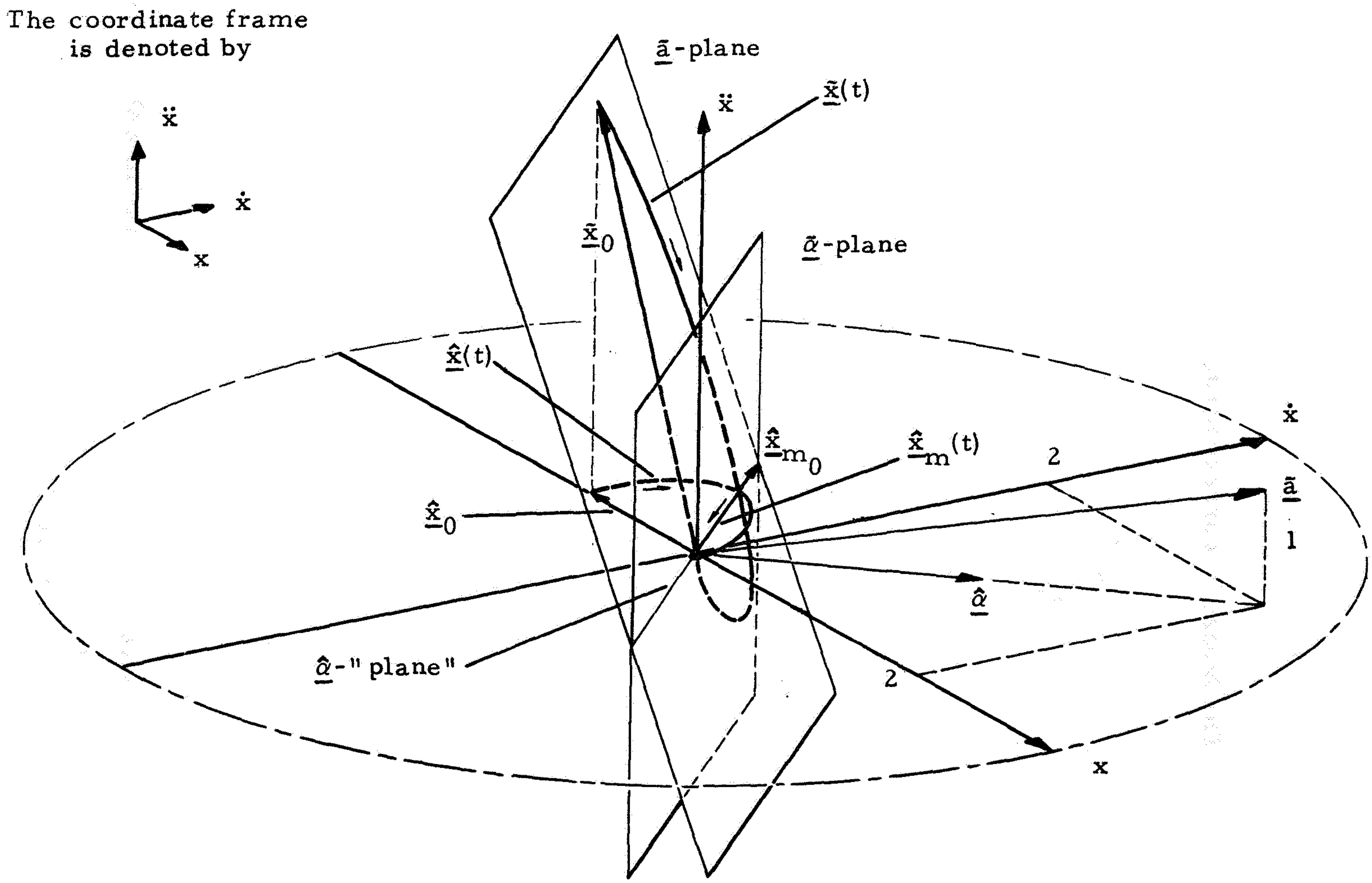


Figure 3-4 Geometrical Representation of Model PI Solution for Example 3-1

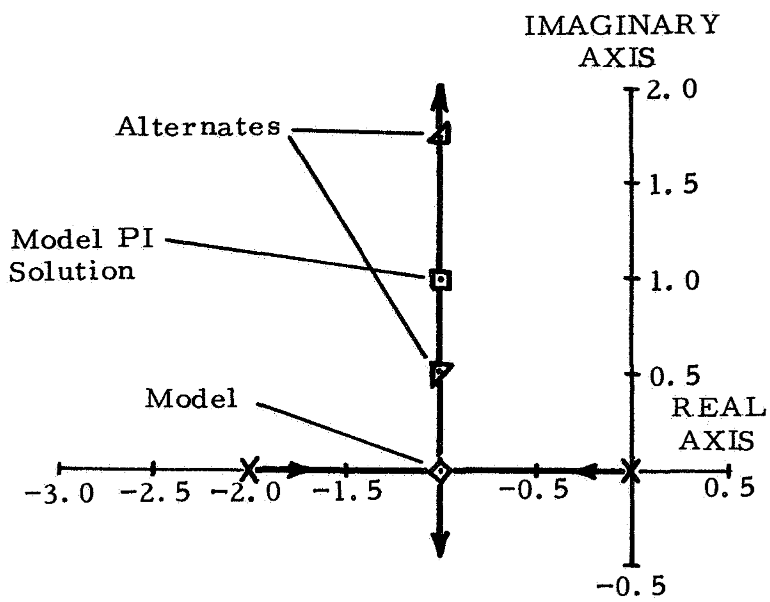


Figure 3-5 Root Locus for Example 3-1

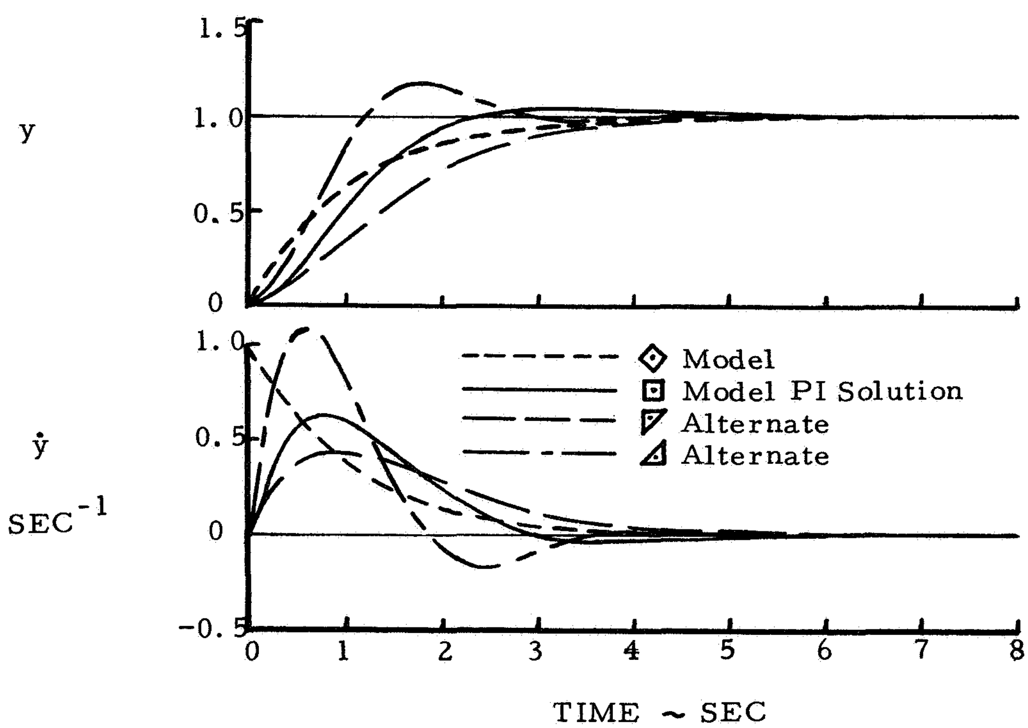


Figure 3-6 Step Responses for Example 3-1

Discussion of the solution is more meaningful if alternate solutions are compared. The root locus is shown in figure 3-5 with the Model PI solution and two alternate solutions denoted. The step responses for these three solutions are compared to that of the model in figure 3-6. For this simple example the Model PI is clearly the best approximation to the model of the three solutions considered. The more general treatment in Appendix A shows that the Model PI solution does approach the model if the coefficient a_1 is very large.

Example 3-2

Consider the same feedback system as example 3-1 but use the second order model in figure 3-7.

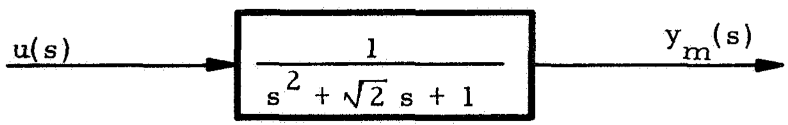


Figure 3-7 Block Diagram of Model for Example 3-2

The corresponding Model PI is

$$PI = \frac{1}{4}\int_0^{\infty} \left([x\ \dot{x}\ \ddot{x}] \begin{bmatrix} 1 \\ \sqrt{2} \\ 1 \end{bmatrix} \right)^2 dt$$

or

$$PI = \frac{1}{4}\int_0^{\infty} (x + \sqrt{2}\,\dot{x} + \ddot{x})^2\, dt \qquad (3\text{-}77)$$

This is a specific example of Case 1c in Appendix A and the solution given by equation (A-13) is k = 1. 26.

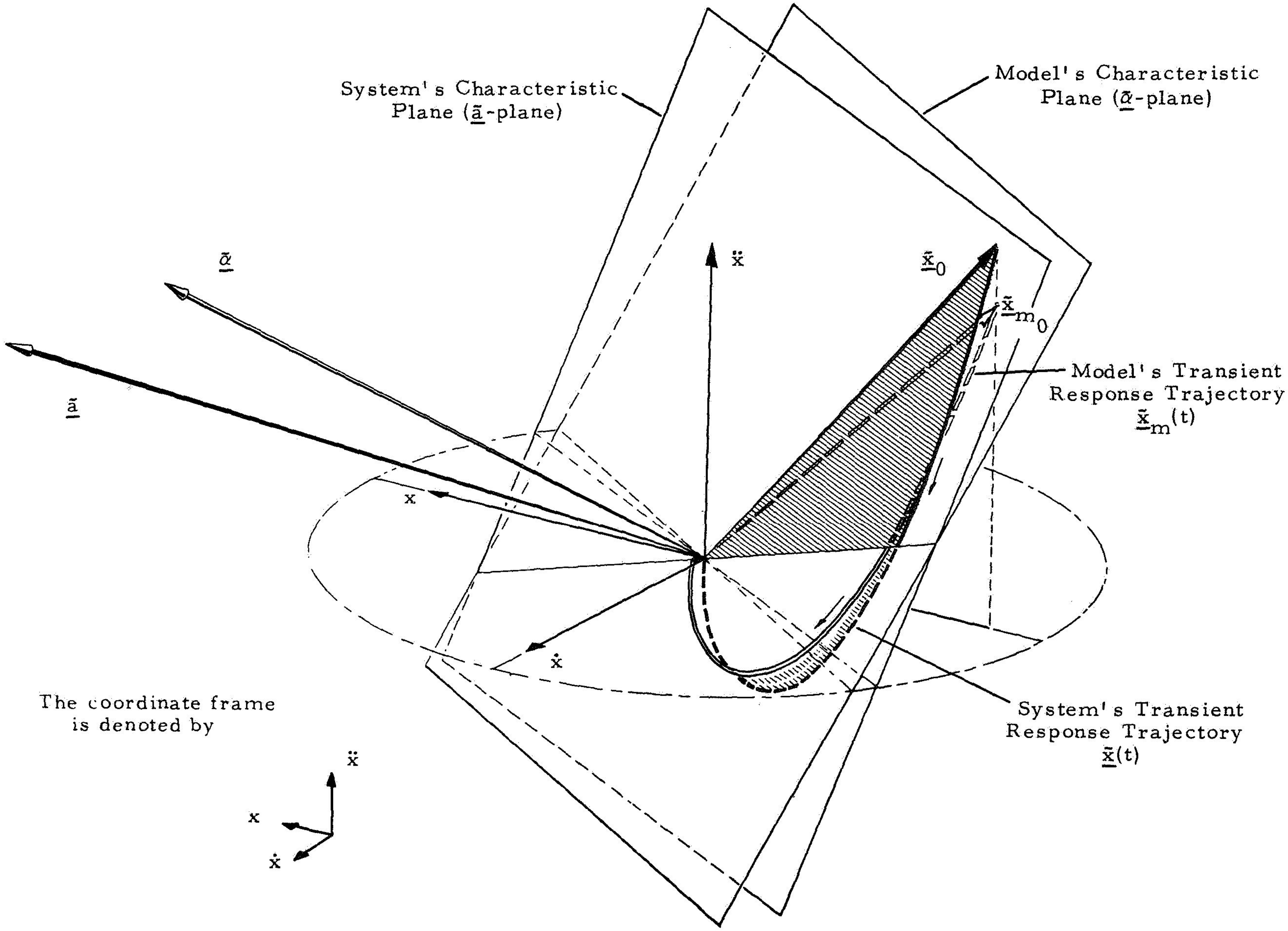


Figure 3-8 Geometrical Representation of Model PI Solution for Example 3-2

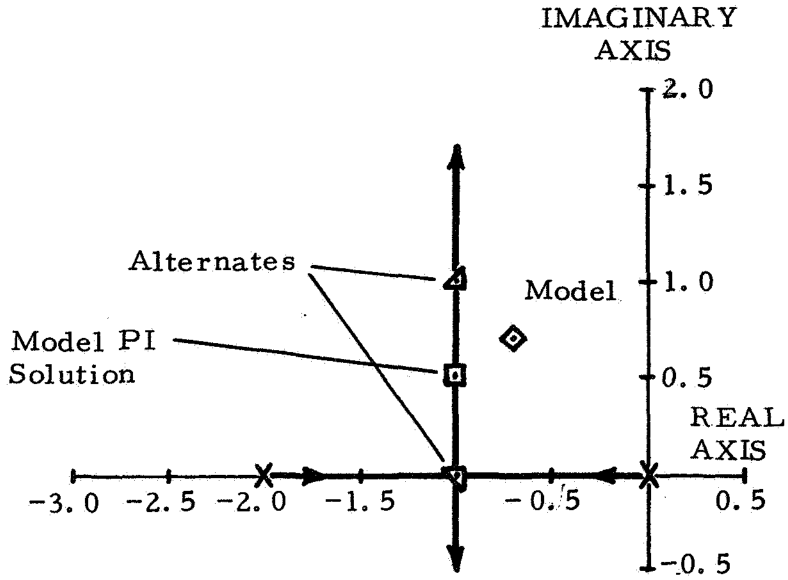


Figure 3-9 Root Locus for Example 3-2

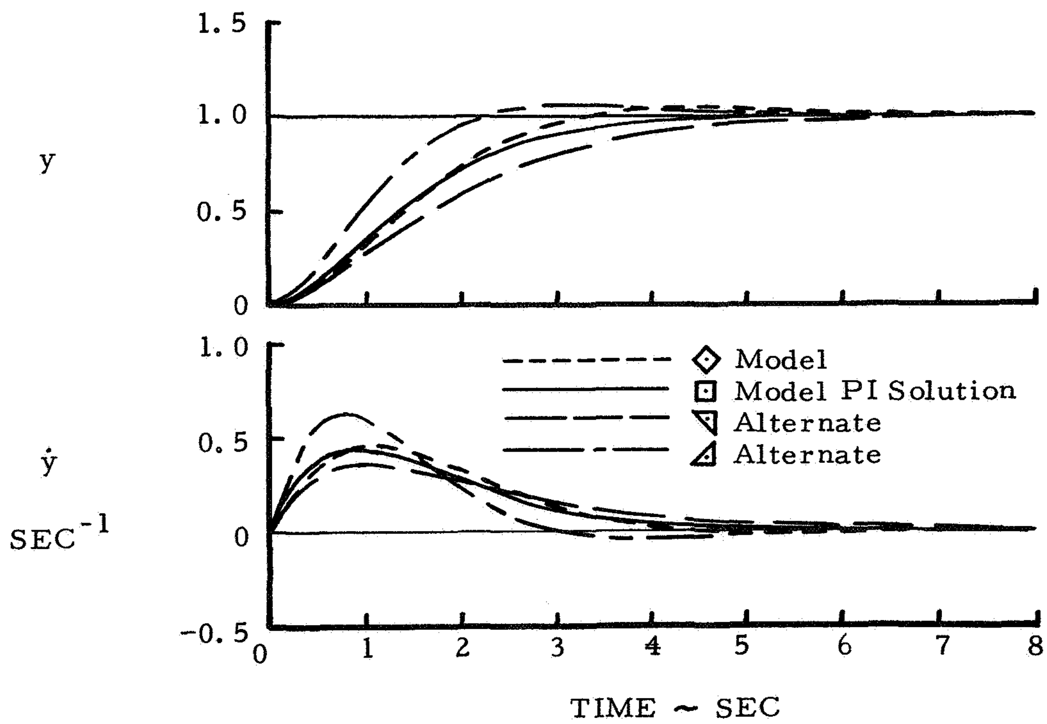


Figure 3-10 Step Responses for Example 3-2

The corresponding geometrical representation is shown in figure 3-8 with the $\underline{\tilde{\alpha}}$-plane and $\underline{\tilde{a}}$-plane normal to

$$\underline{\tilde{\alpha}}' = [\ 1 \quad \sqrt{2} \quad 1\ ] \tag{3-78}$$

and

$$\underline{\tilde{a}}' = [\ 1.26 \quad 2 \quad 1\ ] \tag{3-79}$$

respectively. The trajectories shown lying within the respective characteristic planes compare well in the three-dimensional space. This example was chosen specifically so that it was not possible to make the two planes coincide for any allowable choice of $\underline{\tilde{a}}$ given by (3-76). As indicated in Appendix A, if $a_1 = \sqrt{2}$ then the Model PI solution would give $\underline{\tilde{a}} = \underline{\tilde{\alpha}}$.

The root locus for this example is shown in figure 3-9 with the Model PI solution and two alternate solutions denoted. On the basis of comparing pole locations, the Model PI is a reasonable approximation of the model. The corresponding step responses are compared in figure 3-10. Again in this example the Model PI gives the best approximation of the solutions considered.

One can not generalize at this early point and say that the Model PI always gives the "best" possible solution or even a satisfactory solution when considering practical problems. Before discussing practical applications it is necessary to extend the concept to a larger class of control systems.

3.2.2 Systems with Zeros

The zeros of a system have a very strong effect on its step response. Since the Model PI is based on the transient portion of the step response, it must include this effect to be useful for systems with zeros. The Model PI is extended in this section to the general single input/output system case by referring once again to the geometrical interpretation of autonomous systems in the extended state space.

Recall that the motivation for selecting the Model PI for systems without zeros was to force the system's transient response trajectory to lie close to the model's characteristic plane (see page 43). In that case the first ℓ pseudo IC's of the system and model are always identical, so that matching characteristic planes correspond to matching trajectories. This is not necessarily true for systems with zeros. As mentioned previously in section 3.1, the pseudo IC's contain the effect of the system zeros while the characteristic plane is only dependent on the poles. For a given set of poles, changing the zeros of a system corresponds to picking a different starting point for the trajectory in the characteristic plane. This was illustrated earlier in figure 3-1. A system with zeros and a model with or without zeros would generally not have the same pseudo IC's except in some special cases that are discussed later.

By virtue of the relationship (3-8), the system's pseudo IC's are in general functions of the free design parameters. They are not arbitrary constants as normal initial conditions are usually considered to be. This is a critical point to understand both for extending the Model PI and for deriving an optimization algorithm. The latter is treated in Chapter 4. The design process of selecting the free parameters changes the pseudo IC's as well as the system's characteristic plane. Therefore in the general case it is necessary to add a constraint on the relative location of the system's and model's pseudo IC's. The constraint should be such that if it is satisfied identically, then matching the characteristic planes would again correspond to matching the trajectories.

One such constraint that is convenient to use is to place a quadratic penalty on the error between the pseudo IC vectors projected into the model's ℓ-dimensional state space, i.e.

$$\| \underline{\tilde{x}}_0 - \underline{\tilde{x}}_{m_0} \|^2_{\tilde{W}} \tag{3-80}$$

where

$$\widetilde{W} = \begin{bmatrix} I & | & O \\ - & - & - \\ O & | & O \end{bmatrix} \qquad (3\text{-}81)$$

and I is the $\ell \times \ell$ identity matrix and the O's are appropriately dimensional null matrices. This places constraints only on the first ℓ pseudo IC's, which is sufficient. If (3-80) is identically zero, then matching the characteristic planes would correspond to matching the trajectories, at least in the $(\ell+1)$-dimensional space. Including (3-80) in the performance index penalizes large deviations of $\underline{\tilde{x}}_0' \widetilde{W}$ from $\underline{\tilde{x}}_{m_0}' \widetilde{W}$. It tends to force the first ℓ elements of $\underline{\tilde{x}}_0$ to be near the corresponding model pseudo IC's. Selecting the $\underline{\bar{a}}$-plane, via the free parameters, to make the system's trajectory lie close to the $\underline{\bar{\alpha}}$-plane then corresponds to approximating the model's trajectory.

The general form of the Model PI is thus defined as

$$\boxed{PI = r \| \underline{\tilde{x}}_0 - \underline{\tilde{x}}_{m_0} \|^2_{\widetilde{W}} + \int_0^{\infty} \| \underline{\tilde{x}}(t) \|^2_{\widetilde{Q}} \, dt} \qquad (3\text{-}82)$$

where $\widetilde{W}$ and $\widetilde{Q}$ are given by (3-81) and (3-36) respectively ($\underline{\bar{\alpha}}$ is defined by equation (3-44)). The scalar, r, is selected by the designer to set the relative weighting between matching pseudo IC's and characteristic planes. The form of the Model PI (3-35) defined for systems without zeros is easily shown to be equivalent to (3-82). Consider an nth order system and ℓth order model both without zeros and unity static gain (if they don't have unity static gain, the inputs can be scaled so that they do). From (3-8) the system's pseudo IC's are

$$x_0 = -1$$

$$x_0^{(j)} = 0 \qquad \text{for } j = 1, 2, \cdots (n-1) \qquad (3\text{-}83)$$

and the model's are

$$x_{m_0} = -1$$

$$x_{m_0}^{(j)} = 0 \qquad \text{for } j = 1, 2, \cdots (\ell-1) \tag{3-84}$$

Since $\ell \leq n$, the first ℓ pseudo IC's are the same for the system and model, thus

$$\| \tilde{\underline{x}}_0 - \tilde{\underline{x}}_{m_0} \|^2_{\tilde{W}} \equiv 0 \tag{3-85}$$

for a system and model without zeros.

Sufficient flexibility is provided in the Model PI (3-82) for most single input/output system applications; however the form of the model chosen, i. e. the number of poles and zeros, is important for its most effective use. One simple rule should be followed in selecting the model structure when considering systems with zeros. That is, the model should have the same number of excess poles over zeros as the closed loop control system. If the system has n poles and m zeros and the model has ℓ poles and k zeros, choose a model such that $(\ell - k) = (n-m)$. The rule guarantees that the first ℓ pseudo IC's of the system and model are compatible. Compatibility in this case means that they have the same algebraic form as determined by (3-8), and, if sufficient freedom is allowed in the choice of the system's numerator and denominator polynomial coefficients, they can be numerically equal. An example more clearly illustrates the point. Assume the closed-loop system as five poles and three zeros, and the model has four poles and two zeros, both have two excess poles. The corresponding pseudo IC's as determined by (3-8) are compared in Table 3-1, which shows the first four to be of the same algebraic form. It would be possible to select the a_i and b_j coefficients so that they are numerically equal. On the other hand if the model had three zeros, the system's pseudo IC's would not be compatible with the model's pseudo IC's as illustrated in Table 3-2. It would not be possible for them to be numerically equal. Compatibility then assures that the formulation of the analytical design process meets the conditions that motivated the definition of the general Model PI.

TABLE 3-1

ILLUSTRATION OF COMPATIBLE PSEUDO INITIAL CONDITIONS

Control System Pseudo IC's $n = 5$, $m = 3$	Model Pseudo IC's $\ell = 4$, $k = 2$
$x_0 = -b_0/a_0$	$x_{m_0} = -\beta_0/a_0$
$\dot{x}_0 = 0$	$\dot{x}_{m_0} = 0$
$\ddot{x}_0 = b_3$	$\ddot{x}_{m_0} = \beta_2$
$x_0^{(3)} = b_2 - a_4\ddot{x}_0$	$x_{m_0}^{(3)} = \beta_1 - \alpha_3\ddot{x}_{m_0}$
$x_0^{(4)} = b_1 - a_3\ddot{x}_0 - a_4x_0^{(3)}$	

Even if the system and model pseudo IC's are incompatible, the Model PI may still provide a good solution, depending on the specific situation. For instance, in the example of Table 3-2 (model with $\ell = 4$, $k = 3$) if β_3 were very small compared to β_0, and β_2, it may be possible for the pseudo IC's to be numerically close in value, in which case the Model PI would still be appropriate. However situations involving incompatible pseudo IC's should generally be avoided.

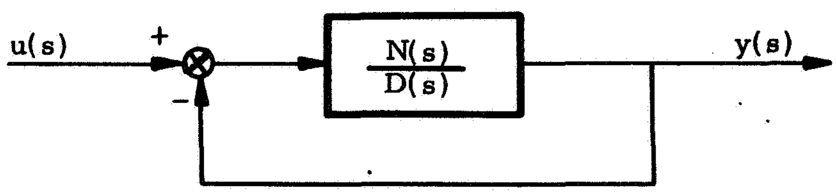
TABLE 3-2

ILLUSTRATION OF INCOMPATIBLE PSEUDO INITIAL CONDITIONS

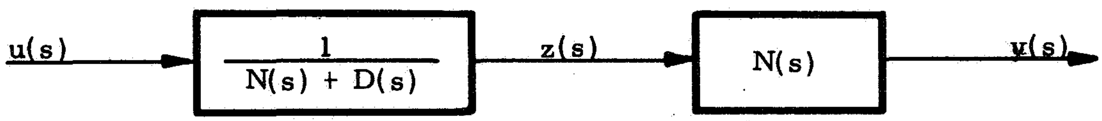
Control System Pseudo IC's $n = 5$, $m = 3$	Model Pseudo IC's $\ell = 4$, $k = 3$
$x_0 = -b_0/a_0$	$x_{m_0} = -\beta_0/\alpha_0$
$\dot{x}_0 = 0$	$\dot{x}_{m_0} = \beta_3$
$\ddot{x}_0 = b_3$	$\ddot{x}_{m_0} = \beta_2 - \alpha_3 \dot{x}_{m_0}$
$x_0^{(3)} = b_2 - a_4 \ddot{x}_0$	$x_{m_0}^{(3)} = \beta_1 - \alpha_2 \dot{x}_{m_0} - \alpha_3 \ddot{x}_{m_0}$
$x_0^{(4)} = b_1 - a_3 \ddot{x}_0 - a_4 x_0^{(3)}$	

There are two types of design problems involving systems with zeros in which the first term of (3-82) is zero and the Model PI reduces to the form defined in section 3.2.1. If the model is selected to have no zeros and the number of poles equal to or less than the number of excess over zeros in the closed-loop system ($\ell \leq n-m$), then the first pseudo IC's for the system and model are equal. The input can always be scaled to make the first pseudo IC's the same. The remaining ones for the model are zero, as are the next (ℓ-1) for the system. Therefore only the original form of the Model PI (3-35) needs to be used.

The second class of problems in which that is true is when only the closed-loop poles of the system have to be considered in the design process. For example, if the design specifications are given completely in terms of the closed-loop poles and the closed-loop zeros are unaffected by the choice of the design parameters, then the system can be designed using the Model PI (3-35), neglecting the pseudo IC's. This approach does not neglect the effect of the open-loop zeros on the closed-loop poles. The simple block diagram sketches in figure 3-11 illustrate the point. Assume the control system has a functional block diagram of the form shown in part a. of figure 3-11. If the design specifications do not include the effect of N(s), then one can design the system based only on the transfer characteristics from u(s) to z(s).



(a) Original Feedback System



b(s) Equivalent Open-Loop System

Figure 3-11 Block Diagram of Equivalent Feedback and Open-Loop Systems

The special cases discussed and the suggestions made for selecting appropriate models do not place any unrealistic restrictions on the use of the Model PI. The one general form of the Model PI (3-82) is valid for all the situations considered, although in the special cases noted it reduces to the simpler form (3-35). The guidelines for choosing the model structure are only minor restrictions and are good guidelines to use for any analytical design technique based on a reference model. Selection of an appropriate model should be given more serious attention in the analytical design process than it has traditionally been given. The designer should consider not only whether the model fits his specifications but also how likely it is that his analytical technique will select a design that closely approximates the model. This topic is discussed more fully in Chapter 5.

It is instructive at this point to consider two simple academic examples for systems with zeros. The main points to be illustrated are the effect of the pseudo IC's and the geometrical interpretation. Since the procedures for minimizing the Model PI are not treated until the next chapter and are not pertinent to the present discussion, those details are omitted and only the results presented.

Example 3-3

Consider the control system shown in figure 3-12 where k_1 and k_2 are free design parameters. The object is to select k_1 and k_2 such that the closed-loop step response approximates that of the model.

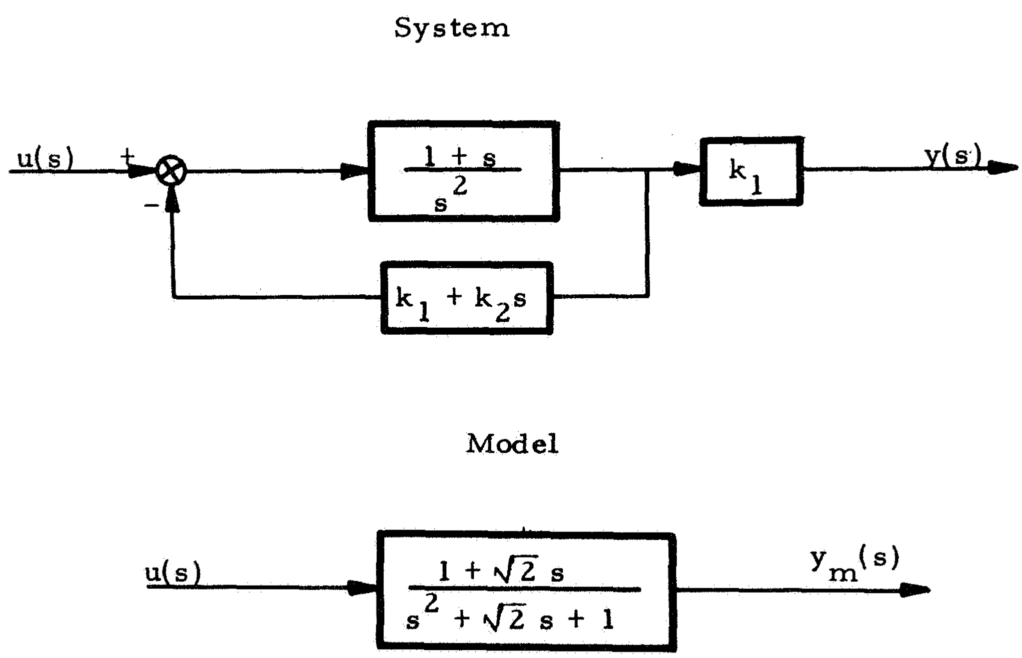


Figure 3-12 Block Diagram of System and Model for Example 3-3

The autonomous system representation of the closed loop system transient response is

$$\ddot{x} + \left[\frac{k_1 + k_2}{1 + k_2}\right]\dot{x} + \left[\frac{k_1}{1 + k_2}\right]x = 0 \qquad (3\text{-}86)$$

with pseudo IC's, $x_0 = -1$, $\dot{x}_0 = \frac{k_1}{1 + k_2}$. The model's pseudo IC's are $x_{m_0} = -1$, $\dot{x}_{m_0} = \sqrt{2}$ so that the corresponding Model PI is

$$PI = r\left[\frac{k_1}{1 + k_2} - \sqrt{2}\right]^2 + \frac{1}{4}\int_0^{\infty} (x + \sqrt{2}\,\dot{x} + \ddot{x})^2\, dt \qquad (3\text{-}87)$$

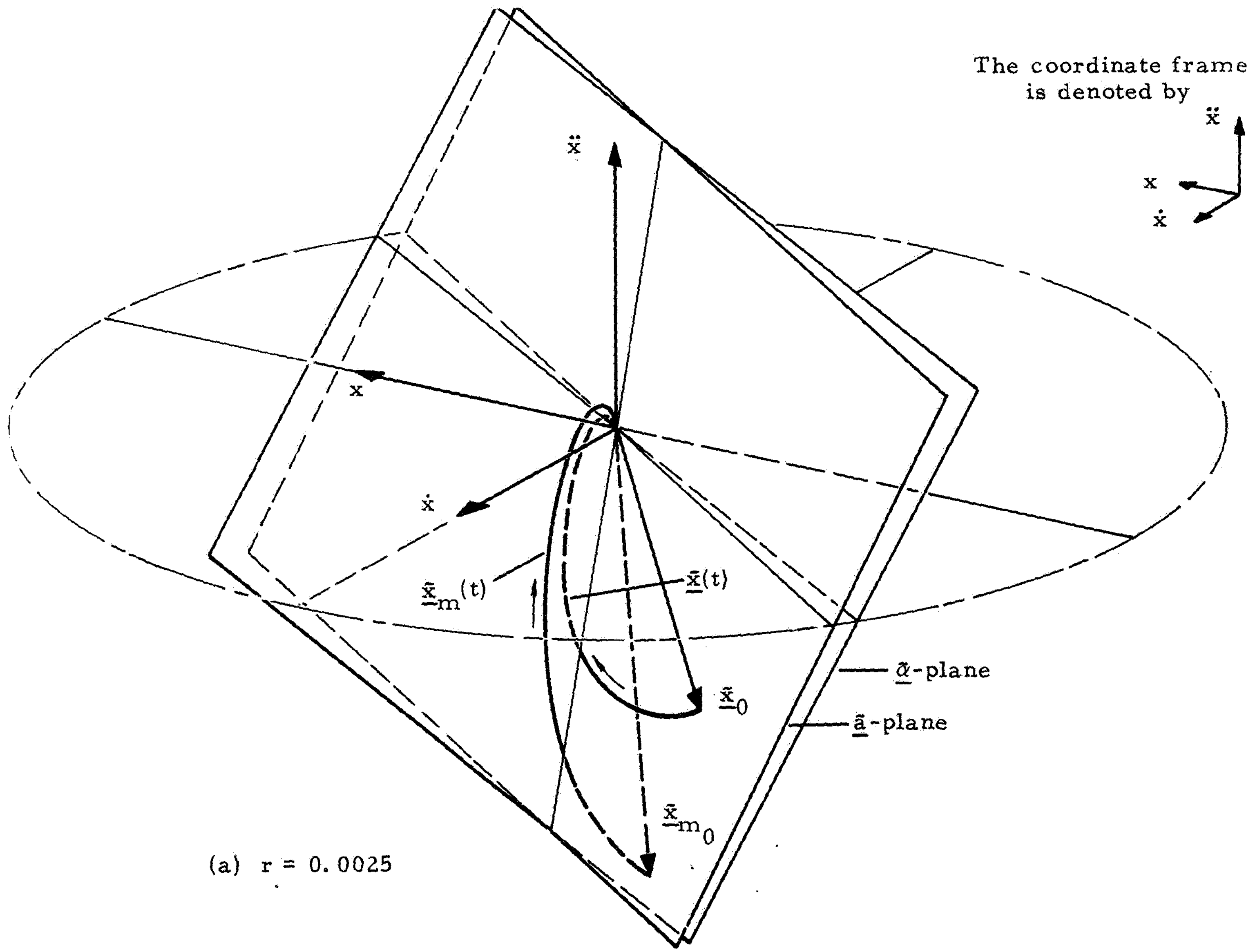
The values of k_1 and k_2 that minimizes (3-87) were computed for three values of the weighting factor, r, and are listed in table 3-3 together with the corresponding values of $\underline{\tilde{a}}$ and $\underline{\tilde{x}}_0$.

TABLE 3-3

NUMERICAL RESULTS FOR EXAMPLE 3-3

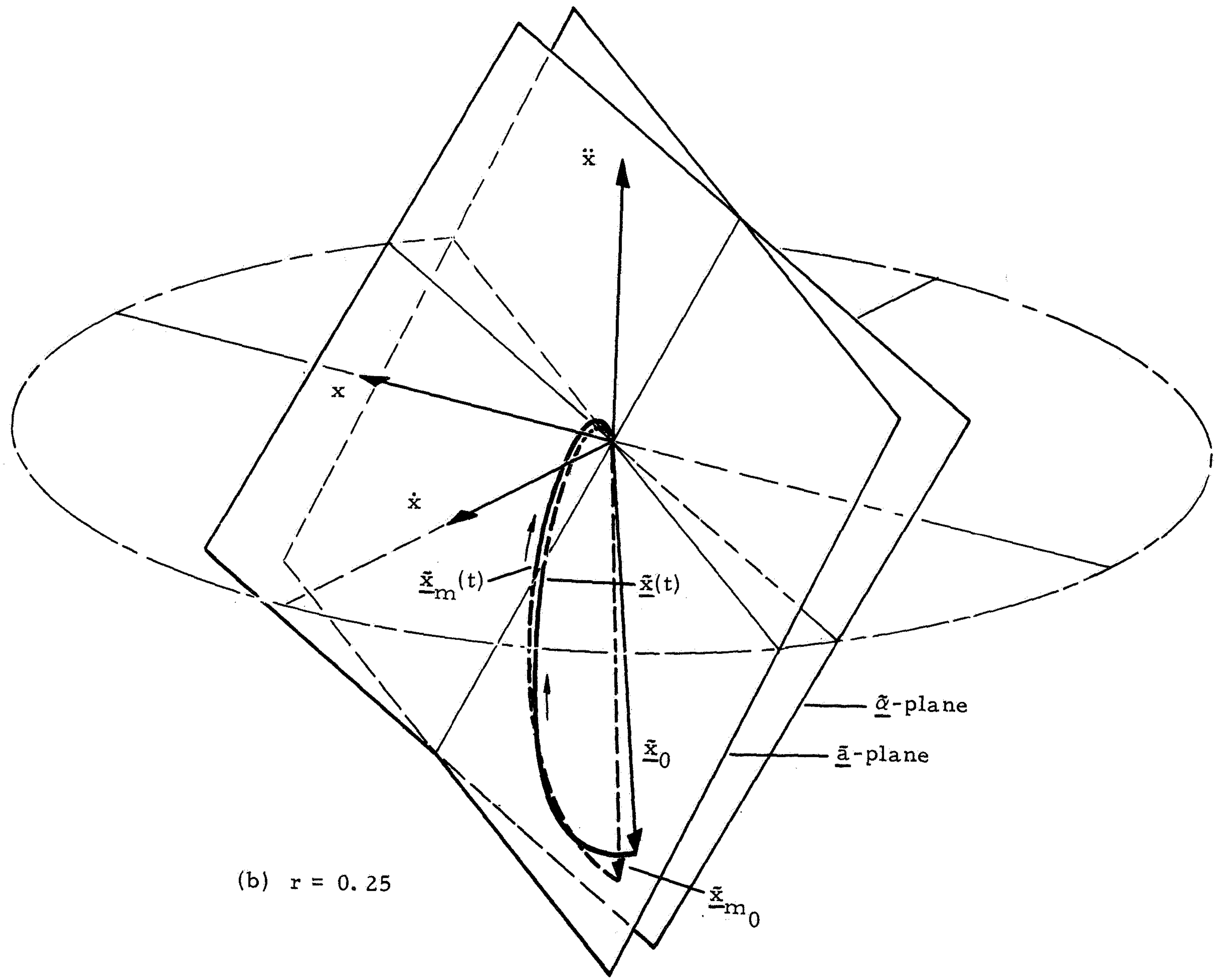
r	k_1	k_2	$\underline{\tilde{a}}'$			$\underline{\tilde{x}}_0'$		
0.0025	1.59	0.48	1.08	1.40	1.0	−1.0	1.08	−0.45
0.25	1.82	0.32	1.38	1.62	1.0	−1.0	1.38	−0.86
25.0	1.83	0.29	1.42	1.64	1.0	−1.0	1.42	−0.91

The solutions for r = 0.0025 and r = 0.25 are compared to the model in figures 3-13 parts a and b respectively from the geometrical viewpoint.



(a) r = 0.0025

Figure 3-13 Geometrical Representation of Model PI Solution for Example 3-3



(b) r = 0.25

Figure 3-13 Concluded

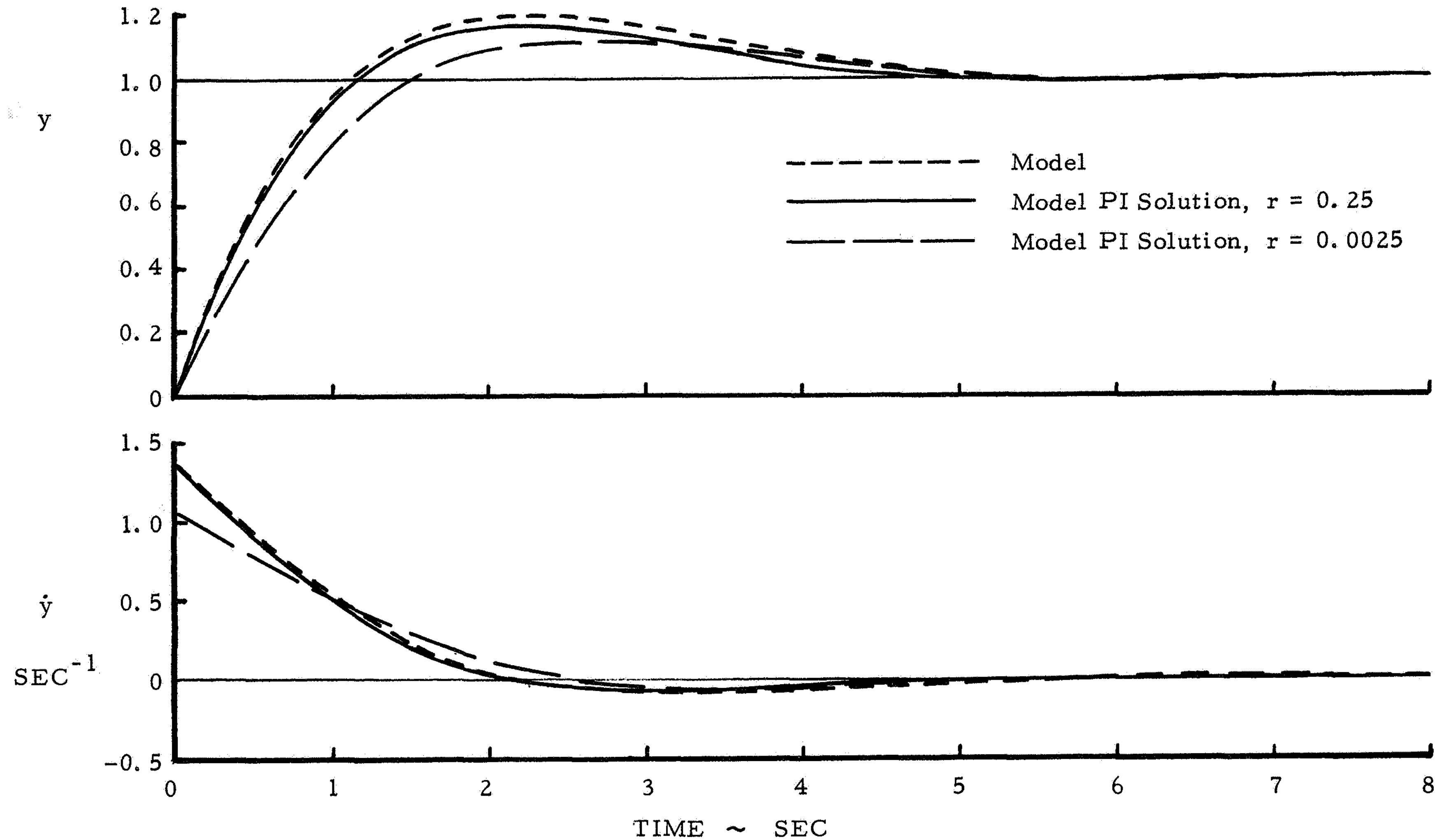


Figure 3-14 Comparison of Step Responses of Model PI Solutions for Example 3-3

The solution for r = 25 is essentially the same as for r = 0. 25. In figure 3-13 part a where the pseudo IC's had almost zero weighting in the Model PI, the solution gives an $\underline{\tilde{a}}$-plane that is very close to the model's characteristic plane, but the trajectories differ substantially because of the different pseudo IC's. On the other hand, in part b where the two terms in the Model PI had equal weight, the pseudo IC's are close, the characteristic planes are close and thus the trajectories are close. The trajectories are more easily compared as time histories of the corresponding step responses as in figure 3-14. The importance of matching the pseudo IC's is clearly shown.

Example 3-4

This next trivial example illustrates a geometrical interpretation of pole-zero cancellation. Consider the system and model in figure 3-15 where both k and τ are free design parameters.

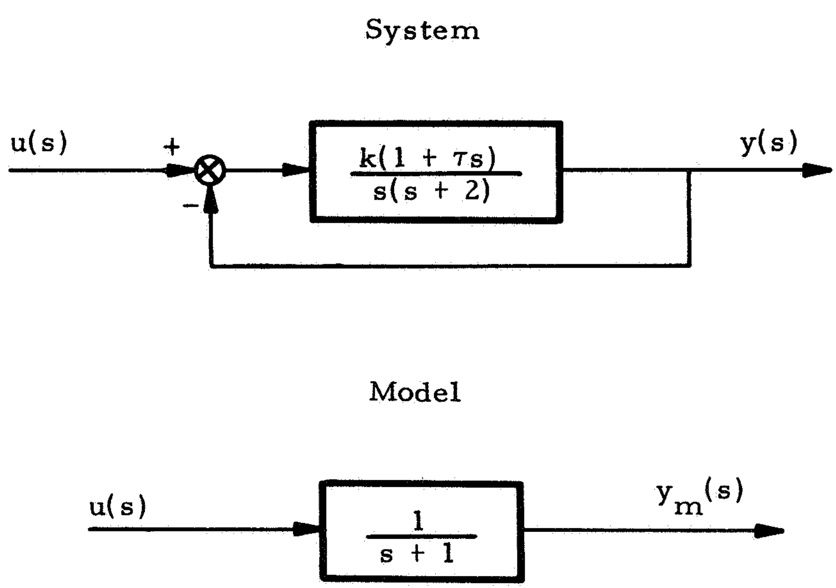


Figure 3-15 Block Diagram of System and Model for Example 3-4

The transient response of the closed-loop system is given by

$$\ddot{x} + (2 + k\tau)\,\dot{x} + kx = 0 \qquad (3\text{-}88)$$

with pseudo IC's, $x_0 = -1$, $\dot{x}_0 = k\tau$. Since the model is the same order as the number of excess system poles(see page 69), there is no quadratic penalty on pseudo IC's and the Model PI is given simply by

$$PI = \tfrac{1}{2}\int_0^{\infty} (x + \dot{x})^2\, dt \qquad (3\text{-}89)$$

The values of k and τ that minimize (3-89) are k = 2.0 and τ = 0.5. Referring to the system block diagram, one can see that for this value of τ the zero cancels one pole, reducing the system to effectively a first order. The k chosen then makes the system match the model exactly. The resulting closed-loop transfer function is

$$\frac{y(s)}{u(s)} = \frac{(s + 2)}{(s^2 + 3s + 2)} \qquad (3\text{-}90a)$$

or

$$\frac{y(s)}{u(s)} = \frac{(s + 2)}{(s + 1)(s + 2)} \qquad (3\text{-}90b)$$

or

$$\frac{y(s)}{u(s)} = \frac{1}{(s + 1)} \qquad (3\text{-}90c)$$

Equations (3-90a) and (3-90c) would appear to present a dilemma for representing the system in the geometrical sense of a characteristic plane and pseudo IC. The system's trajectory must lie within its characteristic plane yet (3-90a) and (3-90c) have different characteristic planes. Equation (3-90a) has a characteristic plane defined in three-dimensional space while equation (3-90c) has a "characteristic plane" defined in two-dimensional space, i.e. a line. The only way for this to be true is for the system's trajectory in three-dimensional space to lie along the intersection of the characteristic plane for (3-90a) and the extended characteristic plane for (3-90c). Then the projection of the trajectory into the two-dimensional space would lie along the line representing the "characteristic plane" for (3-90c). Figure 3-16 illustrates that such is the case. The system's characteristic plane, $\underline{\tilde{a}}$-plane, is actually the one defined for (3-90a), which is normal to

$$\underline{\tilde{a}}' = [2 \quad 3 \quad 1] \tag{3-91}$$

with pseudo IC vector

$$\underline{\tilde{x}}'_0 = [-1 \quad 1 \quad -1] \tag{3-92}$$

The model's characteristic plane, $\underline{\hat{\alpha}}$-plane, which also corresponds to (3-90c), is normal to

$$\underline{\hat{\alpha}}' = [1 \quad 1] \tag{3-93}$$

in two-dimensional space, with pseudo IC vector

$$\underline{\hat{x}}_{m_0} = [-1 \quad 1] \tag{3-94}$$

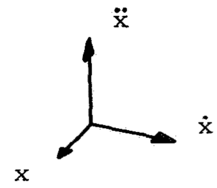
The extension of the $\underline{\hat{\alpha}}$-plane into the three-dimensional space, $\underline{\tilde{\alpha}}$-plane, is normal to

$$\underline{\tilde{\alpha}}' = [1 \quad 1 \quad 0] \tag{3-95}$$

The $\underline{\tilde{a}}$-plane and the $\underline{\tilde{\alpha}}$-plane intersect along the system's pseudo IC vector. If the system's pseudo IC vector for (3-90a) did not lie along the intersection, its projection into the two-dimensional space, the $x\dot{x}$-plane could not be the pseudo IC for (3-90c), which is the same as the model.

One can state that pole-zero cancellation occurs if and only if the system's psuedo IC vector lies along the intersection of the characteristic and extended characteristic planes defined with and without the cancelled pole. This must also be true for hyperplanes in higher dimensional spaces. Several pole-zero cancellations would correspond to the pseudo IC vector lying along the intersection of a like number of hyperplanes.

The coordinate frame
is denoted by



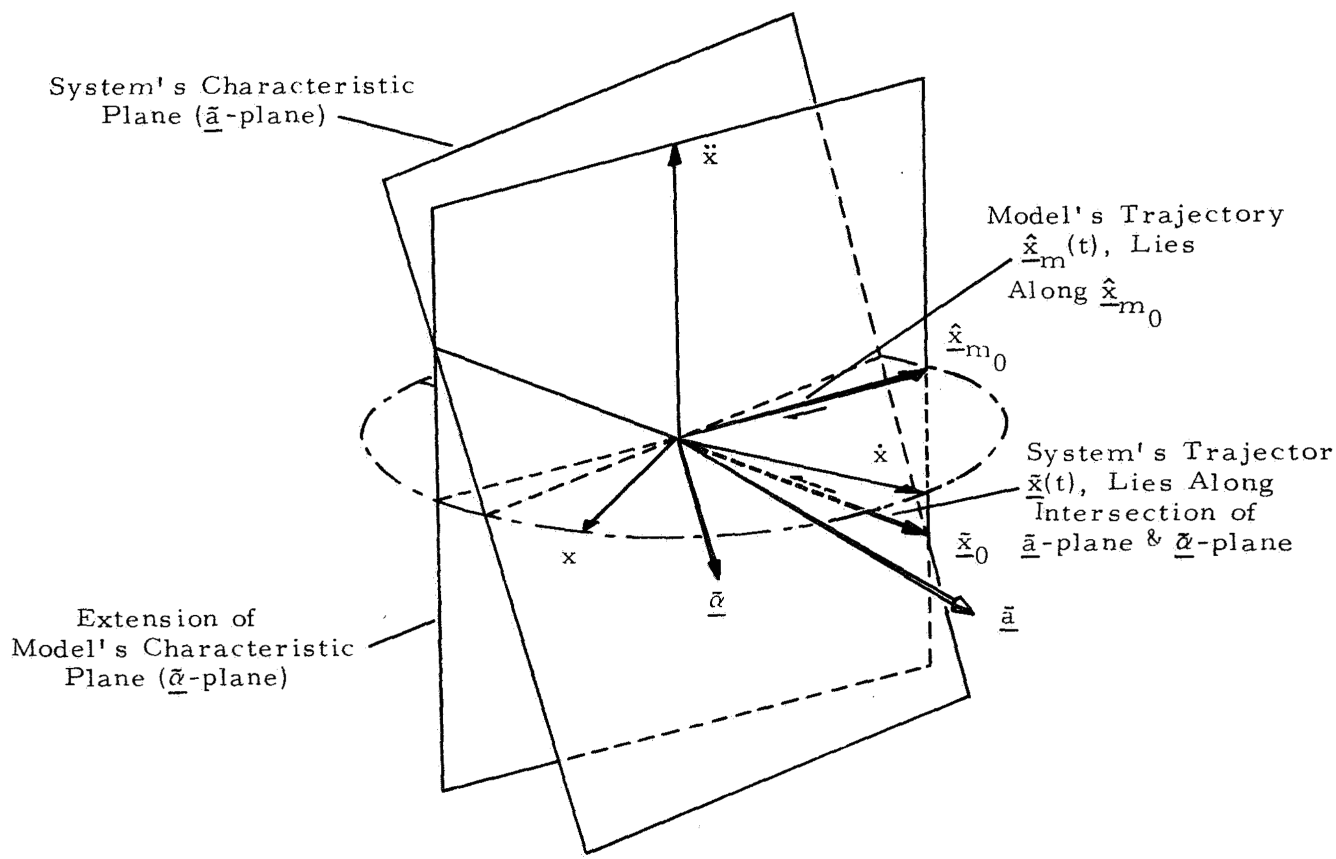


Figure 3-16 Geometrical Representation of Pole Zero Cancellation for Example 3

The importance of the geometrical interpretation of pole-zero cancellation is basically academic. It makes the concept of representing a system by a characteristic plane and pseudo IC in the extended state space more complete. From the practical standpoint it merely indicates that the Model PI approach can result in a pole-zero cancellation.

3.2.3 Multivariable Systems

The extension of the Model PI to systems with zeros, now makes it possible to consider multivariable control systems. If a multivariable system is completely controllable and completely observable (43), the transfer functions of the various input-output channels differ only by their zeros. Design of such systems necessarily involves the effects of zeros.

The intent of this chapter is to develop the theory of the Model PI, leaving the practical aspects to other chapters. However one comment of a practical nature should preface this section. The formulation of a design problem for a multivariable control system only requires those input-output transfer characteristics directly governed by the design specifications. It behooves the designer to reduce these to a minimum number before starting the synthesis effort. Complicated multivariable design problems are not easy by any technique yet devised. A control system with several input, several control devices, and several measurable output variables may be represented as a multivariable system, but if the design specifications are only on one input-output transfer characteristic it could be treated as a single input-output design problem. Multiple feedback to several control devices would be treated as inner loops. In other special cases it may be possible to select a model for one input-output relationship that would represent a satisfactory design for specifications on more than one input-output transfer characteristic. For example, if the zeros of two of the closed-loop transfer functions are unaffected by the choice of free design parameters, a model selected for one transfer function could be referenced to the other by the ratio of their zeros.

Other methods exist for reducing the number of input-output relationships that must be considered in the synthesis process. If the number can be reduced to one, then the methods of the previous section would be suggested.

The techniques developed in this section apply to the situation in which more than one input-output transfer characteristic must be considered in the design process. By reducing these to a minimum number, as discussed above, one should not have to consider more than two or three. Accordingly, the Model PI approach is developed here specifically for the case where design specifications are given for two input-output relationships of a multivariable system. The extension to more complex cases of theoretical interest should be obvious.

Two closed-loop input-output transfer functions of a multivariable control system can be written as

$$\frac{y_1(s)}{u_1(s)} = \frac{b_{1_m} s^m + \cdots + b_{1_1} s + b_{1_0}}{s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0} \quad \text{(3-96a)}$$

$$\frac{y_2(s)}{u_2(s)} = \frac{b_{2_m} s^m + \cdots + b_{2_1} s + b_{2_0}}{s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0} \quad \text{(3-96b)}$$

where some numerator coefficients may be zero. These may represent the response different output variables to different input variables, the same output variable ($y_1 = y_2$) to two different input variables, or different output variables to the same input variable ($u_1 = u_2$). Also (3-96) may be the composite representation of a closed-loop system including multiple inner loops.

Since both transfer functions (3-96) have the same characteristic equation, they also have the same characteristic plane. They are distinguished in the geometrical representation only by their respective pseudo IC vectors. It is possible then to describe the system for two input-output relationships by one characteristic plane,

$$\underline{\tilde{x}}'(t)\,\underline{\tilde{a}} = 0 \tag{3-97}$$

and two pseudo IC vectors, $\underline{\tilde{x}}_{1_0}$ for (3-96a) and $\underline{\tilde{x}}_{2_0}$ for (3-96b). For convenience, a system represented by its characteristic plane, normal to $\underline{\tilde{a}}$, and its pseudo IC vector $\underline{\tilde{x}}_0$, will be denoted by the short hand notation $\{\underline{\tilde{a}};\ \underline{\tilde{x}}_0\}$. Accordingly (3-96a) and (3-96b) are denoted respectively by

$$\begin{matrix} \{\underline{\tilde{a}};\ \underline{\tilde{x}}_{1_0}\} \\ \{\underline{\tilde{a}};\ \underline{\tilde{x}}_{2_0}\} \end{matrix} \tag{3-98}$$

The models of the desired closed-loop transient response for (3-96) can be similarly described by one $\underline{\tilde{\alpha}}$-plane and two pseudo IC vectors, and denoted by

$$\begin{matrix} \{\underline{\tilde{\alpha}};\ \underline{\tilde{x}}_{1m_0}\} \\ \{\underline{\tilde{\alpha}};\ \underline{\tilde{x}}_{1m_0}\} \end{matrix} \tag{3-99}$$

The models (3-99) are assumed to represent the design specifications for the corresponding systems (3-98). One would generally not expect the same set of values for the free design parameter that gives the best match of $\{\underline{\tilde{a}};\ \underline{\tilde{x}}_{1_0}\}$ to $\{\underline{\tilde{\alpha}};\ \underline{\tilde{x}}_{1m_0}\}$ to also give the best match of $\{\underline{\tilde{a}};\ \underline{\tilde{x}}_{2_0}\}$ to $\{\underline{\tilde{\alpha}};\ \tilde{x}_{2m_0}\}$. Therefore it is assumed that some relative priority for matching the two models has been established. The priority as well as the models are selected by the designer based on the actual design specifications as discussed in chapter 2.

Three synthesis methods for this class of multivariable control systems using the Model PI concept are presented. The first is probably the most straight forward analytical approach. It defines a performance index that is a weighted sum of the Model PI's corresponding to two single input-output design problems, with the weighting based on the relative matching priority. The second is less direct but simpler to implement. It defines a new model and hypothetical system that are the weighted averages of the two models (3-99) and of the two systems (3-98) respectively, then uses the single input-output procedure. The third approach is to apply the single input-output method alternately to the first system and model then to the second system and model, with selected parameters fixed in alternate applications. This does not require any extension in theory over that of section 3.2.2, and can be an effective practical method.

3.2.3.1 First Method

Assume that the priority of matching the two models is specified by a number, c, where $0 \leq c \leq 1$, such that c is the priority of matching $\{\underline{\bar{a}};\ \underline{\bar{x}}_{1_0}\}$ to $\{\underline{\bar{\alpha}};\ \underline{\bar{x}}_{1m_0}\}$ and (1-c) is the priority of matching $\{\underline{\bar{a}};\ \underline{\bar{x}}_{2_0}\}$ to $\{\underline{\bar{\alpha}};\ \underline{\bar{x}}_{2m_0}\}$. If these were two separate design problems, the corresponding Model PI's would be

$$PI_1 = r_1 \| \underline{\bar{x}}_{1_0} - \underline{\bar{x}}_{1m_0} \|^2_{\tilde{W}} + \int_0^\infty \| \underline{\bar{x}}_1(t) \|^2_{\tilde{Q}}\, dt \qquad (3\text{-}100)$$

$$PI_2 = r_2 \| \underline{\bar{x}}_{2_0} - \underline{\bar{x}}_{2m_0} \|^2_{\tilde{W}} + \int_0^\infty \| \underline{\bar{x}}_2(t) \|^2_{\tilde{Q}}\, dt \qquad (3\text{-}101)$$

where $\tilde{Q}$, given by (3-36) is the same for both. However the design problem is to treat them jointly with the appropriate priorities. A direct approach is to combine PI_1 and PI_2 into one performance index in such a manner as to reflect their relative priorities. The minimum values of PI_1 and PI_2 may be of grossly different magnitudes due to the scale factors. Thus a suitable combined performance index is

$$PI = \frac{c}{\| \underline{\bar{x}}_{1m_0} \|^2_{\tilde{W}}} PI_1 + \frac{(1-c)}{\| \underline{\bar{x}}_{2m_0} \|^2_{\tilde{W}}} PI_2 \qquad (3\text{-}102)$$

Minimizing (3-102) with respect to the free parameters would then produce a design that compromises meeting the individual performance indices in the ratio of their priorities.

3.2.3.2 Second Method

The geometrical interpretation of selecting the system's characteristic plane to make its trajectory lie close to the model's characteristic plane, motivates this second method. From that viewpoint, the multivariable design process considered here can be thought of as a compromise between making the trajectories of $\{\underline{\tilde{a}};\ \underline{\tilde{x}}_{1_0}\}$ and of $\{\underline{\tilde{a}};\ \underline{\tilde{x}}_{2_0}\}$ lie close to the model's characteristic plane, with the relative weighting being set by the priority number, c. In order to weight the two systems correctly, it is necessary to scale the trajectories by normalizing the pseudo IC's. Since the Model PI (3-82) penalizes deviations of the system's pseudo IC from the model's, it is necessary to also normalize the model's pseudo IC. They must be normalized in the model's state space because they are compared in that space.

Define a single model that is the weighted average of the two models (3-99) as

$$\{\underline{\tilde{\alpha}};\ \underline{\tilde{x}}_{m_0}\} \tag{3-103}$$

with

$$\underline{\tilde{x}}_{m_0} = c\,\frac{\underline{\tilde{x}}_{1m_0}}{\|\tilde{W}\underline{\tilde{x}}_{1m_0}\|} + (1-c)\,\frac{\underline{\tilde{x}}_{2m_0}}{\|\tilde{W}\underline{\tilde{x}}_{2m_0}\|} \tag{3-104}$$

This corresponds to a trajectory lying in the $\underline{\tilde{\alpha}}$-plane somewhere between the trajectories of the two original models (3-99) each normalized by the length of its pseudo IC state vector. The model (3-103) approaches the first model as c goes to unity and approaches the second model as c goes to zero.

Likewise define a single, hypothetical system that is the weighted average of the two systems (3-98) as

$$\{\underline{\tilde{a}};\ \underline{\tilde{x}}_0\} \tag{3-105}$$

with

$$x_0 = c \frac{\underline{\tilde{x}}_{1_0}}{\| \tilde{W}\underline{\tilde{x}}_{1_0} \|} + (1-c) \frac{\underline{\tilde{x}}_{2_0}}{\| \tilde{W}\underline{\tilde{x}}_{2_0} \|} \tag{3-106}$$

The matrix $\tilde{W}$ projects any vector in the (n+1)-dimensional space into the ℓ-dimensional state space of the model. The vectors $\underline{\tilde{x}}_{1_0}$ and $\underline{\tilde{x}}_{2_0}$ are normalized in (3-106) such that their projections into the model's state space are unit vectors. This assures that $\underline{\tilde{x}}_0$ and $\underline{\tilde{x}}_{m_0}$ can be compared on the same scale in the model' state space. The trajectory of the system (3-105) lies in the $\underline{\tilde{a}}$-plane somewhere between the two original systems (3-98) for any choice of free parameter values. Its relative location is dependent on the priorities in the same manner as for the model.

This second method proposes to use the weighted averaged model and system in the Model PI design procedure in place of the two models and systems. The problem is in effect converted to a single input-output design using the Model PI (3-82), repeated here,

$$PI = r \| \underline{\tilde{x}}_0 - \underline{\tilde{x}}_{m_0} \|^2_{\tilde{W}} + \int_0^{\infty} \| \underline{\tilde{x}}(t) \|^2_{\tilde{Q}}\, dt \tag{3-82}$$

with the model taken as (3-103), the system as (3-105) and the respective pseudo IC's as (3-104) and (3-106). Minimizing (3-82) in this case, tends to match the weighted averaged system to the weighted averaged model.

The first method minimizes a weighted average of the performance of the two systems; whereas the second method minimizes the performance of a weighted average of the two systems. In each case the performance is measured by the Model PI criterion. The two are clearly not equivalent. The first represent more closely the true intent of priorities given for satisfying the two separate specifications. However, the second requires about half the computational effort in the minimization process as the first and in application may be just as effective.

3.2.3.3 Third Method

A less sophisticated but practical technique is to apply the single input-output method alternately to the two coupled systems (3-98), with selected parameters fixed, until a satisfactory design is established for both. That is, alternately minimize PI_1 (3-100) then PI_2 (3-101) with respect to the most effective free parameters in each case, until one set of parameters is determined to give satisfactory matching of the respective models (3-99). The designer usually has some ideas as to which parameters are the most effective for minimizing each performance index. The least effective ones are held fixed. In alternate applications the number of free parameters is reduced by retaining only the most effective and fixing the others at the best value from the previous minimization. If the minimum of both PI_1 and PI_2 are equally sensitive to the same parameter, the priority number discussed could be used as a guide for selecting a compromise between the best values of the parameter for each minimization.

This iterative procedure sounds long and involved, but often it converges rapidly and is an efficient method. First of all in practice the number of free parameters should be kept to a minimum, three or four at most. If the two systems are only lightly coupled, the most effective parameters for one are often the least effective for the other. In such a case two or possibly three iterations may be sufficient. Even in highly coupled systems relatively few iterations may be adequate if parameters are averaged in the ratio of their priorities after the initial minimizations.

3.3 Relationship to Previous Works

It is interesting and informative to compare the Model PI to a performance index originated by Aizerman (18) and later generalized by Rekasius (19). Many other papers have considered performance indices that include a weighted combination of the system output and its derivatives, but only these two and the current work share the concept of relating the weighting factors to the desired response model. Aizerman's and Rekasius' works are reviewed briefly in this section to indicate the conceptual similarities and point out some restrictions and difficulties in their application.

3.3.1 Review of Aizerman's Work (Reference 18)

Aizerman originated the concept of specifying the desired response of an autonomous system in a performance index by using a linear combination of the squares of the output and its derivatives. The system considered is of the general form

$$x^{(n)} + a_{n-1}x^{(n-1)} + \cdots + a_2\ddot{x} + a_1\dot{x} + a_0 x = 0 \qquad (3\text{-}107)$$

with constant initial condition. The coefficients a_i, for $i = 0, 1, \cdots, n-1$, are functions of the free design parameters. The general form of the performance index proposed is

$$I = \int_0^\infty \{x^2 + \tau_1^2\dot{x}^2 + \tau_2^4\ddot{x}^2 + \cdots + \tau_\eta^{2\eta}[x^{(\eta)}]^2\}\, dt \qquad (3\text{-}108)$$

where the weighting factors τ_j are determined by the characteristic equation of the desired response.

Aizerman only considered in detail the case for $\eta = 1$ which reduces (3-108) to

$$I = \int_0^\infty (x^2 + \tau^2\dot{x}^2)\, dt \qquad (3\text{-}109)$$

By completing the square of the integrand it can be put in the form

$$I = \int_0^\infty (x + \tau\dot{x})^2 \, dt - 2\tau \int_0^\infty x\dot{x} \, dt \tag{3-110}$$

The second integral can be evaluated directly to give

$$I = \int_0^\infty (x + \tau\dot{x})^2 \, dt + \tau x(0)^2 \tag{3-111}$$

assuming the system is asymptotically stable, i. e. $x(\infty) \to 0$.

Since the last term of (3-111) is independent of the free design parameters, the absolute minimum value of (3-111) occurs if the free parameters are chosen such that the integrand is zero, i. e.

$$x + \tau\dot{x} = 0 \tag{3-112}$$

Aizerman states " the differential equation (3-112) defines the transient response which can be approached in the limit, if it is possible to select the parameters in such a way that $I = I_{min\ min}$. (Aizerman denotes the absolute minimum value of I as $I_{min\ min}$; in this case $I_{min\ min} = \tau x(0)^2$.) The optimum response is described by the exponential $x(t) = x(0)\, e^{-t/\tau}$.

The value of τ shall be selected in such a way that the exponential will satisfy the specifications of the transient response. "

One might erroneously conclude from this simple example considered by Aizerman that the Model PI is equivalent to Aizerman's performance index. For this simple case they would give the same result. If higher order models of the desired response are necessary then one would retain more terms in (3-108). However the weighting factors τ_j do not have a simple relationship to the coefficients of the model's characteristic equation except for the first order model case. It becomes increasingly more difficult to establish this relationship as more terms are retained in (3-108). The Model PI (3-35), on the other hand, is written directly in terms of the model's characteristic equation coefficients.

There are three serious limitations to the application of Aizerman's performance index. First of all it restricts the possible choice of poles of the model to specific regions of the s-plane. This point is easily shown for a second order model, i. e.

$$I = \int_0^{\infty} [x^2 + \tau_1^2 \dot{x}^2 + \tau_2^4 \ddot{x}^2]\, dt \qquad (3\text{-}113)$$

Completing the square as before and integrating where possible gives

$$I = \int_0^{\infty} [x + \sqrt{\tau_1^2 + 2\tau_2^2}\ \dot{x} + \tau_2^2 \ddot{x}]^2\, dt + f(x_0, \dot{x}_0) \qquad (3\text{-}114)$$

The second term is of no interest here because x_0 and $\dot{x}_0$ are independent of free parameters. The $I_{min\ min}$ occurs if

$$\ddot{x} + \frac{1}{\tau_2^2}\sqrt{\tau_1^2 + 2\tau_2^2}\ \dot{x} + \frac{1}{\tau_2^2} x = 0 \qquad (3\text{-}115)$$

Using this approach, the designer select τ_1 and τ_2 so that (3-115) represents the desired response. The natural frequency and damping ration for (3-115) are

$$\omega = \frac{1}{\tau_1}$$

$$\zeta = \tfrac{1}{2}\sqrt{\tau_1^2/\tau_2^2 + 2} \qquad (3\text{-}116)$$

The smallest value that ζ can have is $\zeta = 1/\sqrt{2}$, which occurs when $\tau_1 = 0$. Therefore, the designer is restricted to selecting a model within the region shown in figure 3-17.

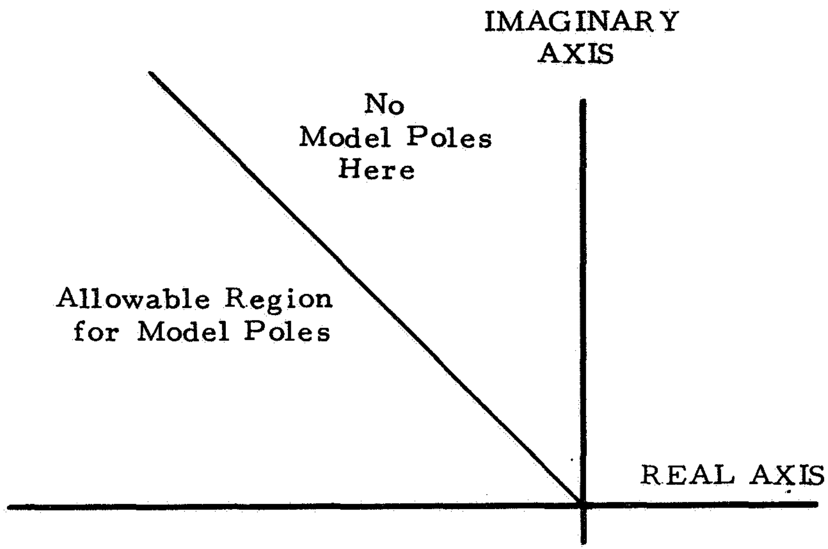


Figure 3-17 Restrictions on Choice of Second Order Models in Aizerman's Approach

Similar but more complex restrictions apply to higher order models. The Model PI clearly has no restrictions on the choice of model poles.

The second limitation is that this approach does not provide any way to consider models with zeros. Relating the integrand of the performance index in the completed square form, such as (3-114), to the desired response represents models with poles only. There is no way to include the effect of model zeros in Aizerman's performance index.

The third limitation is that it is not valid for systems with zeros in general. Aizerman represented the step response of a system by an autonomous system and suitable initial conditions (pseudo IC's), but did not consider the case for systems with zeros in which the pseudo IC's are generally functions of the free parameters. Under such a situation the argument, that the absolute minimum of the performance index occurs when the integrand in the complete square form is zero, becomes invalid. For example, in (3-114), $f(x_0, \dot{x}_0)$ would be a function of the free parameters, and it may be possible to produce a smaller value of I by decreasing $f(x_0, \dot{x}_0)$ and accepting a small nonzero value of the integral. The Model PI, as extended in section 3.2.2, specifically treats models and systems with zeros.

3.3.2 Review of Rekasius' Work (Reference 19)

Rekasius observed that Aizerman's performance index restricted the location of model poles and excluded models with zeros, and proposed a somewhat different one. It lifts the restrictions on model poles, is a more convenient form to use than Aizerman's, but fails to treat models and systems with zeros properly. Rekasius' general performance index is

$$I = \int_0^\infty \left(x^2 + \sum_{i=1}^{\eta} \tau_i^2 \, [x^{(i)}]^2 + 2x \sum_{i=1}^{\eta} \tau_i x^{(i)} + 2 \sum_{i=1}^{\eta} \sum_{j=i+2}^{\eta} \tau_i \tau_j x^{(i)} x^{(j)} \right) dt \qquad (3\text{-}117)$$

By completing the square of the integrand and integrating where possible, it reduces to

$$I = \int_0^\infty \left[x + \sum_{i=1}^{\eta} \tau_i x^{(i)} \right]^2 dt + \tau_1 x_0^2 + \sum_{i=1}^{\eta-1} \tau_i \tau_{i+1} [x_0^{(i)}]^2 \qquad (3\text{-}118)$$

Rekasius then states "the absolute minimum $I_{min\,min}$ of this performance index occurs when the integrand of (3-118) is equal to zero. Consequently the characteristic equation of the ideal model for the proposed performance index (3-117) is

$$x + \tau_1 \dot{x} + \tau_2 \ddot{x} + \cdots + \tau_\eta x^{(\eta)} = 0 \qquad (3\text{-}119)$$

The ideal model itself can be represented by a closed-loop system with the transfer function

$$\frac{1}{\tau_\eta s^\eta + \tau_{\eta-1} s^{\eta-1} + \cdots + \tau_2 s^2 + \tau_1 s + 1} \text{"} \qquad (3\text{-}120)$$

Rekasius refers to the model as the " ideal model" in the sense that if a sufficient number of unconstrained parameters were chosen to minimize the performance index (3-117) the control system would be the same as this ideal model. The first goal of removing the restriction on the location of the model's poles is met by (3-117) since there are no restriction on the choice of τ_i in (3-120). Also the performance index can be written down directly from the model transfer function, which avoids the extra computations required in general to form Aizerman's performance index. In these two respects, the Model PI and Rekasius' performance index are the same. <u>For systems without zeros</u> the last term of (3-118) is zero and the next to last term is constant, so that the two performance indices (3-35) and (3-118) are equivalent in that case.

However Rekasius' performance index does not provide for models with zeros any more than does Aizerman's, although that was one of the objectives of Rekasius' work. Furthermore, if the system considered has zeros, Rekasius' performance index is not even valid except for some special cases.

Rekasius observed correctly that the performance index must depend on the model's pseudo IC's in order to differentiate between models that have the same poles but different zeros. This fact motivated the selection of the form (3-117). The questions not answered in reference 19 are:

1) What relevance do the initial conditions included in (3-118) have to the ideal model?
2) What relationship do they have to the zeros of the system?
3) Why use this particular combination of initial conditions over any other combination that would arise from some quadratic functional?

The initial conditions in (3-118) must be interpreted as pseudo IC's, as defined in section 3. 1, to have any meaning for systems with zeros. As pseudo IC's of the system, they are in general functions of the free parameters. If one assumes they are also the pseudo IC's of the model, several problems arise. First of all, if the system and model are of different order it may be impossible for the pseudo IC's to be the same. Complying with the compatibility rule suggested in section 3. 2. 2 would reduce the likelihood of that occurring. Secondly, even if they can be the same, one can not require this equality without placing an undesirable constraint on the free parameters. In some cases such an equality constraint would completely determine the parameters and the integral portion of (3-118) would be superfluous. Thirdly, if the equality of pseudo IC's is not explicitly included in the performance index the model's zeros become superfluous. Therefore the initial conditions included in (3-118) are not actually relevant to the ideal model.

The pseudo IC's that must be used in (3-118) are related to the systems zeros by equation (3-8). Including them in the manner of (3-118) is not only irrelevant but actually destroys Aizerman's original concept that $I_{min\ min}$ be obtained when the integrand is zero. The last term of (3-118) is a function of the free parameters, so that it is entirely possible to obtain a smaller value of the performance index at a point where the integrand is not zero. This is the same difficulty encountered with Aizerman's performance index. Rekasius' performance index is similarly invalid for systems with zeros, except for some special cases. These exceptions occur when the pseudo IC's are not functions of the free parameters even though the system has zeros. By referring to equation (3-8) one can see that this is possible if none of the numerator coefficients, b_i, and none of the denominator coefficients appearing in (3-8) are functions of the free parameters. In that situation the last term of (3-118) is independent of the free parameters and thus can be discarded as far as the minimization process is concerned. In which case, Rekasius' performance index becomes equivalent to the Model PI.

The third question posed earlier about the particular combination of initial conditions included in (3-118) now becomes academic. There appears to be no justification for this particular selection over other possible ones. Other quadratic functionals exist that can be reduced to the same form as (3-118) with the addition of terms depending only on the pseudo IC's. From the argument presented here, any such quadratic functional would be equivalent in application to Rekasius' performance index.

3.4 Summary of Results

The main results of this chapter are summarized here for easy reference and application. Table 3-4 summarizes the relationship between the transfer function representation of a system and the geometrical representation by its characteristic plane and pseudo IC vector. Using this table one can quickly convert a system or model transfer function into the form required for using the Model PI. Table 3-5 is a guide to selecting the model structure for the Model PI as well as a summary of the Model PI for single input-output system application. The choices of model structure are listed in order of preference for best results with the Model PI. The reader is referred to section 3.2.3 for application to multivariable systems.

TABLE 3-4

SUMMARY OF RELATIONSHIP BETWEEN TRANSFER FUNCTIONS AND GEOMETRICAL REPRESENTATIONS OF CONTROL SYSTEM AND MODEL

	Geometrical Representation of System and Model	
Control System and Model Transfer Functions	Characteristic Planes	Pseudo Initial Conditions
Closed-Loop Control System $\frac{y(s)}{u(s)} = \frac{b_m s^m + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_2 s^2 + a_1 s + a_0}$ where n = Number of Closed-Loop System Poles m = Number of Closed-Loop System Zeros $m \leq n-1$ and the coefficients b_i $(i = 0, 1, \cdots m)$ and a_j $(j = 0, 1, \cdots n-1)$ are generally functions of the free design parameters	Characteristic Plane (n+1)-Space $\underline{\tilde{x}}'\underline{\tilde{a}} = 0$ Extended State Vector $\underline{\tilde{x}}' = [\underline{x}' \mid x^{(n)}] = [\underline{x}' \mid -\underline{x}'\underline{a}]$ Control System State Vector $\underline{x}' = [x \; \dot{x} \; \ddot{x} \cdots x^{(n-2)} x^{(n-1)}]$ Extended Coefficient Vector $\underline{\tilde{a}}' = [\underline{a}' \mid 1]$ Coefficient Vector $\underline{a}' = [a_0 \, a_1 \, a_2 \cdots a_{n-2} \, a_{n-1}]$	Extended Pseudo IC Vector (n+1)-Space $\underline{\tilde{x}}_0' = [\underline{x}_0' \mid -\underline{x}_0'\underline{a}]$ Control System's Pseudo IC Vector $\underline{x}_0' = [x_0 \, \dot{x}_0 \, \ddot{x}_0 \cdots x_0^{(n-2)} x_0^{(n-1)}]$ where $x_0 = b_0/a_0$ $x_0^{(n-i)} = \begin{cases} 0 & , i > m \\ b_i - \sum_{j=n-m}^{n-i-1} a_{j+i} x_0^{(j)}, & i = 1, \cdots, m \end{cases}$
Model $\frac{y_m(s)}{u(s)} = \frac{\beta_k s^k + \cdots + \beta_1 s + \beta_0}{s^\ell + \alpha_{\ell-1}s^{\ell-1} + \cdots + \alpha_2 s^2 + \alpha_1 s + \alpha_0}$ where ℓ = Number of Model Poles k = Number of Model Zeros and the coefficients are constants selected by the designer	Characteristic Plane $(\ell+1)$-Space $\underline{\hat{x}}_m'\underline{\hat{\alpha}} = 0$ $\underline{\hat{x}}_m' = [x_m \, \dot{x}_m \, \ddot{x}_m \cdots x_m^{(\ell-1)} x_m^{(\ell)}]$ $\underline{\hat{\alpha}}' = [\alpha_0 \, \alpha_1 \, \alpha_2 \cdots \alpha_{\ell-1} \quad 1]$ Extension Into (n+1)-Space Extended Characteristic Plane $\underline{\tilde{x}}_m'\underline{\tilde{\alpha}} = 0$ $\underline{\tilde{x}}_m' = [\underline{\hat{x}}_m' \mid x_m^{(\ell+1)} \cdots x_m^{(n)}]$ $\underline{\tilde{\alpha}}' = [\underline{\hat{\alpha}}' \mid 0 \cdots 0]$	Extended Pseudo IC Vector $(\ell+1)$-Space $\underline{\hat{x}}_{m_0}' = [\underline{x}_{m_0}' \mid -\underline{x}_{m_0}'\underline{\alpha}]$ Model's Pseudo IC Vector $\underline{x}_{m_0}' = [x_{m_0} \, \dot{x}_{m_0} \, \ddot{x}_{m_0} \cdots x_{m_0}^{(\ell-2)} x_{m_0}^{(\ell-1)}]$ where $x_{m_0} = -\beta_0/\alpha_0$ $x_{m_0}^{(\ell-1)} = \begin{cases} 0 & , i > k \\ \beta_i - \sum_{j=\ell-k}^{\ell-i-1} a_{j+i} x_{m_0}^{(j)}, & i = 1, \cdots, k \end{cases}$ Model's Pseudo IC Vector in (n+1)-Space $\underline{\tilde{x}}_{m_0}' = [\underline{\hat{x}}_{m_0}' \mid x_{m_0}^{(\ell+1)} x_{m_0}^{(\ell+2)} \cdots x_{m_0}^{(n)}]$ but $x_{m_0}^{(\ell+1)}, \cdots, x_{m_0}^{(n)}$ are unimportant

TABLE 3-5

A GUIDE FOR SELECTING THE MODEL STRUCTURE
WHEN USING THE MODEL PI

	Choice of Model Structural* (ℓ poles , k zeros)	Form of Model Performance Index to Use	Comments
For Control Systems with n Poles and NO Zeros (m = 0)	1. Model with no zeros and the same number of poles as system, i. e. $\ell = n$, $k = 0$. 2. Model with no zeros but fewer poles than system, i. e. $\ell < n$, $k = 0$.	$PI = \int_0^\infty \lVert \underline{\tilde{x}}(t) \rVert^2_{\tilde{Q}}\, dt$ with $\tilde{Q} = \dfrac{\underline{\tilde{\alpha}}\underline{\tilde{\alpha}}'}{\lVert \underline{\tilde{\alpha}} \rVert^2}$	At least the first ℓ pseudo IC's are the same for the system and model.
For Control Systems and m Zeros ($m \neq 0$)	1. Model with same structure as system, i. e. $\ell = n$, $k = m$. 2. Model with fewer poles than system but the same number of excess poles over zeros, i. e. $\ell < n$, but $\ell - k = n - m$	$PI = r \lVert \underline{\tilde{x}}_0 - \underline{\tilde{x}}_{m_0} \rVert^2_{\tilde{W}} + \int_0^\infty \lVert \underline{\tilde{x}}(t) \rVert^2_{\tilde{Q}}\, dt$ with $\tilde{Q}$ the same as above and $\tilde{W} = \left[\begin{array}{c\|c} I & O \\ \hline O & O \end{array}\right]$ I is the $\ell \times \ell$ identity matrix and the O's are appropriately dimensional null matrices.	The scalar r is chosen by the designer to give a good balance between matching the pseudo IC's and making the system's trajectory lie close to the $\underline{\tilde{\alpha}}$-plane.
	3. Model with no zeros and the number of poles is equal to or less than the number of excess system poles, i. e. $\ell \leq n-m$, $k = 0$.	$PI = \int_0^\infty \lVert \underline{\tilde{x}}(t) \rVert^2_{\tilde{Q}}\, dt$ with $\tilde{Q}$ the same as above.	At least the first ℓ pseudo IC's are the same for the system and model.

* Listed in descending preferential order

CHAPTER 4

NUMERICAL OPTIMIZATION METHOD

After defining a performance index the next step in parameter optimization design techniques is to minimize it with respect to the free design parameters. It is instructive to solve simple examples analytically, as is done in Appendix A, but few practical control-system design problems afford an analytic solution and must be solved by numerical techniques. In this chapter the expressions needed for numerical minimization of the Model PI are derived and one optimization algorithm is presented.

The Model PI is a nonlinear, generally non-quadratic, function of the free design parameters. It is a non-negative function and may have multiple stationary points over the parameter space. Several methods for finding local minima for such a function are reviewed in references 20 and 21. Since the main emphasis of this thesis is on the Model PI and its application rather than numerical optimization techniques, no effort is made here to seek out the most efficient optimization method.

It is possible to evaluate the Model PI (3-35) using Parseval's theorem and either tabulated integrals, which are available in several textbooks, e.g. Newton, Gould and Kaiser (15) or Gauchy's residue theorem. A simpler method for general quadratic functionals, suggested by Kalman and Bertran (42), is presented in section 4.1. The next section establishes the first order necessary conditions for a local minimum by using matrix variational calculus in a manner somewhat similar to Johansen (44) or Denham and Speyer (45). It is only practical to solve these analytically for simple academic examples. However this leads to a direct method for evaluating the gradient (section 4.3) that can be used in a numerical optimization technique.

The algorithm used to minimize the Model PI is presented in section 4.4. It is a fairly simple technique based on an averaged gradient direction. No claims are made for its relative efficiency compared to other available techniques, but it proved quite adequate for all the design problems treated here. In applying the Model PI, the reader may wish to use some other numerical optimization method that he is more familiar with or feels is more efficient. Whatever technique chosen, if it requires evaluation of the performance index and its gradient, the developments in sections 4.1 and 4.3 are still quite useful.

The state-space formulation of the design problem and the expression derived here make it possible to establish general digital computer programs for designing linear control systems. It is not necessary to write a new program for every new design problem. Such a program is described in Chapter 5 and Appendix B that only requires a minor subroutine and different input data cards to change from one design problem to another. Similar programs can be written with more efficient numerical optimization techniques if desired.

In each of the following sections, the derivations are presented first for a general quadratic functional, then extended to the Model PI. This is done for two reasons. It is easier to follow the details of the derivations for the general case than for the Model PI. And, the general results should be useful in themselves, since the general quadratic functional is frequently used in modern control theory.

4.1 Evaluation of a Quadratic Functional

Consider the linear autonomous system

$$\dot{\underline{x}} = F\underline{x} \tag{4-1}$$

with initial conditions $\underline{x}(0) = \underline{x}_0$. If the system is asymptotically stable, then the quadratic functional

$$J = \int_0^{\infty} \| \underline{x}(t) \|_Q^2 \, dt \tag{4-2}$$

where Q is a symmetric, positive semi-definite matrix, is equivalent to

$$J = \| \underline{x}_0 \|_P^2 \tag{4-3}$$

where P satisfied the matrix algebraic equation

$$F'P + PF = -Q \tag{4-4}$$

This is almost the second half of a corollary to Lyapunov's well known theorem on stability of linear invariant systems (42). The only difference is that here Q is allowed to be positive semi-definite. The result is easily shown to be true, but the converse, with Q semi-definite, is not necessarily true.

Using the solution to (4-1),*

$$\underline{x} = e^{Ft} \underline{x}_0 \tag{4-5}$$

in (4-2) gives

$$J = \int_0^{\infty} \underline{x}_0' e^{F't} Q e^{Ft} \underline{x}_0 \, dt \tag{4-6}$$

If P is defined as

$$P = \int_0^{\infty} e^{F't} Q e^{Ft} \, dt \tag{4-7}$$

then (4-6) becomes (4-3). The expression (4-7) for P is a known form (46) for a solution to (4-4).

---

* Time arguments are suppressed in equations where the meaning is clear.

Any quadratic functional of the form (4-2) can be evaluated by solving (4-4) for P which is then substituted into (4-3). If one wished to go through the algebraic exercise, it would be possible to reestablish the tabulated integrals in Newton, Gould and Kaiser (table E. 2-1 in reference 15) using this approach. Although it is possible to evaluate the quadratic functional (4-2) analytically, the real usefulness of this approach is that one digital computer program can be written to evaluate it numerically for any order system.

Solving the algebraic matrix equation (4-4) may not be the most efficient way on a digital computer. A possibly faster and more accurate method is to compute P as the steady-state solution of the linear matrix differential equation

$$\dot{P} = F'P + PF + Q \qquad (4\text{-}8)$$

for an arbitrary initial condition P(0). Equation (4-8) has a unique, positive semi-definite solution if F is a stable matrix and Q is positive semi-definite (46).

The Model PI (3-82) can be evaluated similarly, with simple modifications. It is defined in terms of the extended state vector which can be related to the state vector by

$$\underline{\bar{x}} = M\underline{x} \qquad (4\text{-}9)$$

where M is an (n + 1) × n partitioned matrix,

$$M = \begin{bmatrix} I \\ \hline -a' \end{bmatrix} \qquad (4\text{-}10)$$

The Model PI (3-82) can be written as

$$PI = r \| \underline{x}_0 - \underline{x}_{m_0} \|^2_W + \int_0^\infty \| \underline{x}(t) \|^2_{M'\tilde{Q}M} \, dt \qquad (4\text{-}11)$$

where Q is given by (3-36) and W is related to $\tilde{W}$ (3-81) by

$$\tilde{W} = \left[\begin{array}{c|c} W & \underline{0} \\ \hline \underline{0}' & 0 \end{array}\right] \tag{4-12}$$

Relating (4-11) to (4-2), it is clear that the Model PI can be evaluated by

$$PI = r \| \underline{x}_0 - \underline{x}_{m_0} \|^2_W + \| \underline{x}_0 \|^2_P \tag{4-13}$$

where P satisfies

$$F'P + PF = -M'\tilde{Q}M \tag{4-14}$$

4.2 Necessary Condition for a Local Minimum

The first order necessary condition for a local minimum of a general quadratic functional of the form (4-2) is derived then extended to the special case of the Model PI. The system (4-1), is assumed to be asymptotically stable and in the phase-variable canonical form, i. e.

$$F = \left[\begin{array}{c|c} \underline{0} & I \\ \hline \multicolumn{2}{c}{-\underline{a}'} \end{array}\right] \tag{4-15}$$

and the coefficient vector, $\underline{a}$, is some function of the free parameter vector $\underline{p}$, i. e.

$$\underline{a} = \underline{a}(\underline{p}) \tag{4-16}$$

For the present, the initial condition vector, $\underline{x}_0$, is assumed to be independent of $\underline{p}$. Later, when extended to the Model PI, it will correspond to the pseudo IC vector (3-18) which may depend on $\underline{p}$.

The problem is to determine the first order necessary condition that $\underline{p}$ must satisfy for (4-2) to have a local minimum value. The quadratic functional (4-2) can be written as

$$J = \text{tr}\left[Q\int_0^{\infty} \underline{x}\underline{x}'\, dt\right] \tag{4-17}$$

where tr [ ] indicates the trace of the matrix [ ]. Define a matrix X as

$$X = \int_0^{\infty} \underline{x}\underline{x}'\, dt \tag{4-18}$$

which is clearly symmetric and positive semi-definite. Substituting (4-5) into (4-18) gives

$$X = \int_0^{\infty} e^{Ft}\underline{x}_0\underline{x}_0' e^{F't}\, dt \tag{4-19}$$

which is easily shown (46) to be a solution of the matrix algebraic equation

$$FX + XF' = -X_0 \tag{4-20}$$

where $X_0 = \underline{x}_0\underline{x}_0'$.

Using (4-18) in (4-17) the problem can be restated as follows: determine the necessary condition for $\underline{p}$ to locally minimize

$$J = \text{tr}\,[QX] \tag{4-21}$$

subject to the constraint (4-20). This can be solved using Lagrange's method for constraints by adjoining (4-20) to (4-21) with a Lagrange multiplier matrix P. It is sufficient to add

$$\text{tr}\,\{P\,[FX + XF' + X_0]\} \tag{4-22}$$

to (4-21) in order to constrain every element of X to satisfy (4-20), so that P can be assumed to be symmetric without loss of generality. The augmented J becomes

$$J = \mathrm{tr}\,\{QX + P[FX + XF' + X_0]\} \tag{4-23}$$

Both X and F depend on $\underline{p}$ so that a first order variation in J due to a variation $\delta\underline{p}$ is

$$\delta J = \mathrm{tr}\,\{Q\delta X + P[\delta F X + F\delta X + \delta X F' + X\delta F']\} \tag{4-24}$$

The variation δF can be determined directly in terms of $\delta\underline{p}$ from (4-15) and (4-16) as

$$\delta F = -\underline{\epsilon}_n\, \delta\underline{p}' A' \tag{4-25}$$

where $\underline{\epsilon}_n$ is an $n \times 1$ vector defined as

$$\underline{\epsilon}_n' = [0 \quad 0 \quad 0 \cdots 0 \quad 1] \tag{4-26}$$

and A is the sensitivity matrix of $\underline{a}$ with respect to $\underline{p}$, i. e.

$$A = \left[\frac{\partial \underline{a}(\underline{p})}{\partial \underline{p}}\right] \tag{4-27}$$

Using (4-25) in (4-24), the variation of δJ becomes

$$\delta J = \mathrm{tr}\,\{-2\underline{\epsilon}_n' PXA\delta\underline{p} + [PF + F'P + Q]\delta X\} \tag{4-28}$$

If J is at a minimum value, then to first order, the variation δJ due to $\delta\underline{p}$ must be zero, for arbitrary $\delta\underline{p}$ and δX. Therefore $\underline{p}$ must satisfy

$$FX + XF' + X_0 = 0 \tag{4-29a}$$

$$F'P + PF + Q = 0 \tag{4-29b}$$

$$\underline{\epsilon}_n' PXA = 0 \tag{4-29c}$$

at any minimum point. Equation (4-29) is the first order necessary condition for a local minimum of the quadratic functional (4-2).

It is interesting to note that the Lagrange multiplier matrix, P, is exactly the matrix satisfying (4-4) needed to evaluate the quadratic functional. Once the necessary condition (4-29) is solved, the minimum value of the quadratic functional is easily computed from (4-3).

Extending this result to the Model PI (4-11) involves two additional considerations: 1) the state vector weighting matrix contains M which is dependent on $\underline{p}$; and 2) the pseudo IC vector is dependent on $\underline{p}$, so that X_0 and the term $r \| \underline{x}_0 - \underline{x}_{m_0} \|_W^2$ are also dependent on $\underline{p}$. The derivation closely parallels the previous one. Where possible only the additional terms are considered below.

The augmented performance index, corresponding to (4-23), is

$$PI = \text{tr}\ \{r\, W\left[X_0 - 2X_{m_0} + X_{mm_0}\right] + M'\tilde{Q}MX + P\left[FX + X'F + X_0\right]\} \qquad (4\text{-}30)$$

where

$$X_{m_0} = \underline{x}_0 \underline{x}'_{m_0}$$
$$X_{mm_0} = \underline{x}_{m_0} \underline{x}'_{m_0} \qquad (4\text{-}31)$$

Taking the first variation of (4-30) adds two variational terms,

$$\text{tr}\ \{\delta\left[M'\tilde{Q}M\right] X\} \qquad (4\text{-}32)$$

and

$$\text{tr}\ \{\left[r\,W + P\right]\delta X_0 - 2rW\delta X_{m_0}\} \qquad (4\text{-}33)$$

over the corresponding form for δ(PI) (4-28). It is worthwhile reducing these to functions of $\delta\underline{p}$ before writing down the complete δ(PI).

By referring to the definitions of $\tilde{Q}$, $\underline{\tilde{\alpha}}$, and M from (3-36), (3-44), and (4-10) respectively, it is possible to write the term $M'\tilde{Q}M$ as

$$M'\tilde{Q}M = \left[\begin{array}{c:c} I & -\underline{a} \end{array}\right] \left[\begin{array}{c} \underline{\alpha} \\ \hdashline 1 \\ \hdashline \underline{0} \end{array}\right] \frac{\left[\begin{array}{c:c:c} \underline{\alpha}' & 1 & \underline{0}' \end{array}\right]}{\|\underline{\tilde{\alpha}}\|^2} \left[\begin{array}{c} I \\ \hdashline -\underline{a}' \end{array}\right] \tag{4-34}$$

where

- I is the $n \times n$ identity matrix
- $\underline{\alpha}$ is the $\ell \times 1$ model coefficient vector
- $\underline{0}$ is the $(n - \ell) \times$ null vector

Observe in (4-34) that $\underline{a}$ is always multiplied by zero unless $n = \ell$. Since $\underline{a}$ is the only quantity in (4-34) depending on $\underline{p}$, the variation (4-32) due to $\delta\underline{p}$ is therefore zero for $n = \ell$. If $n = \ell$, (4-34) becomes

$$\begin{aligned} M'\tilde{Q}M &= \left[\begin{array}{c:c} I & -\underline{a} \end{array}\right] \left[\begin{array}{c} \underline{\alpha} \\ \hdashline 1 \end{array}\right] \frac{\left[\begin{array}{c:c} \underline{\alpha}' & 1 \end{array}\right]}{\|\underline{\tilde{\alpha}}\|^2} \left[\begin{array}{c} I \\ \hdashline -\underline{a}' \end{array}\right] \\ &= \frac{(\underline{\alpha} - \underline{a})(\underline{\alpha} - \underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} \end{aligned} \tag{4-35}$$

so that (4-32) is

$$\operatorname{tr}\{\delta[M'\tilde{Q}M]X\} = -2\operatorname{tr}\left[\frac{(\underline{\alpha} - \underline{a})}{\|\underline{\tilde{\alpha}}\|^2}\delta\underline{p}'A'X\right] \tag{4-36}$$

For notational convenience, define a scalar ν such that

$$\nu = \begin{cases} 0 & \text{if } n \neq \ell \\ 1 & \text{if } n = \ell \end{cases} \tag{4-37}$$

The general form of (4-32) can be written as

$$\operatorname{tr}\{\delta[M'\tilde{Q}M]X\} = -2\nu\operatorname{tr}\left[\frac{(\underline{\alpha} - \underline{a})}{\|\underline{\tilde{\alpha}}\|^2}\delta\underline{p}'A'X\right] \tag{4-38}$$

The second additional term, (4-33) can be written in terms of $\delta \underline{p}$ by defining a sensitivity matrix, B, for the pseudo IC vector as

$$B = \frac{\partial \underline{x}_0}{\partial \underline{p}} \tag{4-39}$$

and expanding (4-33) as

$$\text{tr}\ \{ [rW + P]\ 2\underline{x}_0\, \delta \underline{x}_0' - 2rW\underline{x}_{m_0}\, \delta \underline{x}_0' \} =$$

$$2\ \text{tr}\ \{ rW(\underline{x}_0 - \underline{x}_{m_0})\ \delta \underline{p}' B' + P\underline{x}_0\, \delta \underline{p}' B' \} \tag{4-40}$$

The variation in PI (4-30) due to a variation $\delta \underline{p}$ is obtained by combining (4-38) and (4-40) with (4-28), using the appropriate choice of Q.

$$\delta(\text{PI}) = \text{tr} \left\{ -2 \left[\underline{\epsilon}_n' PXA + \nu \frac{(\underline{\alpha} - \underline{a})'}{\| \underline{\alpha} \|^2} XA - \underline{x}_0' PB - r(\underline{x}_0 - \underline{x}_{m_0})' WB \right] \delta \underline{p} + [PF + F'P + M'\tilde{Q}M]\, \delta X \right\} \tag{4-41}$$

The necessary condition for a local minimum of the Model PI (4-11) is

$$FX + XF' + X_0 = 0 \tag{4-42a}$$

$$F'P + PF + M'\tilde{Q}M = 0 \tag{4-42b}$$

$$\underline{\epsilon}_n' PXA + \nu \frac{(\underline{\alpha} - \underline{a})'}{\| \underline{\alpha} \|^2} XA - \underline{x}_0' PB - r(\underline{x}_0 - \underline{x}_{m_0})' WB = 0 \tag{4-42c}$$

The value of the Model PI at this minimum can be computed from (4-13) using the solution of P from (4-42b).

The equations in (4-42) are heavily coupled in general. The quantities F, X_0, M, A, $\underline{a}$, $\underline{x}_0$, and B have explicit functional dependence on $\underline{p}$, and the quantities X and P have an implicit dependence on $\underline{p}$ given by equation (4-42). Since X and P are n × n symmetric matrices, (4-42a) and (4-42b) each represent, at most, $\frac{1}{2}n(n+1)$ linearly independent equations. Equation (4-42c) represents κ equation where κ is the number of free parameters, i. e. the dimension of $\underline{p}$. Altogether (4-42) represents, at most, $\kappa + n(n+1)$ linearly independent equations that must be solved for the elements of X, P and $\underline{p}$. Obviously that would not be an easy or enjoyable task to do by hand. Neither is it well suited to digital computation. However this development leads directly to a method for computing the gradient of the Model PI that can be used in various numerical, iterative minimization techniques.

4. 3 Direct Gradient Evaluation

The gradient of the general quadratic functional (4-2) is directly obtained from (4-28) as

$$\nabla J = \frac{\partial J}{\partial \underline{p}} = -2A'XP\underline{\epsilon}_n \qquad (4\text{-}43)$$

where X and P are evaluated by (4-29a) and (4-29b) respectively. The gradient, ∇J, is a κ × 1 vector and is a function of $\underline{p}$. In an iterative technique, ∇J_i would be evaluated at $\underline{p}_i$ on the ith iteration. Instead of using the matrix algebraic equations (4-29), X_i and P_i can be computed as the steady state solutions of linear matrix differential equations as suggested in section 4. 1. The value of the quadratic functional and its gradient on the ith iteration can be computed from

$$J_i = \|\underline{x}_0\|^2_{P_i} \qquad (4\text{-}44a)$$

$$\nabla J_i = -2A_i'X_iP_i\underline{\epsilon}_n \qquad (4\text{-}44b)$$

$$\dot{X}_i = F_iX_i + X_iF_i' + X_0 \qquad (4\text{-}44c)$$

$$\dot{P}_i = F_i'P_i + P_iF_i + Q \qquad (4\text{-}44d)$$

where F_i and A_i are evaluated at $\underline{p}_i$, and $X_i(0)$ and $P_i(0)$ are arbitrary symmetric matrices.

The extension to the Model PI can be written down immediately by applying similar logic to (4-41) and (4-42), as

$$(PI)_i = r \| \underline{x}_{0_i} - \underline{x}_{m_0} \|^2_W + \| \underline{x}_{0_i} \|^2_{P_i} \quad (4\text{-}45a)$$

$$\nabla(PI)_i = -2 \; A'_i X_i P_i \underline{\epsilon}_n + \nu A'_i X_i \frac{(\underline{\alpha} - \underline{a}_i)}{\| \tilde{\underline{\alpha}} \|^2}$$

$$- B'_i P_i X_{0_i} - r B'_i W (\underline{x}_{0_i} - \underline{x}_{m_0}) \quad (4\text{-}45b)$$

$$\dot{X}_i = F_i X_i + X_i F'_i + X_{0_i} \quad (4\text{-}45c)$$

$$\dot{P}_i = F'_i P_i + P_i F_i + M'_i \tilde{Q} M_i \quad (4\text{-}45d)$$

where $\underline{x}_{0_i}$, A_i, $\underline{a}_i$, B_i, F_i, X_{0_i} and M_i are all evaluated at $\underline{p}_i$ and $X_i(0)$ and $P_i(0)$ are arbitrary symmetric positive semi-definite matrices. For reference in the above equation, the definitions of some terms are denoted by equation number below.

Term..... defined by.....	equation number
A	(4-27)
$\underline{\epsilon}_n$	(4-26)
ν	(4-37)
B	(4-39)
W	(4-12)
M	(4-10)

Equations (4-44) and (4-45) are well suited for digital computation. They can be used in any numerical optimization technique that requires evaluation of the gradient.

In the application of (4-45) it may be easier and sufficiently accurate to compute the sensitivity matrices, A_i and B_i, numerically rather than from their functional relationship to $\underline{p}$, e. g.

$$A \approx \frac{[\underline{a}(\underline{p} + \Delta\underline{p}) - \underline{a}(\underline{p} - \Delta\underline{p})]}{2\Delta\underline{p}}$$

$$B \approx \frac{[\underline{x}_0(\underline{p} + \Delta\underline{p}) - \underline{x}_0(\underline{p} - \Delta\underline{p})]}{2\Delta\underline{p}} \qquad (4\text{-}46)$$

4.4 An Averaged Gradient Direction Optimization Algorithm

The gradient or steepest descent method is probably the simplest computational algorithm for seeking the minimum of a function over a parameter space. But it suffers severely from the "ravine problem". This is illustrated in figure 4-1, which shows contours of constant function values in the p_1 and p_2 parameter space. When the computation path crosses a ravine it tends to jump back and forth across it, making slow progress in the direction of the true minimum. Notice, however, that the average gradient direction for two steps is generally in the desired direction. The averaged gradient direction optimization algorithm presented here is based on stepping in that direction. The specific procedure is illustrated in figure 4-2. Starting at point 1, the method is to take a half step in the negative gradient direction to point 1a. The average direction of the negative gradient at points 1 and 1a is computed. Take a full step from the midpoint between 1 and 1a in the averaged direction to point 2. The actual step for this iteration is then from point 1 to point 2. Repeat at point 2, etc.

It is necessary to have some step size control so that the process will converge as it approaches the minimum. A very simple rule is used. At each point, such as 1a and 2 in figure 4-2, the function is evaluated and compared to the value at the starting point for that iteration. If the new value is smaller, proceed with the previous step size. If it is larger, cut the step size in half, evaluate the function and compare again. Repeat until the new value is smaller. Thus the new

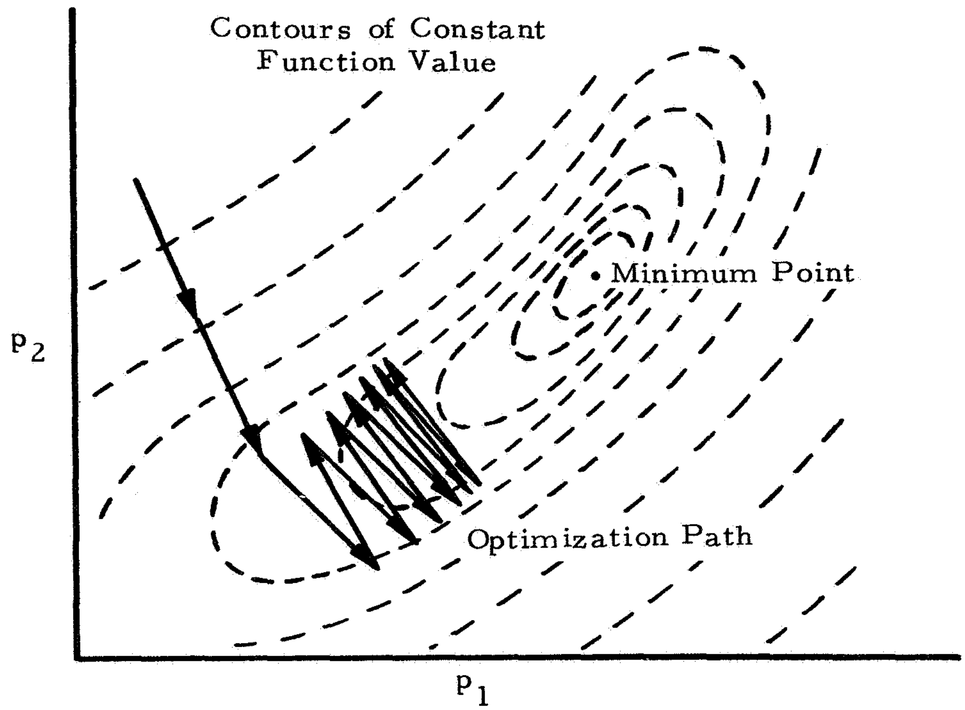


Figure 4-1 Illustration of the Ravine Problem in the Gradient Method

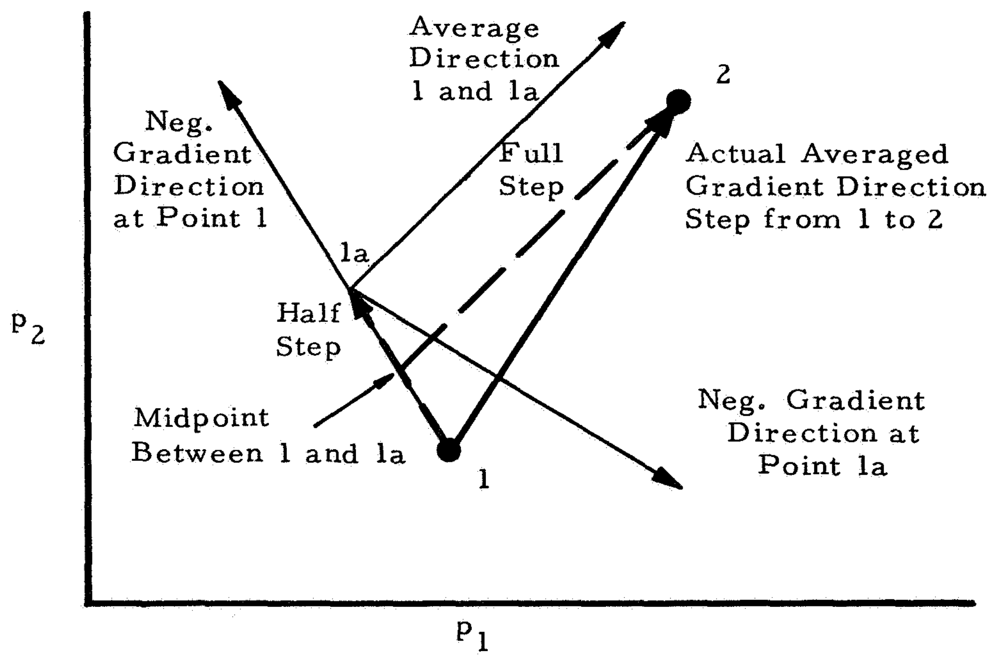


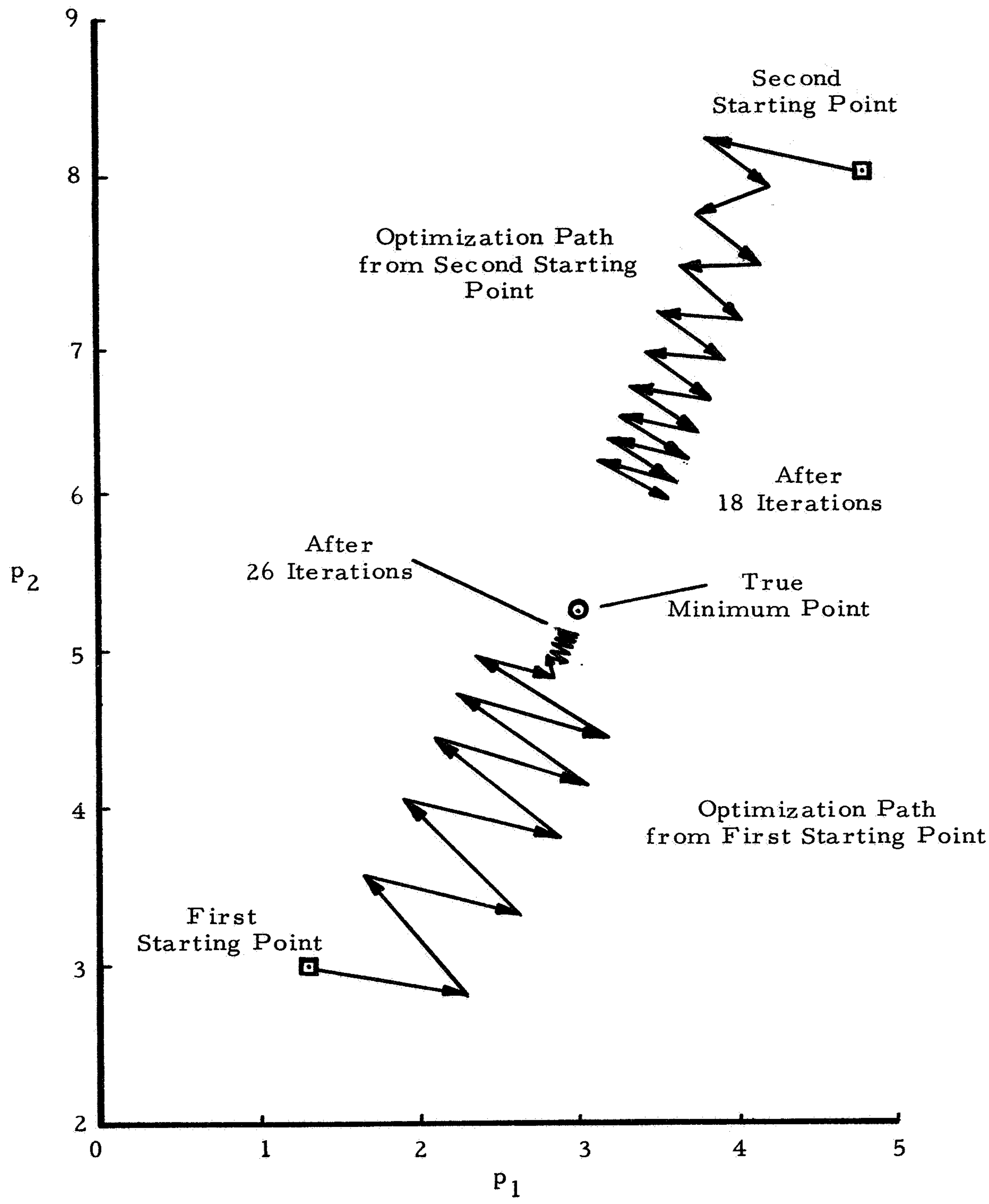
Figure 4-2 The Averaged Gradient Direction Step

iteration point always has a function value smaller than that at the old point which guarantees convergence.

The averaged gradient method has greatly improved convergence characteristics over the conventional gradient method. Figure 4-3 shows a comparison of these two for minimizing the same function for two different starting points. The function is the Model PI for the example in section 5. 1. 1 of the next chapter but its specific formulation is not important for this comparison. The conventional gradient method shown in figure 4-3a has the same type step size control as mentioned above. The ravine problem is clearly visible. In figure 4-3b, the averaged gradient direction method is shown to converge towards the true minimum much faster. A third starting point is shown in figure 4-3b to illustrate how the averaged gradient direction method works away from the ravine.

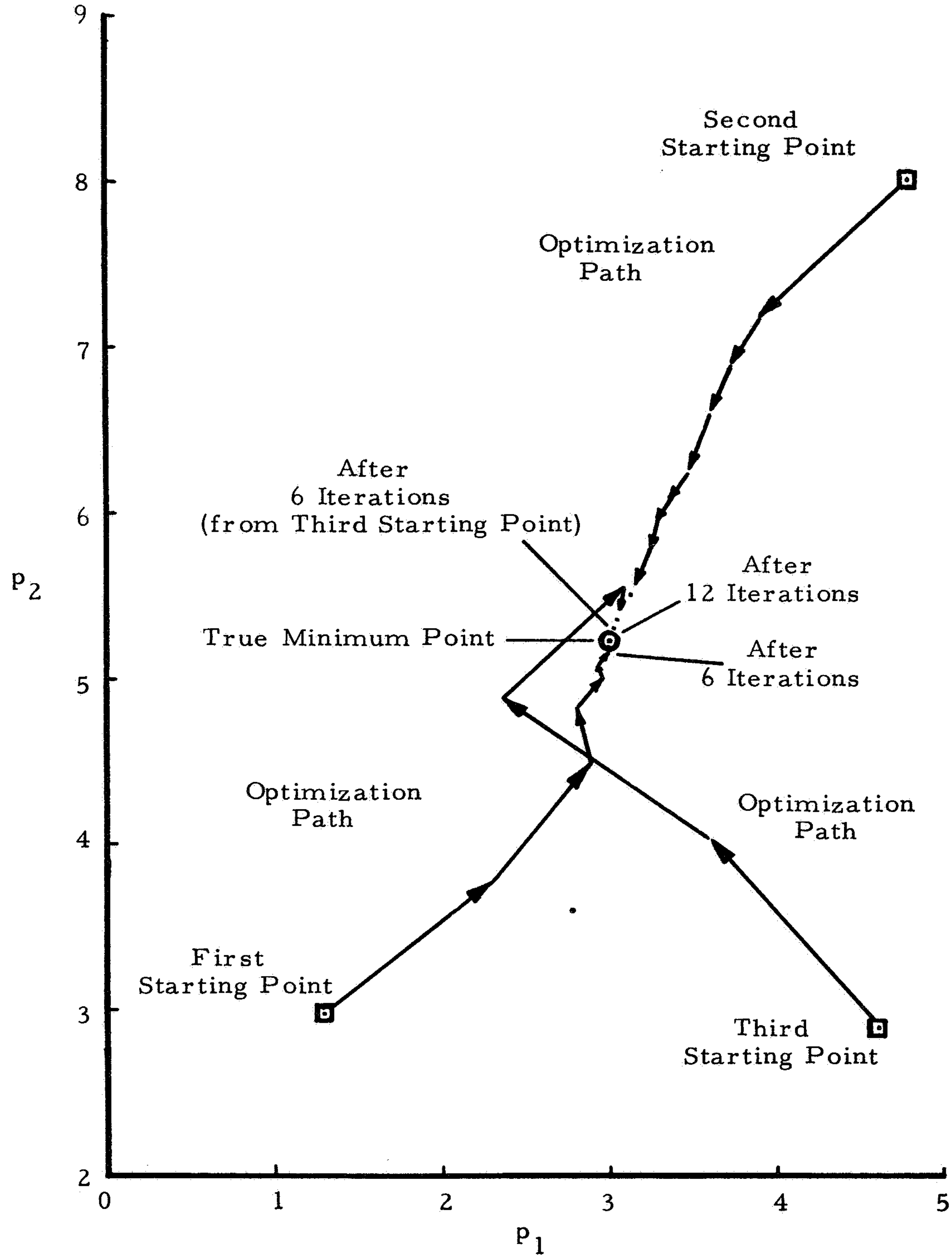
Another important point in numerical optimization algorithms is the stopping condition. Some way is needed to tell how close to the minimum is close enough. There is enough error in computing the function and its gradient to create sufficient "noise" in the vicinity of the minimum for the algorithm to continue searching for it unless given some measure by which to terminate. Two tests are used in the averaged gradient method. The program terminates if both the decrease in the performance index is less than 0. 01% between two iterations and the length of the gradient vector times the current step size is less than 0. 1% of the performance index. One might think that the first test is sufficient, but it can be satisfied by straddling a ravine at any point such that the value of the performance function is nearly the same on both sides. The second test prevents that situation from falsely stopping the program.

A simplified functional flow diagram for the averaged gradient direction optimization algorithm is shown in figure 4-4. This algorithm is used in the program described in Appendix B. In that application the function J indicated on figure 4-4 is the performance index, PI, and is computed together with its gradient, $\nabla(PI)$, by equation (4-45).



(a) Gradient Method

Figure 4-3 Comparison of the Gradient Method and the Averaged Gradient Direction Method



(b) Averaged Gradient Direction Method

Figure 4-3 Concluded

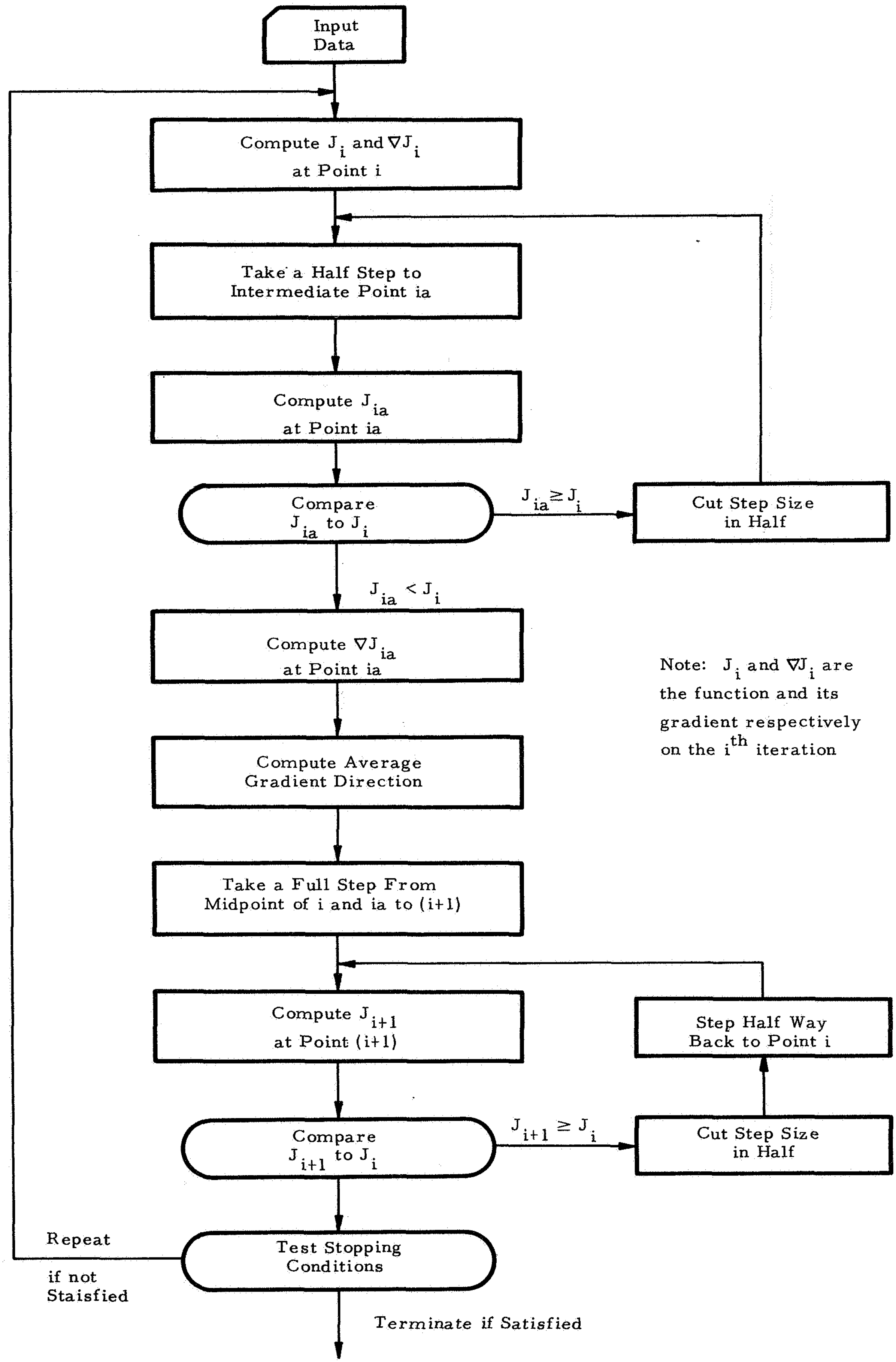


Figure 4-4 Functional Flow Diagram for Averaged Gradient Direction Optimization Algorithm

CHAPTER 5

DESIGN VIA PARAMETER OPTIMIZATION

The previous two chapters presented the theoretical development of the Model PI and some mathematical techniques necessary for its practical application. In this chapter the practical aspects of designing control systems by parameter optimization with the Model PI are considered. The step by step procedure for using the Model PI in design problems with realistic engineering design specifications is presented and demonstrated by examples. A comparison is made between the Model PI and the model-referenced ISE design methods to evaluate their relative effectiveness and computational efficiency. The model-referenced ISE technique is formulated in state-space notation in such a way that the same general computer program for control system design can be used for both techniques. The procedure for using this computer program is demonstrated in the examples presented.

In some applications the designer may want to constrain the magnitude of the free design parameters because of known physical limitations. There are several ways this can be done. Some methods for including parameter constraints in analytical design techniques are presented here although they are not unique to the Model PI approach.

This chapter treats linear control systems in general rather than specifically flight control systems. Thus a wider basis of potential application is established. It also allows for a more concise description of the basic procedures by avoiding some of the details involved in formulating a specific flight control system design problem. Accordingly, the engineering design specifications considered here are of the forms more common to general control system design problems (section 2. 1. 2) rather than those related strictly to the design of flight control systems (section 2. 1. 1).

The basic design procedure is presented for single input/output systems. Extension to multivariable systems should be clear from the discussion here and in section 3. 2. 3 of Chapter 3. A multivariable system design example is presented in the next chapter that should help clarify any remaining questions.

5. 1 Control System Design Using the Model PI

The procedure for using the Model PI to design linear control systems follows the general procedure for analytical design discussed in Chapter 2, section 2. 2. It is assumed here that the design specifications for the dynamic response of the closed-loop system are given in one of the two standard graphical representations discussed in section 2. 1. 2. Design examples for both the time domain and frequency domain forms are presented. Starting with these engineering design specifications, the basic Model PI design method proceeds as outlined in the following steps:

1. Select a linear model to represent the dynamic response specifications.
2. Select a compensation configuration for the control system.
3. Form the closed-loop transfer function as a function of the free design parameters.
4. Apply the general computer program for control system design (Appendix B).
5. Compare the resulting closed-loop design to the engineering specifications. If they are not satisfied repeat steps 3 to 5 with a different compensation configuration.

Some variations in these steps, especially step 5, may be necessary for particular design problems.

The first step, selection of an appropriate model, is not a trivial task and should be given more serious attention in the analytical design process than it has in the past. Li and Whitaker (41) discuss some aspects of selecting models for characterizing the performance of adaptive control systems that would also apply to the analytical design problem for non-adaptive systems. But often the model is taken to be the starting point in analytical design processes, completely ignoring the fact that practical design specifications must include a tolerance. The model per se is not "the" specification.

In the Model PI, the model is used as a guide to the actual engineering design specifications which are given as a range of allowable closed-loop responses, e.g. figures 2-5 and 2-6. The objective is to select a model that has a good chance of forcing the closed-loop system, via the optimization process, to have a response that lies within the allowable range. The designer would like a guarantee that the model he uses would indeed produce such a design. Unfortunately there is, as yet, no set of rules for selecting a model that can guarantee this, whether using the Model PI or any other model-referenced performance index. However some general guidelines can be deduced from the nature of the Model PI that can help in selecting an appropriate model. The most obvious one is to consider only models with response curves lying within the response specification tolerance. This is not a rigid requirement. It is possible, at least in some examples, to use a model that is partially outside the response tolerance envelope and still produce a design meeting the specifications. But such a model is inconsistent with the mathematical and physical motivations for the Model PI (Chapter 3) and can not be recommended as a general rule.

The next guideline is to select a model, from among those meeting the first guideline, that is most likely to be matched by one of the possible closed-loop system designs. The motivation for this is obvious. The more likely it is that some closed-loop system design will match the model chosen, the more likely it is that the specifications will be met. Finding a model that could be matched exactly by the system would generally be very time consuming and might be tantamount to designing the system itself. On the other hand, using the lowest order model that fits within the tolerance envelope when designing a much higher order system gives very little assurance that a satisfactory design would result, unless its response is completely dominated by a low order mode. A practical choice for a model would generally lie somewhere between these two extremes.

Several other guidelines for selecting models were suggested in Chapter 3 specifically for the Model PI. These were summarized in table 3-5 together with the appropriate form of the Model PI to use in various situations.. Table 3-5 is a convenient reference for applying the Model PI. The model must be put into the geometrical representation form indicated in table 3-4 to be used in the Model PI.

The second step in the design procedure, that of selecting a compensation configuration, depends heavily on the designer's experience in control system synthesis. The basis for selecting a compensation configuration is the same here as it is in conventional linear servo design techniques (9), and experience with these is directly applicable. But in parameter optimization techniques only the form of the compensation is selected by the designer, and the values of the free design parameters, i. e. the loop gain and compensation time constants, are selected by the optimization process. One would generally start with the simplest form of compensation that might give an acceptable design and only add to it if the simpler form is inadequate. It is assumed that the reader has sufficient familiarity with conventional techniques to select a compensation configuration, and so this step will not be discussed further.

Once the compensation configuration is selected, the next step is to form the closed-loop transfer function in terms of the free design parameters. That is, determine the numerator and denominator coefficients of the closed-loop transfer function as algebraic functions of the free design parameters. The problem is then ready for the general computer program for control system design using the Model PI, which is step 4.

A description and complete listing of the computer program developed for this thesis effort are presented in Appendix B. To use the program for any specific single input/output control system design problem only requires providing the appropriate input data cards and writing one simple subroutine, COEF, that merely lists the functional relationships of the free parameters to the system transfer function coefficients. The model's characteristics, several constants needed in the numerical optimization process, and an initial choice for the free parameters enter the program by input data cards. The input data format and program output are described in sections B. I. 1 and B. 1. 2 of Appendix B respectively. The detailed procedure for using the program will be demonstrated in the examples to follow subsequently.

Two general remarks should be made about the initial choice for the free parameters in relationship to this computer program. First of all, the optimization algorithm is for finding a local minimum only so that if several relative minima exist in a given problem the initial choice of the free parameters determines which of these minima the algorithm will seek. Secondly the procedure for evaluating the Model PI and its gradient involves numerically integrating two linear matrix equations to their steady state values (see Chapter 4). The initial choice of the parameters must be such that a suitable approximation to these steady state values can be computed within a reasonably short computational time. This requires that the closed-loop system corresponding to the initial parameter values be asymptotically stable.

The last step in the Model PI design procedure is to determine if the closed-loop system design resulting from the optimization process meets the engineering specifications. If the system response lies within the tolerance envelope the design problem is complete. At this point it is immaterial whether the system response closely matches the model response or not as long as it satisfied the engineering specifications. If they are not satisfied, then the design process is repeated with some alteration that is likely to result in a better design. The alteration would generally be in the form of a different compensation configuration. When using the general Model PI (3-82) for systems with zeros, there is a scalar constant, r, that determines the relative weighting between the quadratic penalty on the vector error in pseudo IC's and the quadratic functional. In that case it may be necessary to repeat the design process for an alternate value of r (with the same compensation configuration) to obtain an acceptable design. In multivariable system design problems other alterations are possible but those will be discussed in Chapter 6 where an application of the Model PI method to multivariable system design is considered.

The following examples are presented to illustrate the basic Model PI design procedure and the use of the aforementioned general digital computer program.

5.1.1 Design Example for Step Response Specifications

Consider the open-loop system shown in a functional block diagram in figure 5-1.

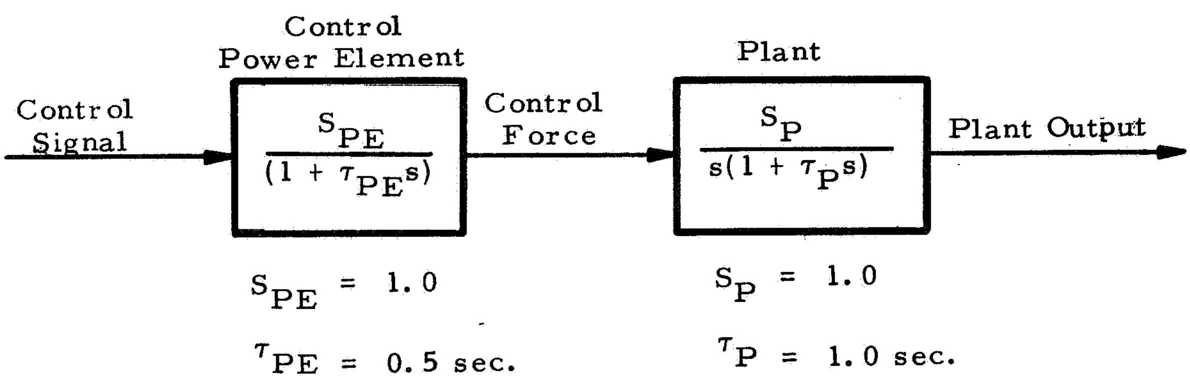


Figure 5-1 Functional Block Diagram of Open-Loop System

The control power element has already been selected and is not free to be altered in the design process. The closed-loop system is to be a positional servo that has a step response lying within the design specification envelope shown in figure 5-2.

The first step in the Model PI design procedure is to select a model to represent the design specifications. According to table 3-5 the best choice for this example would be a third order model with no zeros since the system is third order with no zeros. It actually isn't

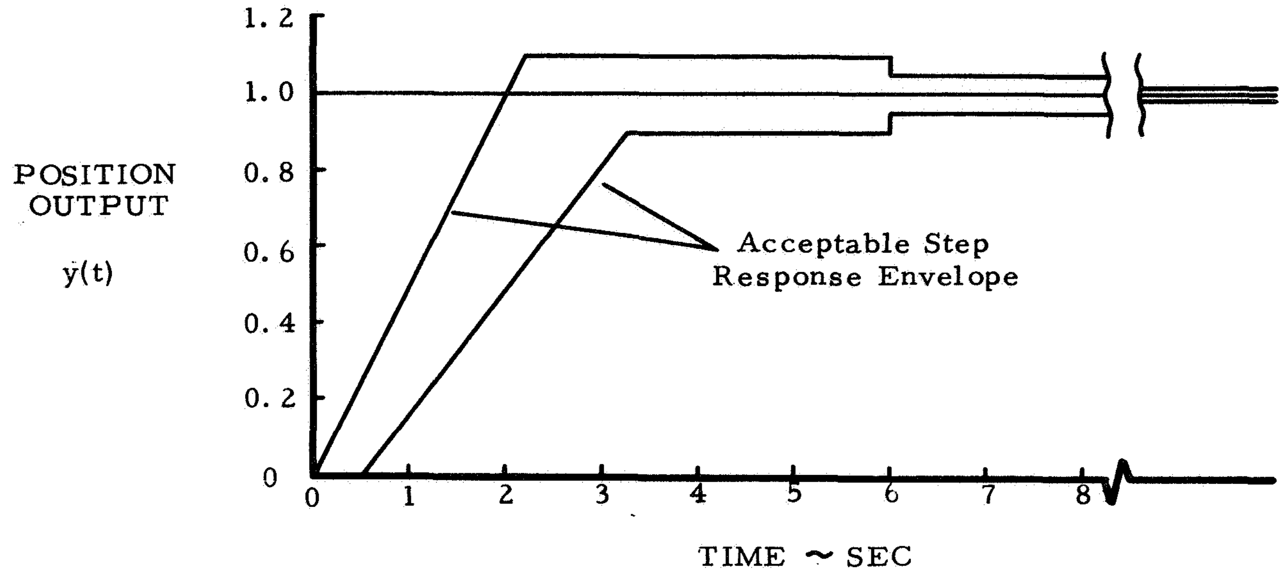


Figure 5-2 Step Response Design Specification.

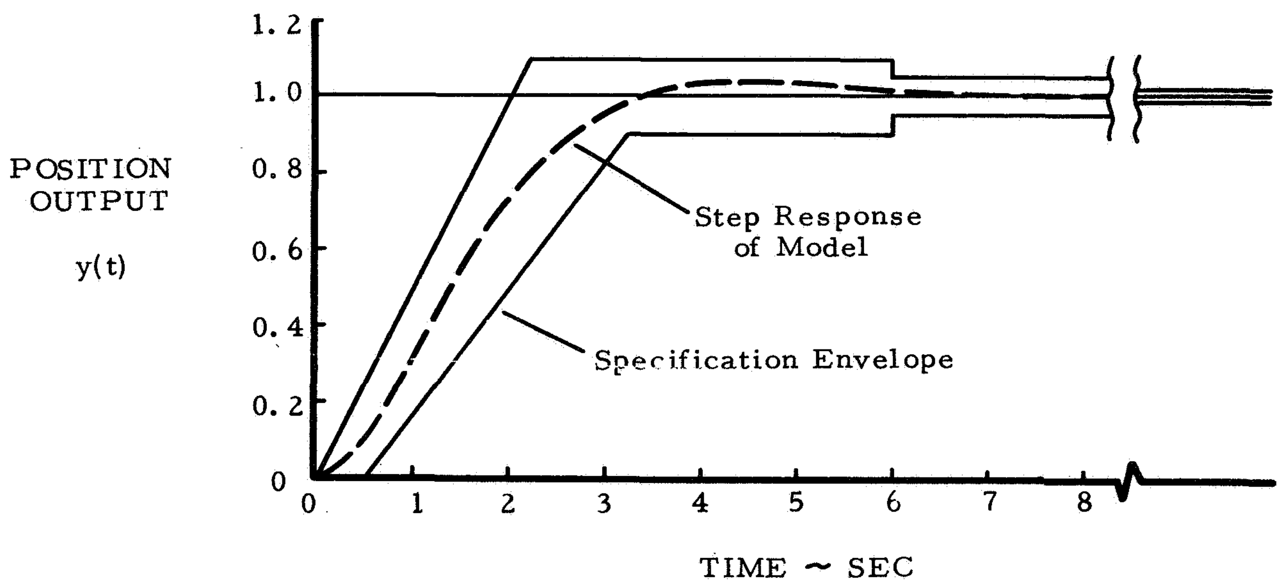


Figure 5-3 Step Response of a Second Order Model To Represent the Design Specification

very difficult to establish a third order model that would lie within the design envelope if one uses non-dimensionalized graphs of third order step responses, e.g. those found in reference 49. But a second order model will be used here to illustrate the use of a lower order model. In this example no first order model lies within the envelope so a first order model would not be considered. The second order model given by

$$\frac{y_m(s)}{u(s)} = \frac{1}{s^2 + \sqrt{2}\, s + 1} \tag{5-1}$$

lies almost in the center of the step response specification envelope, as shown in figure 5-3, and thus appears to be a good choice. It is reasonable to expect that if the closed-loop system design is to meet the specifications its step responses would have to be fairly close to that of this model. Table 3-5 indicates that for this type of model the form of the Model PI to use is

$$PI = \int_0^\infty \| \underline{\tilde{x}}(t) \|^2_{\tilde{Q}}\, dt \tag{5-2}$$

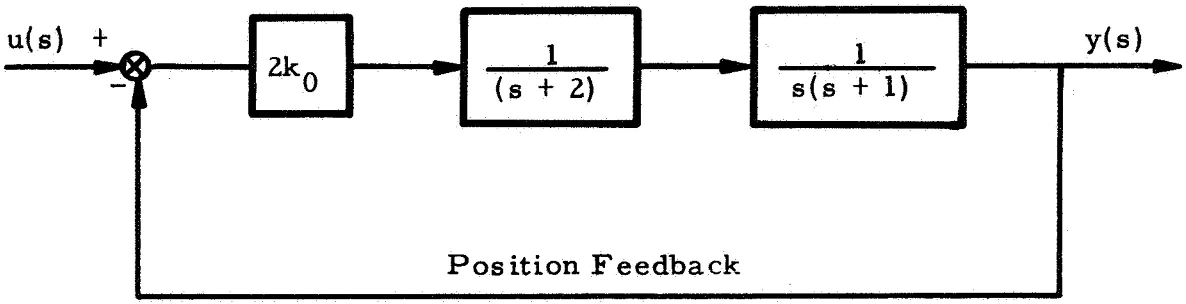
The model's extended coefficient vector, $\underline{\tilde{\alpha}}$, which defines $\tilde{Q}$, for the model (5-1) is

$$\underline{\tilde{\alpha}}' = [\,1 \quad \sqrt{2} \quad 1 \quad 0\,] \tag{5-3}$$

It isn't necessary to compute the model's pseudo IC vector in this case, as indicated in table 3-5.

The simplest type of feedback (compensation) that might provide an acceptable design is a position feedback. A mathematical block diagram of a position feedback configuration is shown in figure 5-4, where the free design parameter, k_0, is the system loop gain. The closed-loop transfer function for the position feedback system is

$$\frac{y(s)}{u(s)} = \frac{2k_0}{s^3 + 3s^2 + 2s + 2k_0} \tag{5-4}$$



k_0 is a free design parameter

Figure 5-4 Block Diagram of Closed-Loop System with Position Feedback

The problem is now ready for the computer program, which will find the value of k_0 that minimizes the Model PI. For this design example the proper subroutine COEF corresponding to (5-4) is written in FORTRAN-IV as

```
SUBROUTINE COEF(ACOF, BCOF, PAR)
DIMENSION ACOF(1), BCOF(1), PAR(1)
BCOF(1)  =  PAR(1)
ACOF(1)  =  PAR(1)
ACOF(2)  =  2.0
ACOF(3)  =  3.0
RETURN
END
```

where PAR(1) is $2k_0$, BCOF(1) is the numerator coefficient of (5-4) and ACOF(1), I = 1, 2, 3, are the denominator coefficients of (5-4). The input data cards are as follows (see section B. 1. 1 for the format and definition of terms):

First Card (Constants used in numerical optimization algorithm)

N,	M,	K,	STEP,	ITMAX,	H,	IMAX,	SNE,	YNE,	RXO,	RPC,	LI
3,	0,	1,	1.0,	50,	0.1,	200,	0.0,	0.001,	-0.0,	-0.0,	3

Second Card (Initial choice for k_0)

PAR(1)

0.1

The next two cards are blank since no quadratic penalty on the pseudo IC vector error is used.

Fifth Card (Model's extended coefficient vector)

ALPHA(I), I = 1, 3

1.0 1.414 1.0

Using these input data cards and COEF in the computer program gave the solution $k_0 = 0.68$, which corresponds to a closed-loop transfer function

$$\frac{y(s)}{u(s)} = \frac{1}{\left(1 + \frac{1}{2.4} s\right)\left[1 + \frac{2(0.4)}{(0.75)} s + \left(\frac{s}{0.75}\right)^2\right]} \tag{5-5}$$

The step response for this solution is compared to the specification envelope in figure 5-5, which shows that it does not meet the specifications. The response is somewhat slow and has too little damping. Position feedback alone is not sufficient.

Adding a velocity feedback is the logical next step for improving the system response. Velocity and position feedbacks, as shown in figure 5-6, would allow independent adjustment of the rise time and damping. The Model PI design procedure will be repeated using the same model but with this new feedback configuration.

The closed-loop transfer function for the position and velocity feedback system is

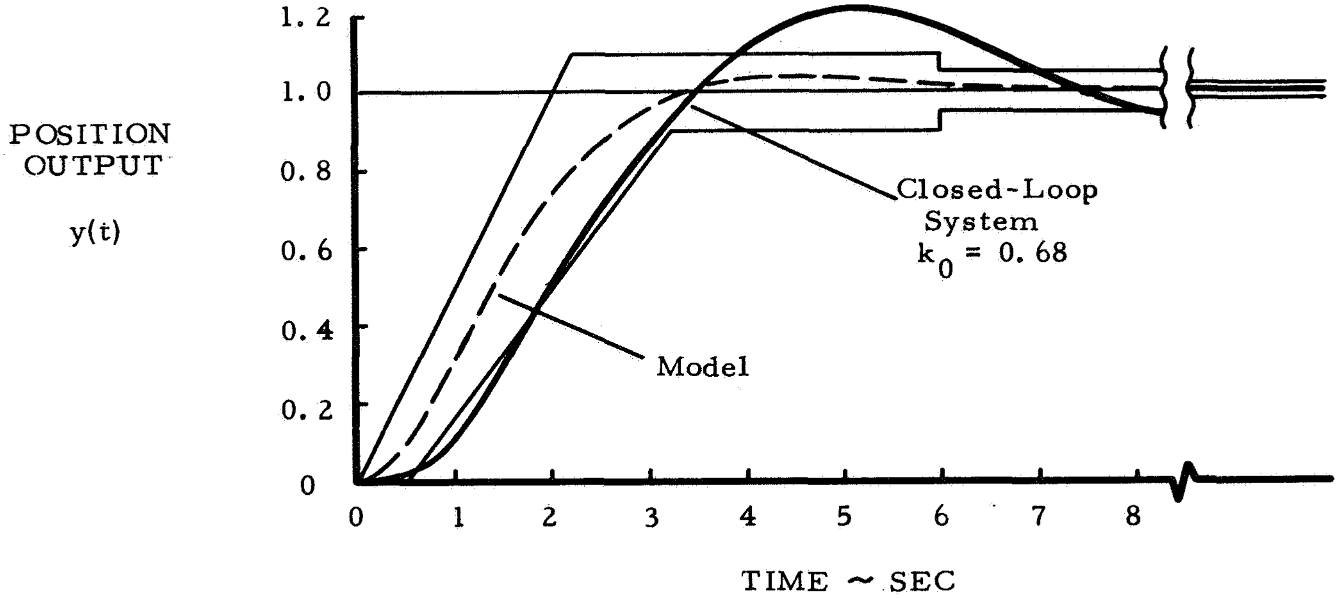


Figure 5-5 Comparison of the Model PI Design Step Responses to That of the Model and Specification Envelope for Position Feedback

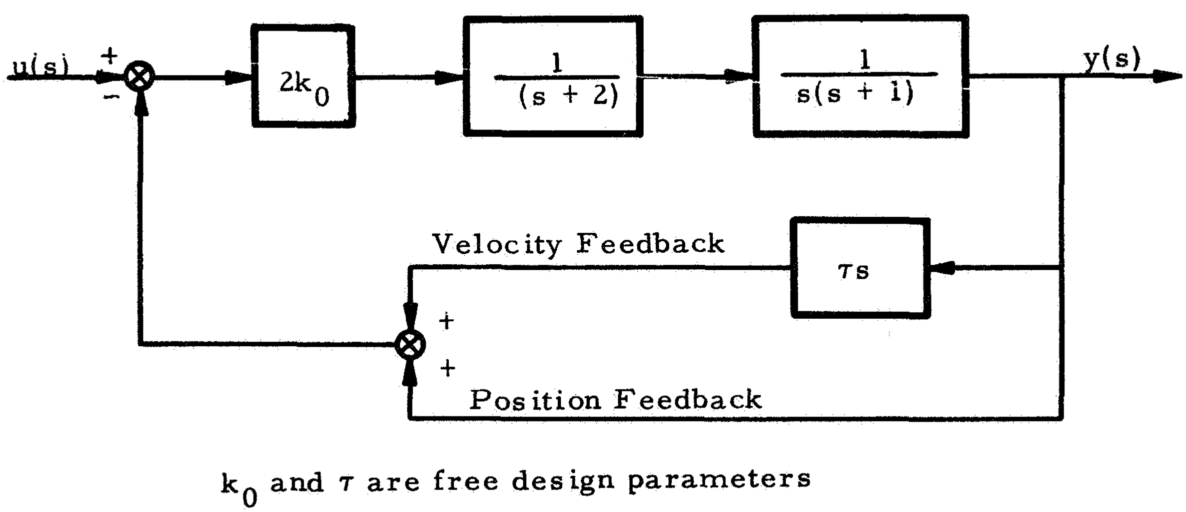


Figure 5-6 Block Diagram of Closed-Loop System with Position and Velocity Feedback

$$\frac{y(s)}{u(s)} = \frac{2k_0}{s^3 + 3s^2 + 2(1 + k_0\tau)s + 2k_0} \qquad (5\text{-}6)$$

The only changes required to use the computer program for this case are in the subroutine COEF and the first two data cards. In the subroutine COEF listed above change ACOF(2) to

ACOF(2) = PAR(2)

where PAR(2) is $2(1 + k_0\tau)$. One could use PAR(2) to be τ with ACOF(2) = 2.0 + PAR(1)*PAR(2) just as well. On the first data card change K, the number of free parameters, from 1 to 2 and on the second card add an initial choice for the second free parameter, PAR(2).

The resulting computer solution* gave PAR(1) $\doteq$ 2.99 and PAR(2) = 5.21 which corresponds to $k_0 = 1.5$ and $\tau = 1.07$. The closed-loop transfer function for this solution is

$$\frac{y(s)}{u(s)} = \frac{1}{\left(1 + \frac{1}{0.9}s\right)\left[1 + \frac{2(0.57)}{(1.83)}s + \left(\frac{s}{1.83}\right)^2\right]} \qquad (5\text{-}7)$$

Figure 5-7 shows that the step response for this Model PI design of the position and velocity feedback system does meet the design specifications. The design problem is complete.

There is sometimes a tendency at this point in the design process to judge the design obtained relative to the model even though it satisfied the specifications. For example, in analyzing figure 5-7 one could argue that some other design might match the step response of the model better than the Model PI solution shown. That may indeed be true since figure 5-7 only compares the position time response, and the Model PI in this example attempts to match the response trajectories in a three dimensional space, i.e. it tries to match position, velocity and acceleration.

---

* This example was used in Chapter 4 to compare the optimization algorithm used to the gradient method. The optimization paths from three starting points to this solution are shown in figure 4-3b.

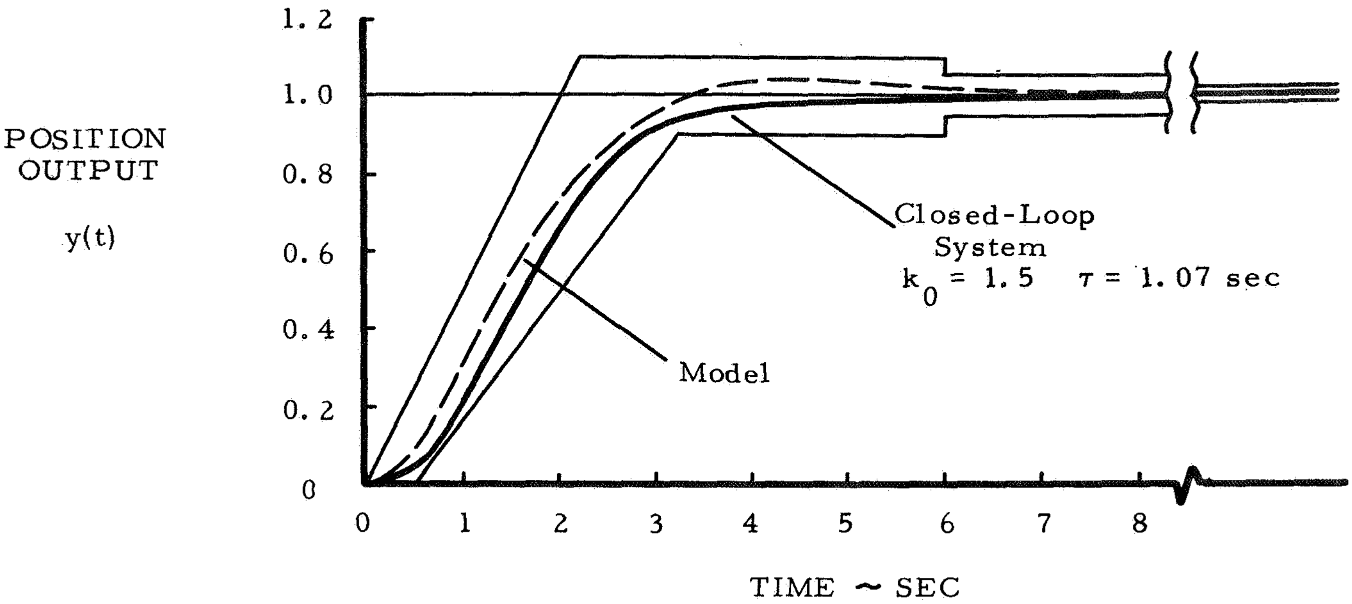


Figure 5-7 Comparison of the Model PI Design Step Response to that of the Model and Specification Envelope for Position and Velocity Feedback

When comparing all three of these for the model and the Model PI design in figure 5-8 one can see that the Model PI produces a reasonable compromise in matching position, velocity and acceleration. Matching the model's position time response closer would produce larger errors in velocity and acceleration.

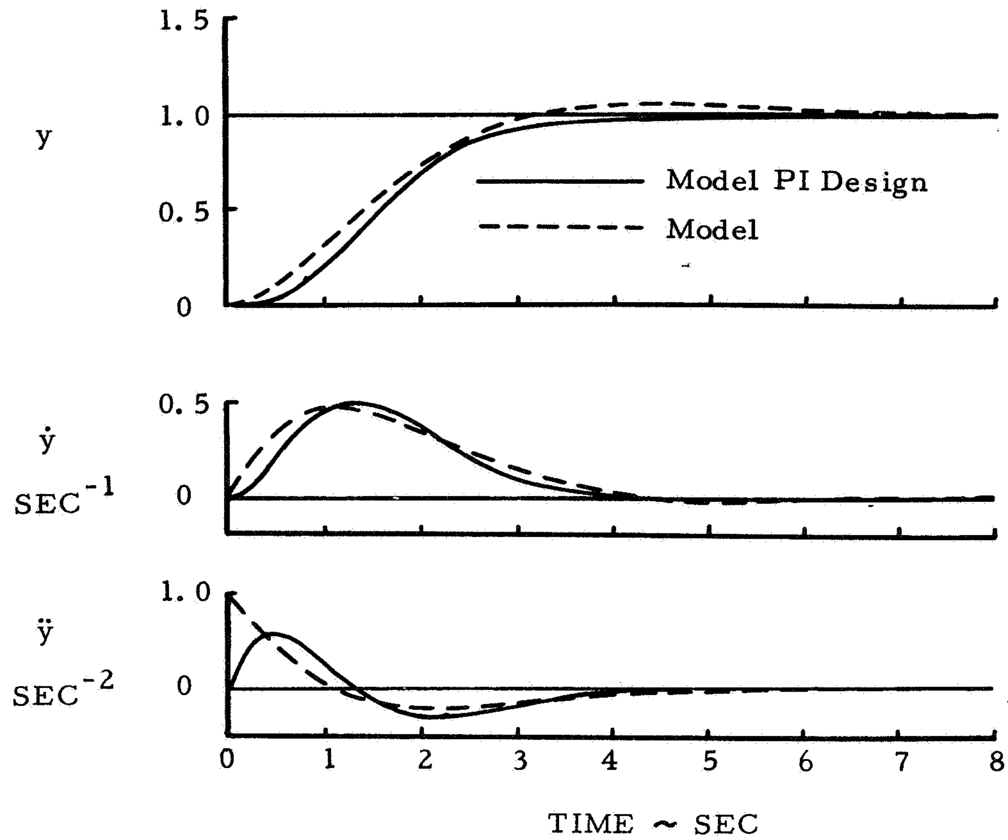


Figure 5-8 Step Response Comparison of Position, Velocity and Acceleration for the Model and Model PI Design

5.1.2 An Example for Frequency Response Specifications

When the specifications are given in the frequency domain, the Model PI design procedure is the same as that illustrated in the previous section except for the method of establishing a model and comparing the resulting design to the specifications. Once a model is selected it doesn't matter in the optimization process whether it was based on time or frequency domain specifications. However the resulting design has to be compared to and satisfy the specifications in the form given. These differences are relatively minor and can be illustrated by considering the same design problem as section 5.1.1 but with frequency domain specifications.

A feedback control is to be designed for the system shown in figure 5-1 to form a positional servo that has a frequency response within the envelope specified in figure 5-9.* The amplitude frequency response envelope is a sufficient specification unless the system is non-minimum phase. It is easier in general to establish higher order models

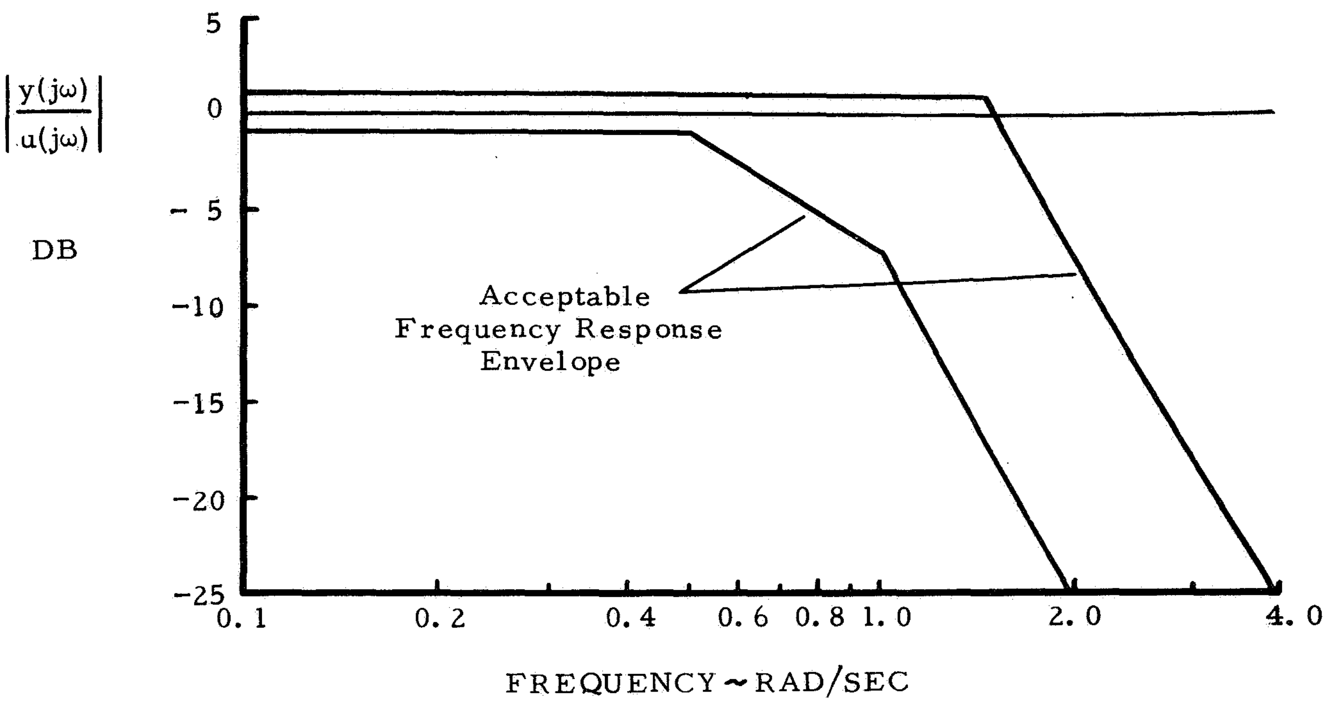


Figure 5-9 Frequency Response Design Specification

---

* This design specification envelope was chosen specifically for this example so that the model used in section 5.1.1 would also be a logical choice here, which allows the desired points to be illustrated with the minimum of redundant discussion.

for this type of specifications than for step response specifications because the asymptotes to the frequency response can be used as a guide (Bodes's Theorem). Even so, the task becomes increasingly more difficult as the model's order is increased. The breakpoints of the asymptotes and the damping ratio of oscillatory modes completely define the model's transfer function, excluding non-minimum phase models, which is directly transformable into the extended coefficient vector and pseudo IC vector needed in the Model PI. Since normalized frequency response curves for first and second order systems are readily available, e.g. references (9), (14), (15), and (44), it is wise to consider one of these forms first as a possible model unless it is obviously unappropriate. The frequency responses of a first and second order system are plotted in figure 5-10 together with the design specification envelope for this example. The first order system does not appear

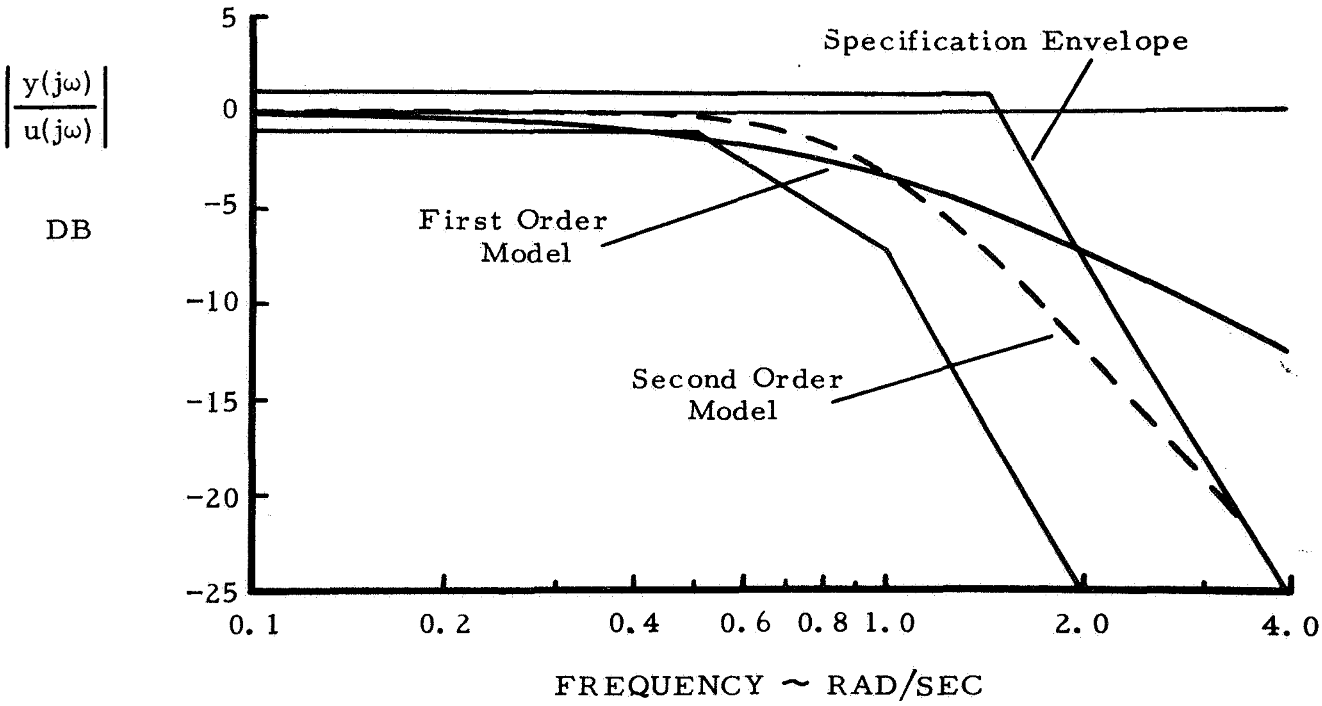


Figure 5-10 Comparison of the Frequency Responses of First and Second Order Models to Specification Envelope

to be a good choice for the model. If the response of the closed-loop control system closely matched this first order response over the frequency range shown, the design specifications would not be satisfied. The second order system represented in figure 5-10 would be a reasonable model over the frequency range considered. Therefore it isn't necessary to go to a third order model in this case. The second order system in figure 5-10 is the same model (5-1) used in section 5.1.1 Using the same model here means that the mechanics of the Model PI design procedure are identical to the previous example up to the point of comparing the resulting designs to the specifications.

Consider the case in which only position feedback is used. The Model PI solution resulted in the closed-loop transfer function (5-5). The corresponding frequency response is compared to the frequency domain specifications in figure 5-11. The peaking-up of the response

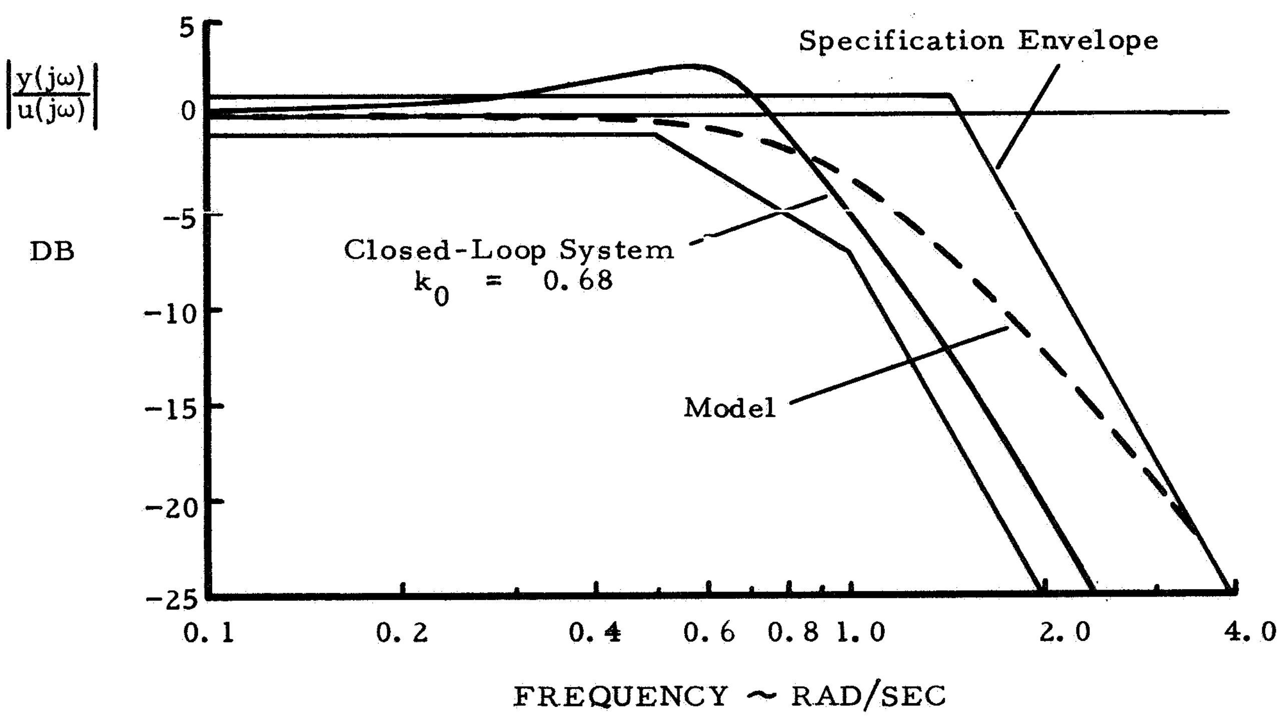


Figure 5-11 Comparison of the Model PI Design Frequency Response to That of the Model and Specification Envelope for Position Feedback

above the envelope indicates that the system is too lightly damped and thus position feedback alone is inadequate. Adding a velocity feedback is again the appropriate action for improving the response. The frequency response of the Model PI design for position and velocity feedback (5-7) is compared to the specification envelope in figure 5-12. It satisfied the specifications so the design problem is complete.

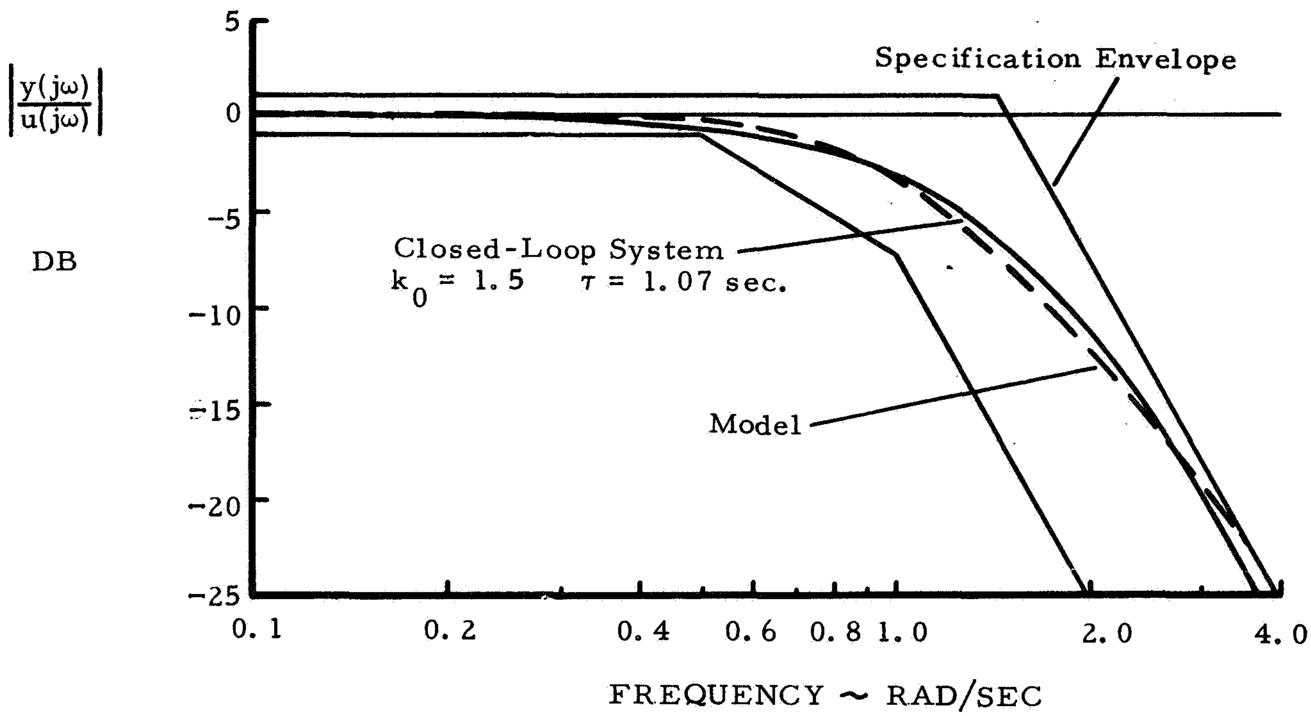


Figure 5-12 Comparison of the Model PI Design Frequency Response to That of the Model and Specification Envelope for Position and Velocity Feedback

It is interesting to note that the Model PI solution has a frequency response that matches the model's frequency response very well in figure 5-12. Even in figure 5-11, where the design is unsatisfactory, the Model PI solution tends to balance matching the model's frequency response in the low and high frequency ranges. This imperical observation will be discussed in section 5.2 in comparing the

Model PI and model-referenced ISE design methods.

5.2 Comparison to the Model-Referenced ISE Method

The basic concept of the model-referenced ISE method for designing linear control systems is illustrated by the functional block diagram in figure 5-13. In this approach the performance index is a measure of the error between the time response of a linear model, representing the desired closed-loop response, and the actual closed-loop system time response. The input to the system is also fed into the model and the two resulting outputs are compared to form an error signal. The performance index is taken to be the integral of the squared error over the time interval 0 to ∞. Minimizing this ISE with respect to the free design parameters tends to make the system's time response match that of the model.

This process can be implemented directly for each specific design problem on an analog or hybrid computer, e.g. see references 4, 22, 23 and 24. The parameter adjustment procedure for minimizing the ISE is usually mechanized as some form of steepest descent. Each time a test input signal, such as a unit step, is applied the local gradient of the ISE with respect to the free parameters is computed. The parameters are adjusted during the transient in the direction that would decrease the ISE. This process is repeated until the value of the ISE is judged by the designer to be sufficiently close to a minimum point, or, equivalently, the gradient of the ISE is sufficiently close to zero. The distance traveled towards a minimum point during each transient, i.e. the step size, is usually adjusted manually by the designer to give good convergence and prevent a possible unstable parameter adjustment loop. High speed, repetitive operation computers are ideally suited for this procedure. Once the design problem is mechanized and checked out on the computer, minimizing the ISE for the particular compensation configuration chosen can be done rather quickly.

There are several ways to implement the concept represented in figure 5-13 on a digital computer. One way is to use a digital computer program equivalent to the above analog procedure with some logic

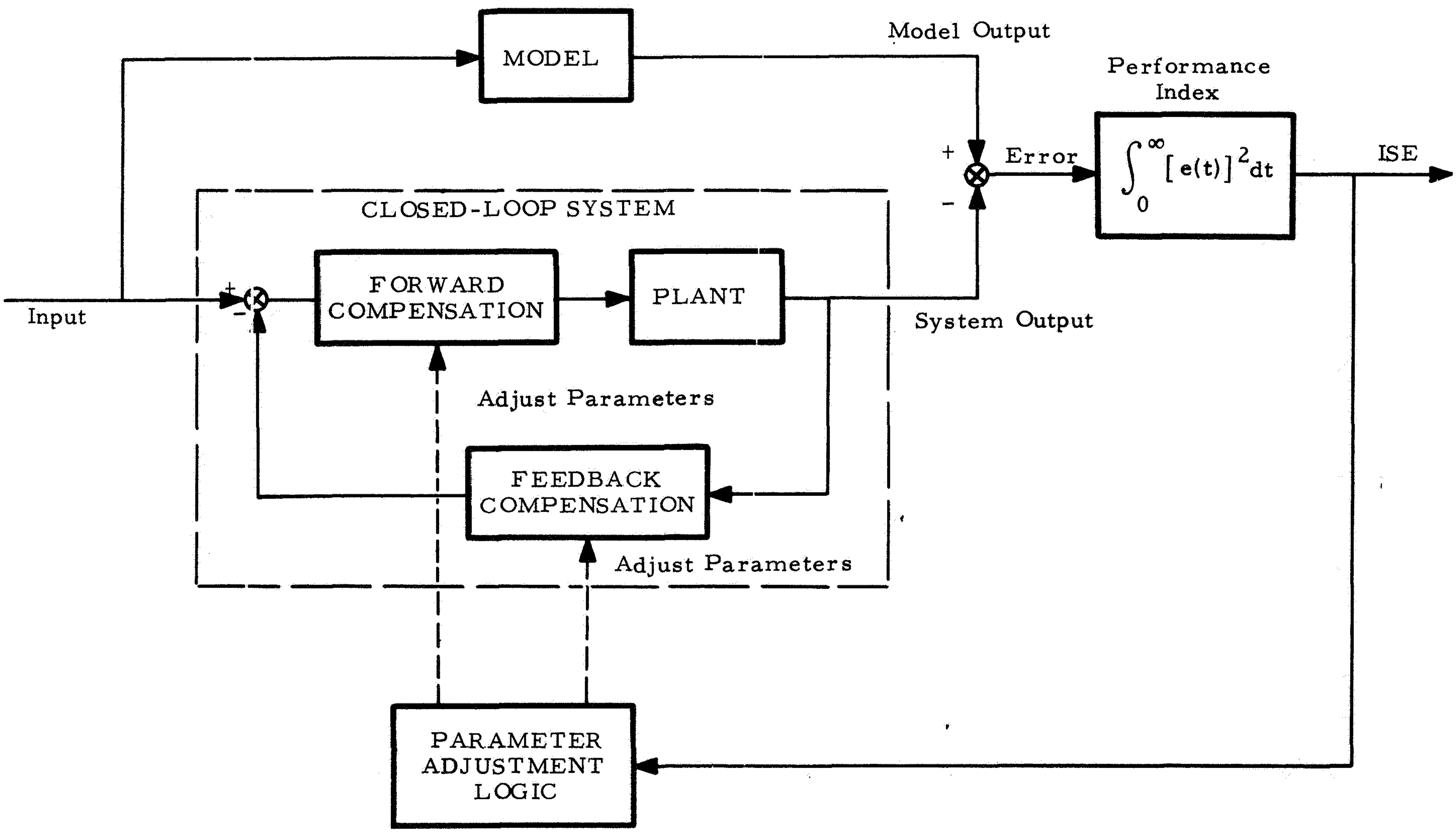


Figure 5-13 Functional Block Diagram of the Model-Referenced ISE Method

added for step size control and the stopping condition. A second way is to follow the approach described in Newton, Gould and Kaiser (15) to first obtain an algebraic function for evaluating the ISE in terms of the free parameters, then use a numerical optimization algorithm (21) to find a minimum point of the algebraic function with respect to the parameters. This second method is probably the one most often associated with control system design by parameter optimization. Both of these digital computer methods and the analog or hybrid computer method require reprogramming for each new design problem. However, by formulating the model-referenced ISE design method in state-space form and by appropriately redefining certain terms, it is possible to use the general digital computer program described in Appendix B for numerous design problems. It only requires changing two simple subroutines and the input data cards to change from one design problem to another when using the model-referenced ISE performance index. In addition to simplifying the implementation, this provides a common basis for comparing the Model PI and model-referenced ISE design methods.

The state-space formulation of the model-referenced ISE concept is presented in the following section and related to the general computer program. Then, in section 5.2.2, a control system example is designed by both the Model PI and model-referenced ISE methods using the computer program listed in Appendix B. The results are compared and discussed along with some general subjective comments on the two design methods in section 5.2.3.

5.2.1 State-Space Formulation of the Model-Referenced ISE Method

The closed-loop control system and the model can be represented as autonomous systems in state-space notation (see section 3.1) as

$$\dot{\underline{x}}_s(t) = F_s \underline{x}_s(t) \tag{5-8a}$$

$$y_s(t) = \underline{h}_s' \underline{x}_s(t) + y_{ss} \tag{5-8b}$$

$$\underline{x}_s(0) = \underline{x}_{s_0} \tag{5-8c}$$

and

$$\dot{\underline{x}}_m(t) = F_m \underline{x}_m(t) \qquad (5\text{-}9a)$$

$$y_m(t) = F_m \underline{x}_m(t) + y_{ss} \qquad (5\text{-}9b)$$

$$\underline{x}_m(0) = \underline{x}_{m_0} \qquad (5\text{-}9c)$$

respectively, where the subscripts s and m denote the closed-loop system and model respectively, the scalars $y_s(t)$ and $y_m(t)$ are the output variables, y_{ss} is the steady-state value of both $y_s(t)$ and $y_m(t)$, and the vectors $\underline{h}_s$ and $\underline{h}_m$ are defined as

$$\begin{aligned} \underline{h}'_s &= [\, 1 \;\vdots\; \underline{0}' \,] \\ \underline{h}'_m &= [\, 1 \;\vdots\; \underline{0}' \,] \end{aligned} \qquad (5\text{-}10)$$

with the null vectors appropriately dimensioned. The coefficient matrices, F_s and F_m, are in the phase variable form (3-11) and the pseudo IC vectors, $\underline{x}_{s_0}$ and $\underline{x}_{m_0}$ are of the form (3-14). The closed-loop system is nth order and the model is ℓth order.

Define a partitioned, $(n + \ell) \times 1$ state vector, $\underline{x}(t)$, as

$$\underline{x}(t) = \begin{bmatrix} \underline{x}_m(t) \\ \hdashline \underline{x}_s(t) \end{bmatrix} \qquad (5\text{-}11)$$

Equations (5-8a) and (5-9a) can be combined and written in terms of $\underline{x}(t)$ as

$$\dot{\underline{x}}(t) = F\underline{x}(t) \qquad (5\text{-}12)$$

where F is an $(n + \ell) \times (n + \ell)$ partitioned matrix

$$F = \left[\begin{array}{c|c} F_m & O \\ \hline O & F_s \end{array}\right] \tag{5-13}$$

and the O's are appropriately dimensioned null matrices. The pseudo IC vector for (5-12) is

$$\underline{x}_0 = \left[\begin{array}{c} \underline{x}_{m_{\underline{0}}} \\ \hline \underline{x}_{s_0} \end{array}\right] \tag{5-14}$$

The error, e(t), between the model and closed-loop system output variables, i. e.

$$\begin{aligned} e(t) &= y_m(t) - y_s(t) \\ &= \underline{h}'_m \underline{x}_m(t) - \underline{h}'_s \underline{x}_s(t) \end{aligned} \tag{5-15}$$

can be written in terms of $\underline{x}(t)$ as

$$e(t) = \underline{h}'x(t) \tag{5-16}$$

where

$$\underline{h}' = [\underline{h}'_m \mid -\underline{h}'_s] \tag{5-17}$$

The model-referenced ISE performance index is defined as

$$I = \int_0^\infty [e(t)]^2 \, dt \tag{5-18}$$

Using (5-16) in the above definition gives

$$I = \int_0^{\infty} [\underline{h}'\underline{x}(t)]^2 \, dt = \int_0^{\infty} [\underline{x}'(t)\,\underline{h}\,\underline{h}'\underline{x}(t)] \, dt \tag{5-19}$$

Then defining an $(n + \ell) \times (n + \ell)$ matrix Q as

$$Q = \underline{h}\,\underline{h}' \tag{5-20}$$

the model-referenced ISE performance index can be written as

$$I = \int_0^{\infty} \| \underline{x}(t) \|_Q^2 \, dt \tag{5-21}$$

The optimization portion of the model-referenced ISE design method can be interpreted as minimizing (5-21) with respect to the free design parameters with $\underline{x}(t)$ constrained to satisfy (5-12) for the pseudo IC (5-14). This is essentially the same form that was used in Chapter 4 in the derivation of the numerical optimization algorithm used in the general computer program of Appendix B. Equation (5-21) is the same as the quadratic function (4-2) and can be evaluated from equations (4-3) and (4-4) or (4-8). The derivation in section 4.2 of the necessary condition for a local minimum applies here up to the point where the variation δF is expressed in terms of the variation in the parameter vector, $\delta \underline{p}$ (4-25). The structure of F in this case is

$$F = \left[\begin{array}{c|c} \begin{array}{c:c} \underline{0} & I \\ \hdashline \multicolumn{2}{c}{-\underline{\alpha}'} \end{array} & O \\ \hline O & \begin{array}{c:c} \underline{0} & I \\ \hdashline \multicolumn{2}{c}{-\underline{a}'} \end{array} \end{array}\right] \tag{5-22}$$

where $\underline{\alpha}$ is the model's coefficient vector and $\underline{a}$ is the closed-loop system's coefficient vector which is a function of the free parameters, i.e. $\underline{a} = \underline{a}(p)$. Then the variation in F due to a variation in $\underline{p}$ is

$$\delta F = -\underline{\epsilon}_{n+\ell} \delta \underline{p}' A' \tag{5-23}$$

where $\underline{\epsilon}_{n+\ell}$ is an $(n + \ell) \times 1$ vector defined as

$$\underline{\epsilon}'_{n+\ell} = [0 \quad 0 \quad 0 \cdots\cdots 0 \quad 1] \tag{5-24}$$

and A is defined in this case as

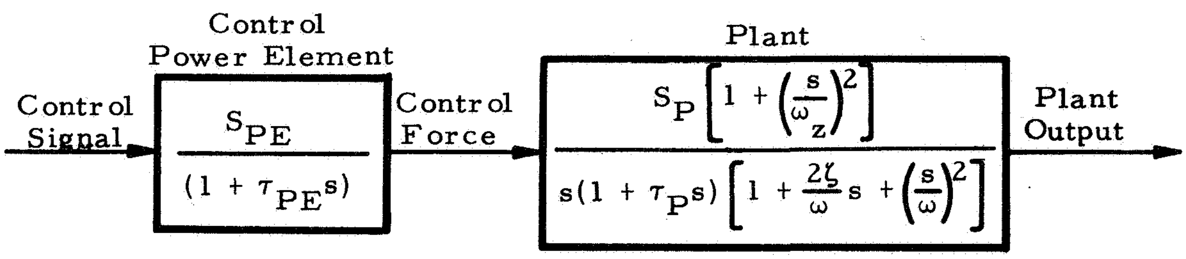
$$A = \left[O \;\middle|\; \frac{\partial \underline{a}(p)}{\partial \underline{p}} \right] \tag{5-25}$$

where O is an appropriately dimensioned null matrix. Using (5-23) in place of (4-25), the necessary condition for a local minimum is the same as (4-29) except that $\underline{\epsilon}_{n+\ell}$ must replace $\underline{\epsilon}_n$ in (4-29c). Since the direct gradient evaluation method of section 4.3 was obtained directly from the necessary condition derivation, the iterative scheme given by (4-44), with $\underline{\epsilon}_{n+\ell}$ replacing $\underline{\epsilon}_n$ and other quantities defined as above, can be used here also.

The important point of this development is that the basic form of the numerical optimization procedure is the same for both the Model PI and the model-referenced ISE and furthermore this form is independent of the specific design problem. A digital computer program can be written for the general form and check-out once. After that, it can be used for many design problems. The data required for a specific design problem are quite easily entered into the general program. The details for applying the program with the model-referenced ISE performance index are presented in section B.3 of Appendix B.

5.2.2 Comparison of a Design Example

The example considered is the design of a positional servo similar to that in section 5.1.1 but with a higher order description of the plant. A functional block diagram of the open-loop system is shown in figure 5-14. The plant contains complex zeros located on the imaginary axis and complex poles with very low damping. This oscillatory mode has a significant residue and cannot be neglected. The design



$S_{PE} = 1.0$

$\tau_{PE} = 0.5$ sec.

$S_P = 1.0$

$\omega_z = 3.0$ rad./sec.

$\tau_P = 1.0$ sec.

$\zeta = 0.01$

$\omega = 5.0$ rad./sec.

Figure 5-14 Functional Block Diagram of Open-Loop System

specifications are assumed to be the same as for the example in section 5.1.1, that is, the step response of the positional servo must lie within the envelope shown in figure 5-2.

For comparative purposes, it is desirable to use the same model in both the Model PI and the model-referenced ISE design procedures. According to table 3-5 for systems with zeros, it would be preferable to use a model with the same structure as the closed-loop system, i.e. in this case a fifth order model with two zeros. But the computational effort in the model-referenced ISE procedure increases as the order of the model increases* so that a lower order model would be advantageous to that design method (the order of the model does not affect the

---

* The justification for this statement and its importance are discussed subsequently in section 5.2.2.3.

computational effort in the Model PI method). A third order model seems to be a reasonable compromise on which to base a comparison of the two methods. With the aid of the non-dimensionalized graphs of step responses for third order systems in reference 49, the following model was easily established to represent the design specifications:

$$\frac{y_m(s)}{u(s)} = \frac{1}{(1+s)\left[1 + \frac{2(0.7)}{3.5}s + \left(\frac{s}{3.5}\right)^2\right]}$$

$$= \frac{12.22}{s^3 + (5.9)s^2 + (17.12)s + (12.22)} \qquad (5\text{-}26)$$

The step response for the model (5-26), as shown in figure 5-15, does lie within the specification envelope.

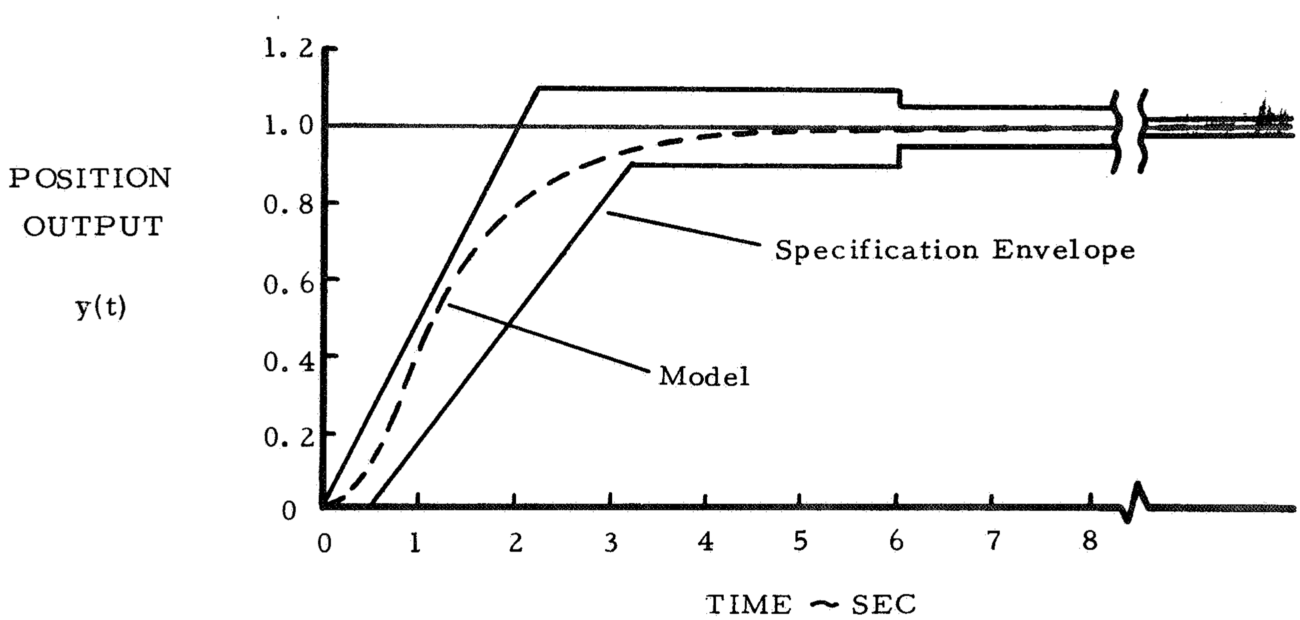
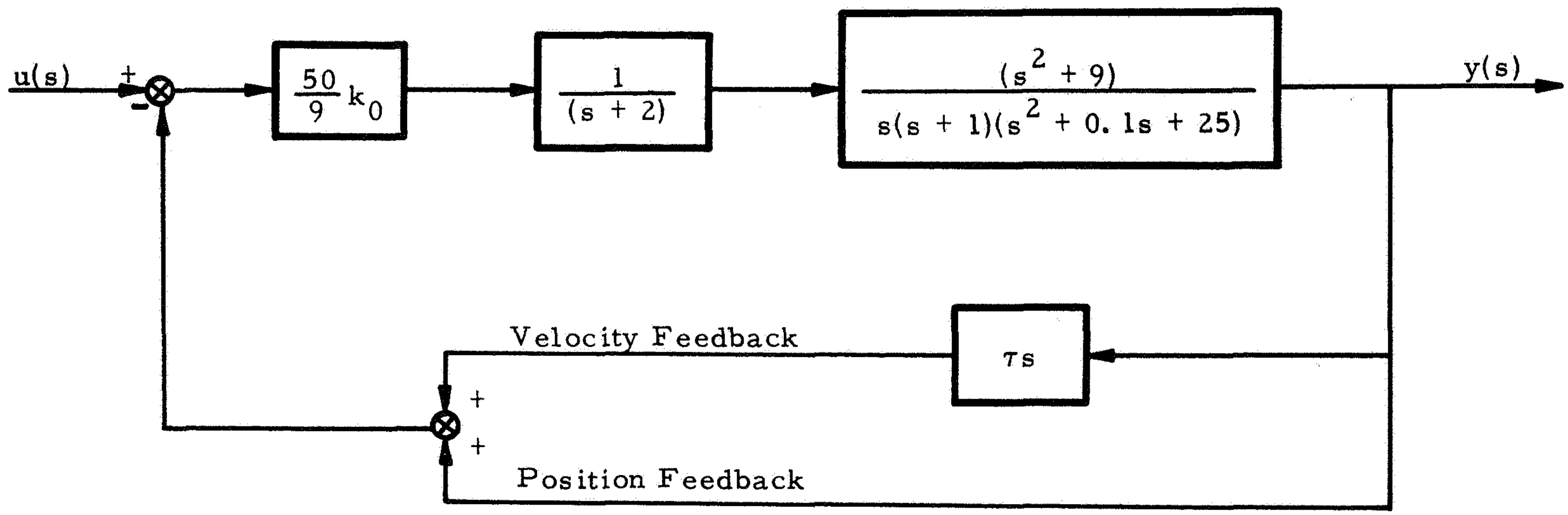


Figure 5-15 Step Response of a Third Order Model Compared to Specification Envelope



k_0 and τ are free design parameters

Figure 5-16 Block Diagram of Closed-Loop System with Position and Velocity Feedback

By comparing the open-loop system considered here, figure 5-14, to that in section 5.1. 1, figure 5-1, one can see that the previous example is an approximate design for this example based on the dominant lower order effects. Since it was necessary to use both position and velocity feedback to obtain an acceptable design in the previous example, the same feedback configuration will be assumed here, i. e., that shown in figure 5-16. The closed-loop transfer function for this feedback configuration in terms of the free parameters, k_0 and τ, is

$$\frac{y(s)}{u(s)} = \frac{k_0 \frac{50}{9} (s^2 + 9)}{D(s)} \qquad (5\text{-}27)$$

where

$$D(s) = s^5 + (3.1)s^4 + (27.3 + \frac{50}{9} k_0 \tau)s^3 + (75.2 + \frac{50}{9} k_0)s^2 + (50 + 50k_0\tau)s + (50k_0) \qquad (5\text{-}28)$$

5.2.2.1 Model PI Design

Table 3-5 indicates that for this system-model combination the simpler forms of the Model PI,

$$PI = \int_0^\infty \| \underline{\tilde{x}}(t) \|^2_{\tilde{Q}} \, dt \qquad (5\text{-}29)$$

can be used. The $\underline{\tilde{\alpha}}$ defining $\tilde{Q}$ for the model (5-26) is

$$\underline{\tilde{\alpha}}' = [\, 12.22 \quad 17.12 \quad 5.9 \quad 1 \quad 0 \quad 0 \,] \qquad (5\text{-}30)$$

The subroutine COEF for this design problem is

```
SUBROUTINE COEF(ACOF, BCOF, PAR)
DIMENSION ACOF(1), BCOF(1), PAR(1)
BCOF(1) = 9.0*PAR(1)
BCOF(2) = 0.0
BCOF(3) = PAR(1)
ACOF(1) = BCOF(1)
ACOF(2) = 9.0*PAR(2) + 50.0
ACOF(3) = PAR(1) + 75.2
ACOF(4) = PAR(2) + 27.3
ACOF(5) = 3.1
RETURN
END
```

where PAR(1) is $(50/9)k_0$ and PAR(2) is $(50/9)k_0\tau$. The input data card set-up follows that illustrated in section 5.1.1.

Starting from an initial choice of $k_0 = 1.0$ and $\tau = 1.45$, the computer solution for the Model PI design procedure was $k_0 = 1.59$ and $\tau = 1.30$, which gives a closed-loop system transfer function of

$$\frac{y(s)}{u(s)} = \frac{\left[1 + \left(\frac{s}{3}\right)^2\right]}{\left(1 + \frac{1}{0.71}s\right)\left[1 + \frac{2(0.46)}{(1.86)}s + \left(\frac{s}{1.86}\right)^2\right]\left[1 + \frac{2(0.06)}{(5.7)}s + \left(\frac{s}{5.7}\right)^2\right]} \tag{5-31}$$

The step response for this solution is compared to the design specification envelope in figure 5-17, which shows it to be an acceptable design. In this example, the specifications are assumed to be given completely by the step response envelope so that the lightly damped, higher frequency mode oscillation apparent in figure 5-17 should not be considered objectionable. If such an oscillation were unacceptable the design specification would have included an additional requirement such as a minimum damping ratio for all oscillatory modes. To increase the damping of the higher frequency mode would require additional compensation, but since that is not a requirement for this example, it will not be pursued further.

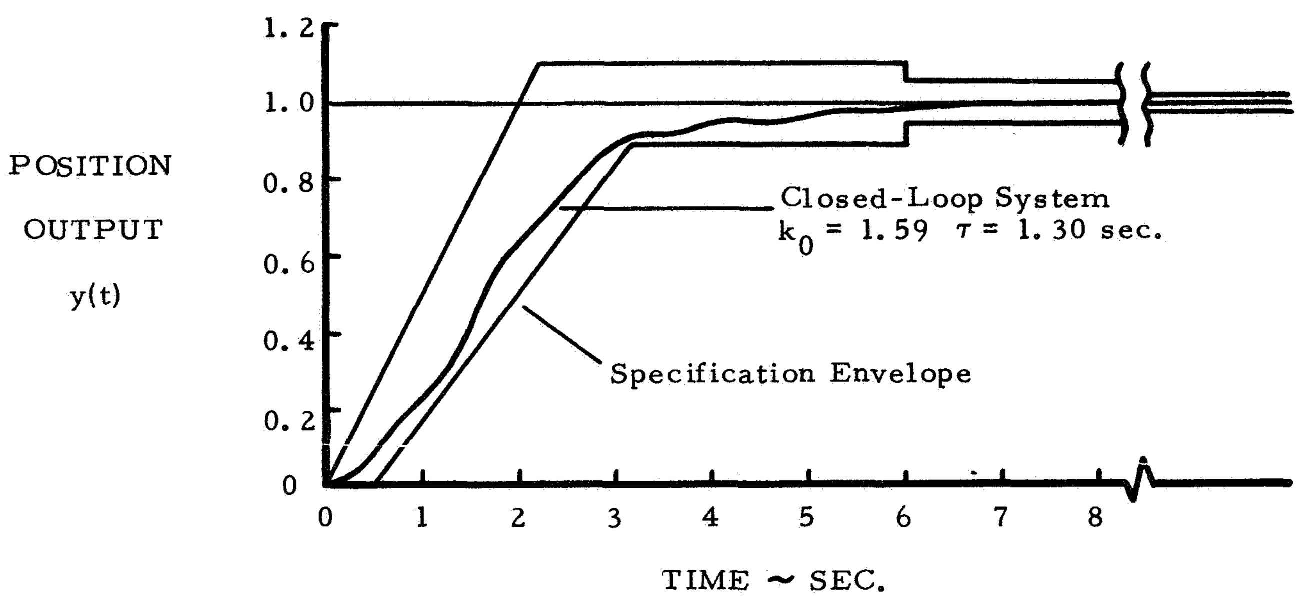


Figure 5-17 Comparison of the Model PI Design Step Response to the Specification Envelope

5.2.2.2 Model-Referenced ISE Design

When using the general computer program with the model-referenced ISE performance index it is necessary to use subroutine CALSYS MOD 3, in place of CALSYS MOD 1 (see section B. 3 of Appendix B). All of the required modifications to the optimization algorithm indicated in section 5.2.1 are performed in CALSYS MOD 3. In this procedure the model's characteristics enter the program through a subroutine COEFM used in CALSYS MOD 3. The subroutine COEFM for the model (5-26) is

```
SUBROUTINE COEFM (ALPHA, BETA, NM, MM)
DIMENSION ALPHA(1), BETA(1)
NM = 3
MM = 0
BETA(1) = 12.22
ALPHA(1) = 12.22
ALPHA(2) = 17.12
ALPHA(3) = 5.9
RETURN
END
```

where NM and MM are the number of model poles and zeros respectively, BETA(1) is the numerator coefficient of (5-26) and ALPHA(I), I = 1, 2, 3 are the denominator coefficients of (5-26). The functional relationships of the closed-loop system transfer function coefficients to the free parameters, i. e. the coefficients of (5-27) and (5-28) enter the program through the same subroutine COEF used in the Model PI design, section 5. 2. 2. 1.

The input data card set-up follows the format described in section B. 3. 1. Note that the integer N required on the first data card is the order of the system plus the order of the mode, i. e. $N = n + \ell = 5 + 3 = 8$. This is necessary because the main program is written to find the minimum of a general quadratic functional of the form given by equation (4-2) or (5-21), repeated here

$$I = \int_0^{\infty} \|\underline{x}(t)\|_Q^2 \, dt \tag{5-21}$$

in which the dimension of the state vector, x(t), is N. In the Model PI procedure N was the order of the closed-loop system, n, but in the model-referenced ISE procedure the vector, $\underline{x}(t)$, defined by (5-11), is of dimension $(n + \ell)$.

Starting from an initial choice of $k_0 = 2.2$ and $\tau = 1.0$, the computer solution for the model-referenced ISE performance index was $k_0 = 4.19$ and $\tau = 1.20$, which gives a closed-loop system transfer function of

$$\frac{y(s)}{u(s)} = \frac{\left[1 + \left(\frac{s}{3}\right)^2\right]}{\left(1 + \frac{1}{0.81}s\right)\left[1 + \frac{2(0.22)}{(2.35)}s + \left(\frac{s}{2.35}\right)^2\right]\left[1 + \frac{2(0.09)}{(6.83)}s + \left(\frac{s}{6.83}\right)^2\right]} \quad (5\text{-}32)$$

Figure 5-18, in which the step response for this solution is compared to the design specifications, shows that it is an acceptable design. The comments made in section 5. 2. 2. 1 about the lightly damped, higher frequency mode apply here also.

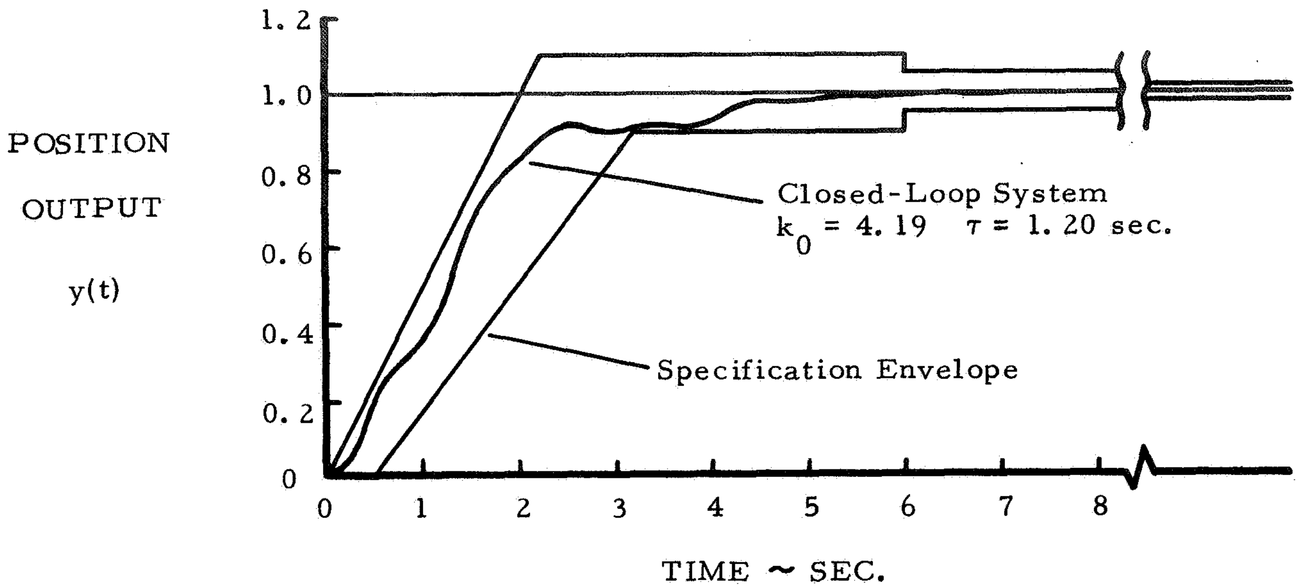


Figure 5-18 Comparison of the Model-Referenced ISE Design Step Response to the Specification Envelope

5.2.3 Comparison of Results and Discussion

Both procedures resulted in a positional servo design satisfying the specifications, so what should be the basis for comparison? From the customer's viewpoint the two designs are equally acceptable. When using a parameter optimization method in general to synthesize a control system to meet engineering specifications, the design process is not complete until the specifications are met even if several design iterations are necessary. Ultimately the basis for comparison has to be the relative efficiency of successfully applying the techniques. In this specific example, that basis reduces essentially to the relative efficiency of the digital computation task for each method. Definite conclusions can be drawn from this comparison that are indicative of the two procedures in general.

Up to the point of preparing the subroutines COEF and COEFM and the input data cards the two procedures are the same. The difference in effort to prepare these is negligible. But there is a substantial difference in the computational effort required to obtain the two designs. This effort can not be measured fairly in terms of the total elapsed computer times from the starting points to the final designs because those depend too heavily on the starting points, i.e. the initial choices for the free parameters. A fair measure of the relative computation effort is the ratio of the times required to compute one averaged gradient step. During the execution of the digital computer program in each of the above design processes the computer was instructed to print out the elapsed time for one averaged gradient step. In the Model PI approach the elapsed time was about 25 seconds and in the model-referenced ISE approach the elapsed time was about 110 seconds. Using these numbers as an indication of their relative computational times means that the model-referenced ISE method was about 1/4 as efficient as the Model PI method in terms of computational effort. This is due to the fact that the Model PI procedure does not have to compute any terms involving the model's time response whereas the model-referenced ISE procedure does.

It is possible to establish a rough estimate of the relative computational effort for a general design problem in terms of the orders of the system and model, n and ℓ respectively. By far the most time consuming portion of the optimization algorithm is the evaluation of the performance index and its gradient. This results from the numerous multiplications involved in numerically integrating two $N \times N$ matrix equations of the form (4-44c) and (4-44d), where N is the dimension of the state vector appearing in the quadratic performance index. The number of multiplications involved in multiplying two $N \times N$ matrices, which is done repeatedly in the Runge-Kutta numerical integration scheme, is N^3. For the Model PI the dimension is $N = n$ and for the model-referenced ISE the dimension is $N = n + \ell$ so that a rough estimate of the relative computation times is given by*

$$\left(\frac{\text{Model PI Compute Time}}{\text{Model-Ref. ISE Compute Time}}\right) \approx \frac{n^3}{(n + \ell)^3} \tag{5-33}$$

In the example considered here, equation (5-33) gives

$$\frac{n^3}{(n + \ell)^3} = \frac{5^3}{(5 + 3)^3} = \frac{125}{512} \approx \frac{1}{4} \tag{5-34}$$

which checks with the actual time measurements stated above.

It should be pointed out that this relative computation effort is essentially independent of the optimization algorithm used once the performance index and its gradient are evaluated. The averaged gradient direction algorithm used in the computer program described in

---

* It is possible to write a more efficient algorithm for the model-referenced ISE method than the one used here, by substituting (5-13) into equations (4-44c) and (4-44d) of Chapter 4 and using the resulting partitioned equations. In that case the similar rough estimate of the relative computation times is given by

$$\frac{n^3}{[n^3 + \ell^3 + \frac{1}{2}(n^2\ell + n\ell^2)]}$$

the trend would be the same as discussed here but not as severe.

Appendix B was not proposed as the most efficient one. More efficient techniques could reduce the number of steps required to reach a minimum point, thus reducing the total computation task for either performance index. The relative effort considered here results from the manner in which the model enters the computation of the performance index and its gradient so that the relative computation effort would still be given approximately by (5-33).

The importance of the relative computation effort may not be fully realized until considering an extensive control system design task, such as designing a flight control system. Because of the wide range of flight conditions, weight, center of gravity locations, and flight configurations (flaps, landing gear, speed brakes, external store, etc.) the total design problem involves applying the computer program numerous times to meet the specifications under all these conditions. Depending on the order of the closed-loop flight control system and model used, the savings in computer time at each test condition using the Model PI could be anywhere from 20% to 85% over that required using the model-referenced ISE method. In such a large design problem this would represent a substantial monetary savings.

Comparing the overall efficiency of successfully applying the two methods would in general involve other factors in addition to the computation times. The order and structure of the model selected, the difficulty in establishing an appropriate model, and the manner in which the design procedure tends to match the model are some of the other important factors. These are strongly dependent on the particular design problem considered. However some interesting points can be illustrated by comparing the manner in which the two designs in this example tend to match the model in both the time and frequency domain.

The position, velocity and acceleration time responses of the positional servo for a step input are compared to those of the model in figures 5-19 and 5-20 for the Model PI design and the model-referenced ISE design respectively. By comparing these figures one can see that the Model PI design attempts to obtain a balance among matching all three time responses whereas the model-referenced ISE design emphasizes matching the position time response, as would be expected. This tendency of the Model PI design is a result of the Model PI attempting to match the model's response trajectory in its extended state space. This characteristic of the Model PI can aid the designer in selecting an appropriate model. Of course the model-referenced ISE method can be used in a more general form, including the errors in velocity, acceleration and higher derivatives, to produce a design that compromises between matching several variables as with the Model PI design. In fact, if the weighting factors used in such a general form are selected to be equivalent to those defined by Q in (3-36) used in the Model PI then the resulting design would be identical to the Model PI design (this was proven in Chapter 3).

The frequency responses of the two designs are compared to that of the model in figures 5-21 and 5-22 respectively. The point illustrated in comparing these figures is that the Model PI design tends to match the model's frequency response over a wider frequency range then does the model-referenced ISE design. It also tends to attenuate the higher frequencies more than the model-referenced ISE design. These characteristics are obtained at the expense of a poorer match in the low frequency range.

In comparing these two methods, one additional point is worth mentioning. It is more concerned with the process of teaching the design methods than applying them to practical problems. Instructors have been known to assign design examples as home problems to illustrate a particular design method. Examples using the model-referenced ISE method have to be restricted to almost trivial design situations because the analytical solution and hand computation of a somewhat realistic problem involves an unreasonable amount of tedious work for a home problem. This is easily illustrated by considering the approach described in Newton, Gould and Kaiser (15) for a third order system with two free design parameters and a second order model which is about the simplest problem that is somewhat realistic. A fifth order tabulated integral, I_5, of the form shown below would be required to evaluate the performance index.

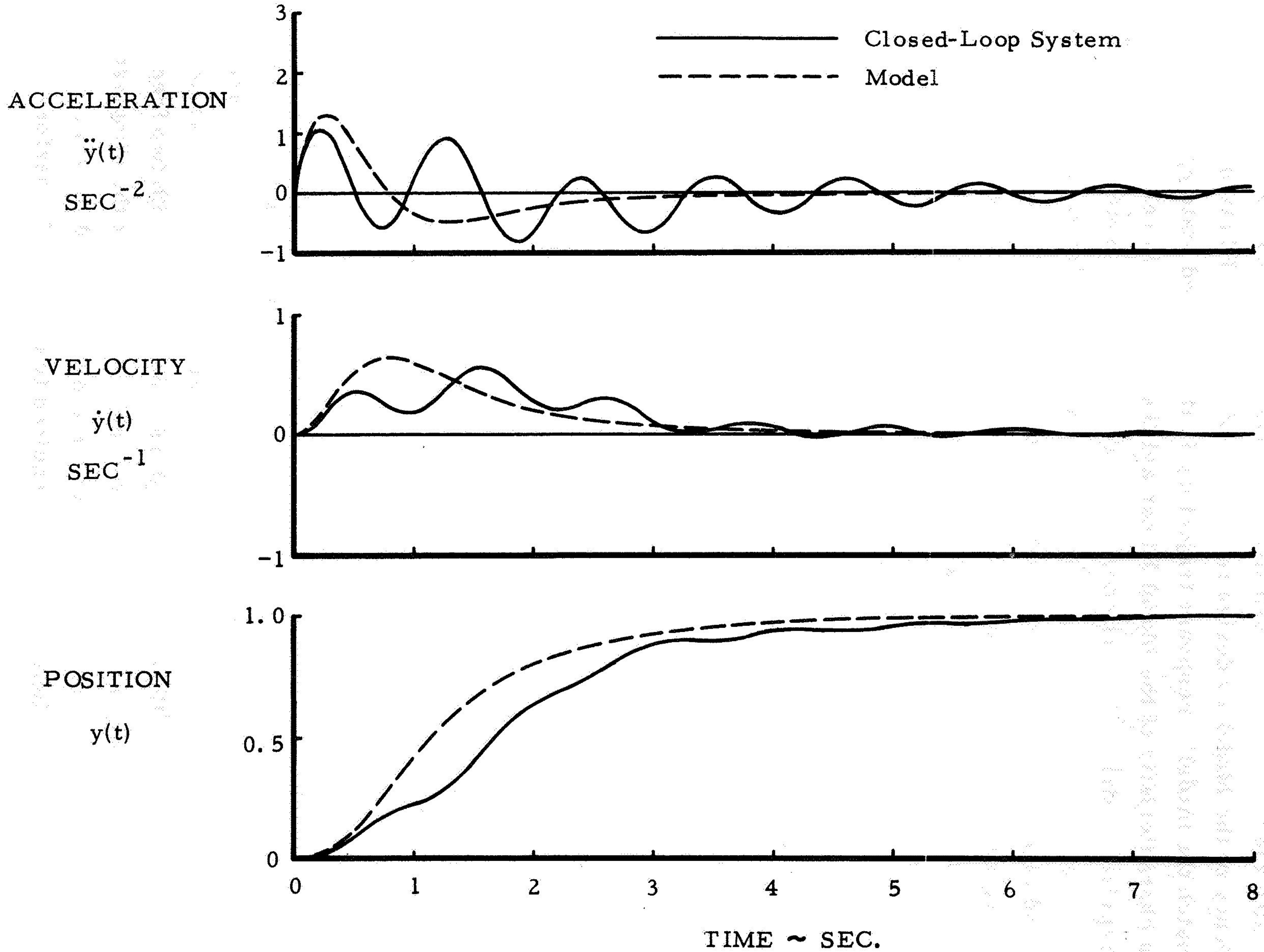


Figure 5-19 Comparison of Position, Velocity and Acceleration for the Model and Model PI Design

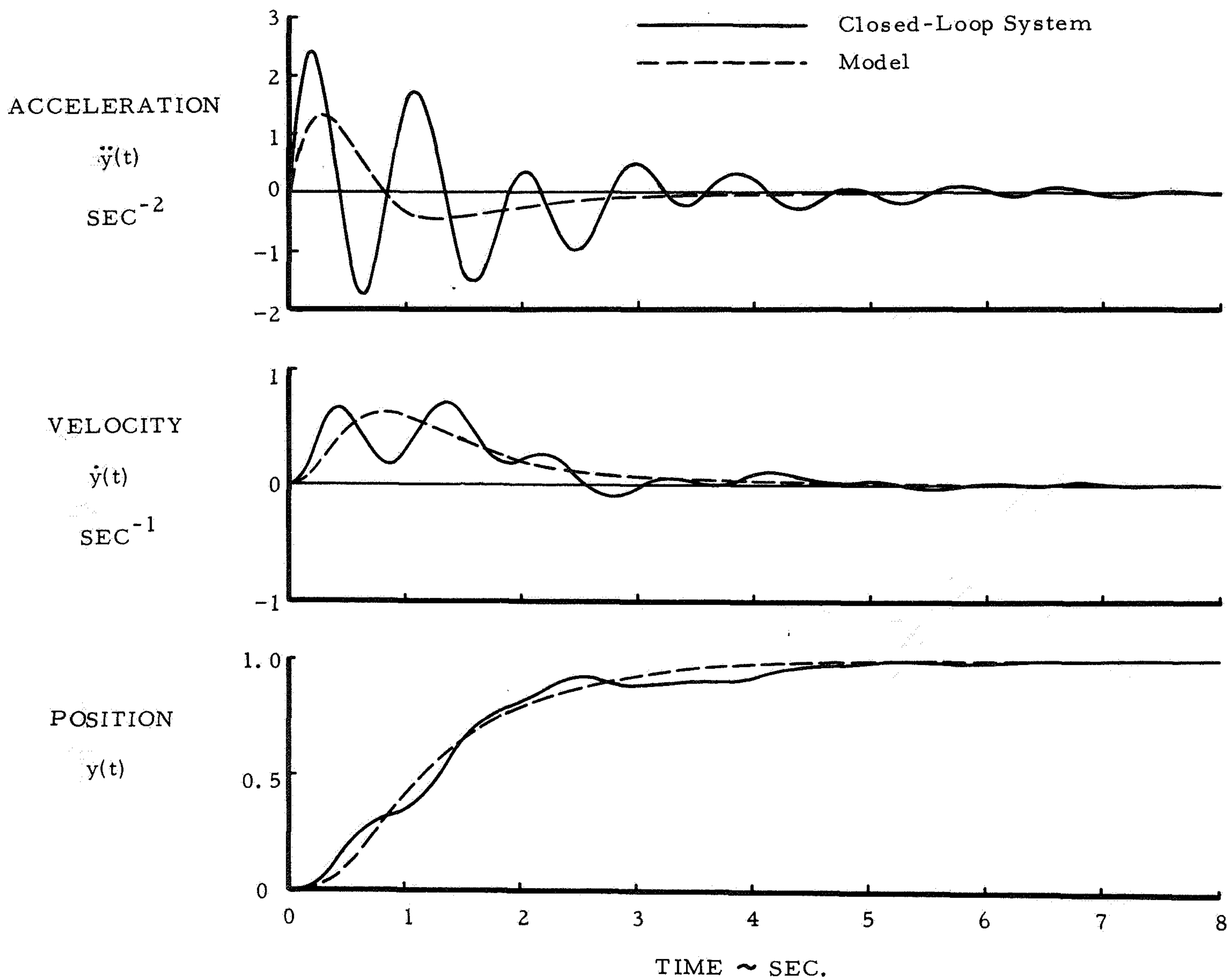


Figure 5-20 Comparison of Position, Velocity and Acceleration for the Model and Model-Referenced ISE Design

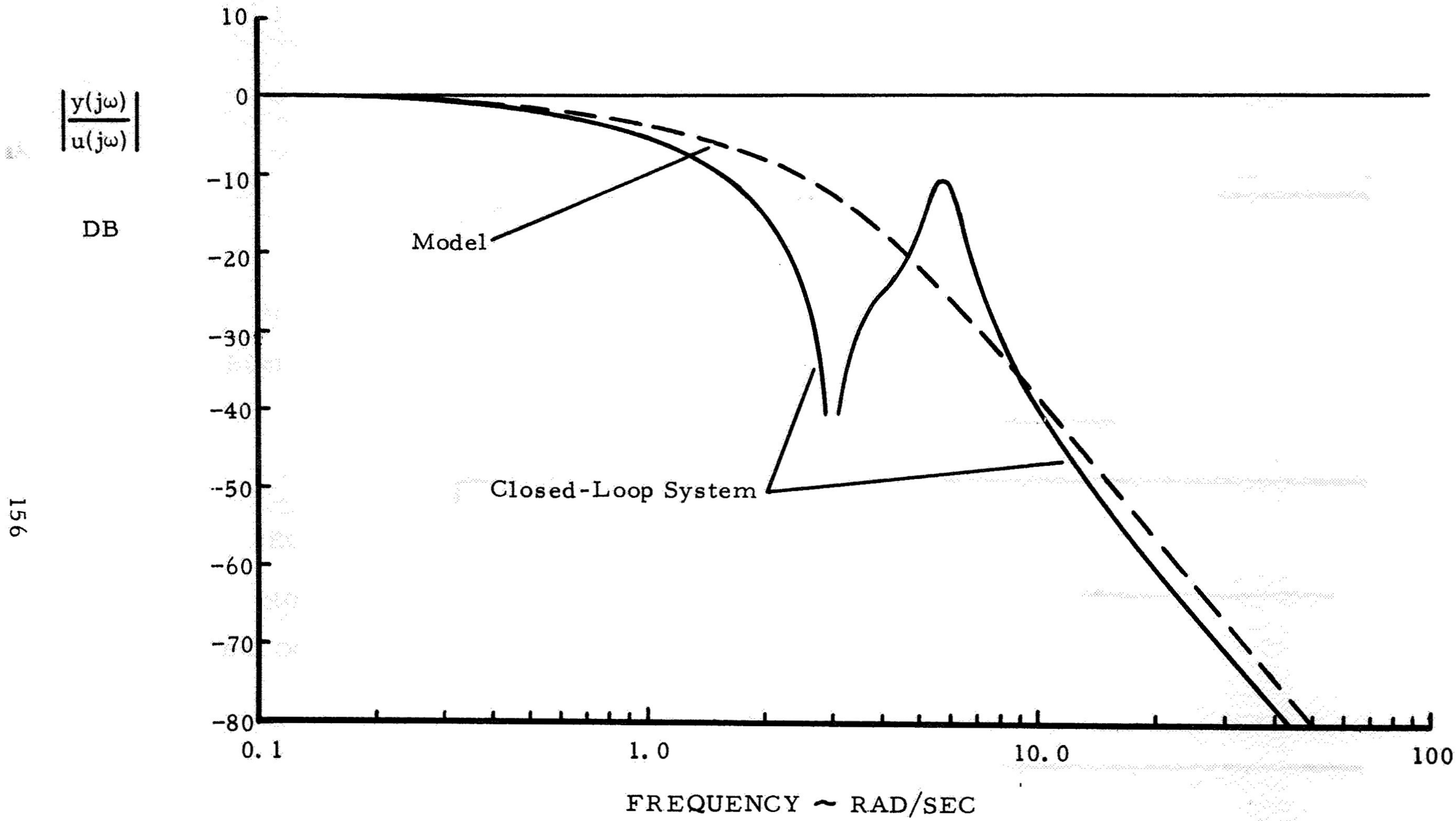


Figure 5-21 Comparison of the Model and Model PI Design Frequency Responses

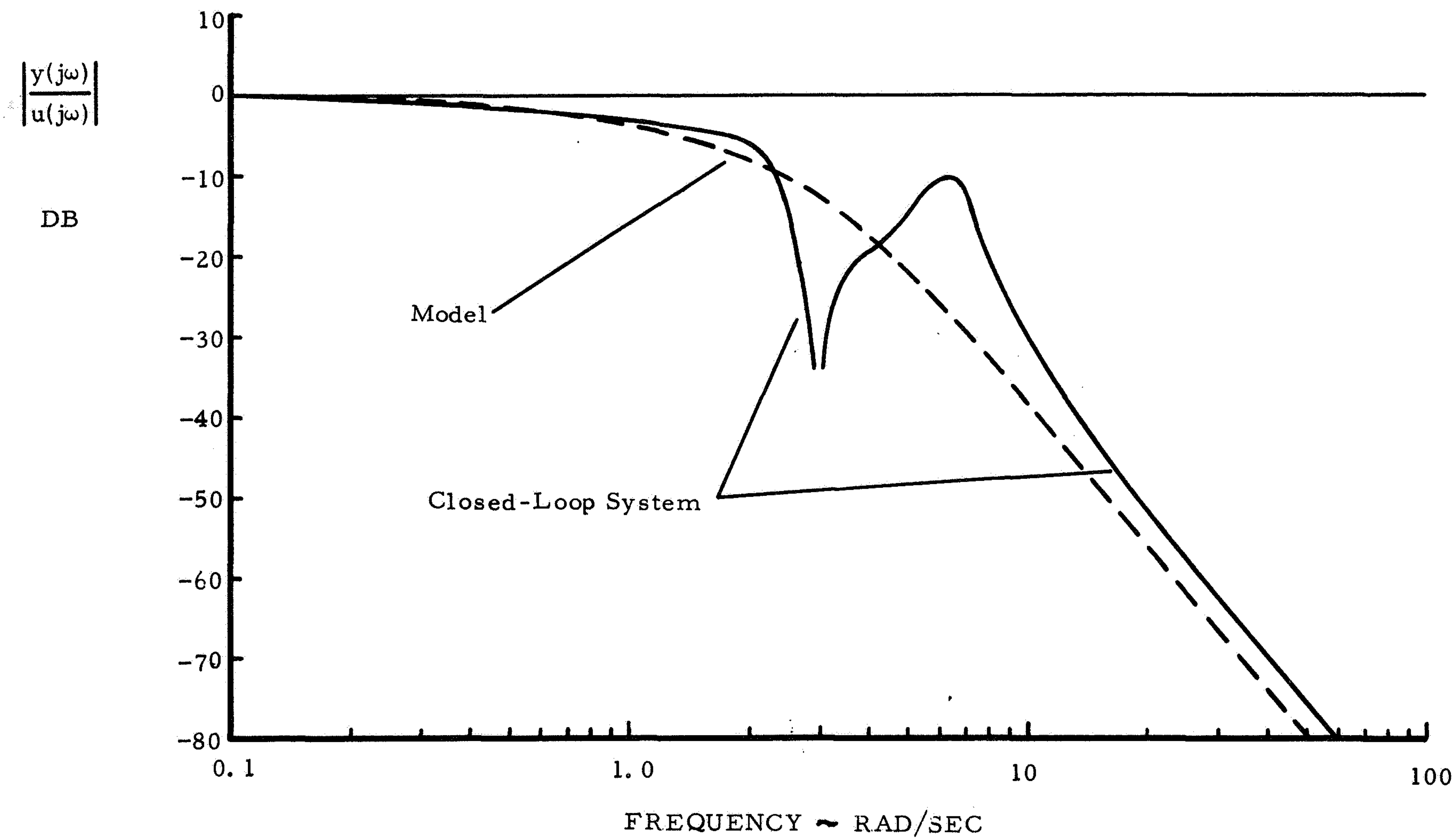


Figure 5-22 Comparison of the Model and Model-Referenced ISE Design Frequency Responses

$$I_5 = \frac{1}{2\Delta_5}\left[c_4^2 m_0 + (c_3^2 - 2c_2c_4)m_1 + (c_2^2 - 2c_1c_3 + 2c_0c_4)m_2 + (c_1^2 - 2c_0c_2)m_3 + c_0^2 m_4\right] \qquad (5\text{-}35)$$

where

$$m_0 = \frac{1}{d_5}(d_3 m_1 - d_1 m_2) \qquad m_3 = \frac{1}{d_0}(d_2 m_2 - d_4 m_1)$$

$$m_1 = -d_0 d_3 + d_1 d_2 \qquad m_4 = \frac{1}{d_0}(d_2 m_3 - d_4 m_2)$$

$$m_2 = -d_0 d_5 + d_1 d_4 \qquad \Delta_5 = d_0(d_1 m_4 - d_3 m_3 + d_5 m_2)$$

The coefficients c_i, $i = 0, 1, 2, 3, 4$ and d_j, $j = 0, \cdots, 5$ are functions of the model's numerator and denominator coefficients and the system's numerator and denominator coefficients which are, in turn, functions of the free design parameters. To minimize I_5 with respect to two free parameters is not a reasonable task to do by hand. This same example formulated with the Model PI would involve solving a 3×3 matrix algebraic equation (4-14) to form the PI (4-13) then minimizing this PI with respect to the free parameters. This is equivalent to minimizing a third order tabulated integral, i.e.

$$I_3 = \frac{c_2^2 d_0 d_1 + (c_1^2 - 2c_0c_2)\, d_0 d_3 + c_0^2 d_2 d_3}{2d_0 d_3(-d_0 d_3 + d_1 d_2)} \qquad (5\text{-}36)$$

Comparing (5-35) and (5-36) illustrates dramatically the difference in effort between the two approaches. Examples of the Model PI method with up to fourth order systems and models could reasonably be given as home problems, whereas reasonable examples of the model-referenced ISE would be restricted to no more than second order systems and models.

5.3 Some Parameter Constraint Methods

It may be necessary in some design problems to limit the range of specific free design parameters or possibly of a function of the parameters. The necessity for this may arise from various sources. If sufficient free parameters are allowed in a design problem where the model is of lower order than the system, then the optimization solution tends to make one or more parameters infinitely large, except in cases where pole-zero cancellation occurs. To be of any practical value, some limit must be set on the parameters that tend to get very large. Or the constraint on parameters may be directly related to economical hardware implementation. For example, by limiting the allowable ranges of the compensation parameters the design might be implemented with a passive network whereas exceeding these ranges might require an active compensation network, which would be more expensive. Or the open-loop gain may have to be larger than some minimum allowable value required to satisfy a steady-state disturbance specification. In general this would place a constraint on a function of the free parameters since the open-loop gain may depend on several parameters.

There are several ways that parameter constraints can be included in parameter optimization problems. The methods considered here are divided into two categories: "hard" constraint methods; and "soft" constraint methods. In "hard" constraint methods one sets up an inequality relationship that is a mathematical representation of the desired parameter range limitations. Satisfying this inequality then becomes a rigid requirement in the optimization process. In the "soft" constraint methods, one augments the original performance index with a function of the free parameters that tends to force the parameters to lie within the allowable range by penalizing large excursions outside the range. Since the methods considered do not depend on the particular performance index used, the treatment will be for a general performance index, I, which is a non-negative function of the free parameter vector, $\underline{p}$, i.e.

$$I = I(\underline{p}) \tag{5-37}$$

This can represent the Model PI, the model-referenced ISE, or some other performance index.

5.3.1 "Hard" Constraints

Assume that the desired restrictions on the design parameters are adequately represented by

$$\underline{f}(\underline{p}) \geq \underline{0} \tag{5-38}$$

for example (5-38) might correspond to the set of inequality constraints

$$\underline{f}(\underline{p}) = \begin{bmatrix} p_{1_{max}} - p_1 \\ \hline p_2 - p_{2_{min}} \\ \hline p_{2_{max}} - p_2 \\ \hline K_{OL}(p_1, p_3) - K_{OL_{min}} \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} . \tag{5-39}$$

where p_1, p_2 and p_3 are free parameters and K_{OL} is the open-loop gain.

The optimization problem is to find the value of $\underline{p}$ that minimizes $I(\underline{p})$ subject to the inequality constraint (5-38). This problem has been treated extensively in the literature (20). One procedure based on the classical method of Lagrange for constraints is the following:

> At the current value of $\underline{p}$ test the inequality (5-38). Components of $\underline{f}(p)$ satisfying the inequality are neglected. Components of $\underline{f}(p)$ violating the inequality are set equal to zero and the standard Lagrange method for equality constraints (50) is used for those components of $\underline{f}(p)$. At the next value of $\underline{p}$ the process is repeated.

Note that the test must be made on all components of $\underline{f}(p)$ at each value of $\underline{p}$ because it is possible for the optimization path to go along a constraint boundary for a while then come off it. That is, once a boundary is reached, one can not assume that the equality in (5-38) holds from then on. Other "hard" constraint procedures are reviewed in reference 20.

It is difficult to allow for arbitrary "hard" constraints in a general computer program, such as that in Appendix B, because they involve providing special logic within the optimization algorithm that depends on the specific constraints used in the design problem. "Soft" constraint methods are shown in the next section to be more convenient in that respect.

5.3.2 "Soft" Constraints

Assume that the desired restrictions on the design parameters are represented by

$$\underline{g}_{min} \leq \underline{g}(p) \leq \underline{g}_{max} \tag{5-40}$$

for example (5-40) might correspond to the sets of inequality constraints

$$\begin{bmatrix} -\infty \\ \hline p_{2_{min}} \\ \hline K_{OL_{min}} \end{bmatrix} \leq \begin{bmatrix} p_1 \\ \hline p_2 \\ \hline K_{OL}(p_1, p_3) \end{bmatrix} \leq \begin{bmatrix} p_{1_{max}} \\ \hline p_{2_{max}} \\ \hline \infty \end{bmatrix} \tag{5-41}$$

which is equivalent to (5-39). Clearly one can always transform an inequality of the form(5-38) into the form (5-40) and vice versa. The object of a "soft" constraint is to penalize values of $\underline{g}(p)$ that are outside the desired range given by (5-40). This is done by defining a scalar penalty function, $\phi[\underline{g}(\underline{p})]$, that has a relatively small value within the desired range and a relatively large value outside the desired range, then augmenting the original performance index (5-37) with $\phi[\underline{g}(p)]$, i.e.

$$I = \phi[\underline{g}(\underline{p})] + I(\underline{p}) \tag{5-42}$$

Minimizing (5-42) is similar to minimizing the original performance index as long as $\underline{g}(\underline{p})$ is within the desired range since $\phi[\underline{g}(\underline{p})]$ is small there. If $\underline{g}(\underline{p})$ is outside the range (5-40) then the first term in (5-42) tends to force $(\underline{g}(\underline{p})$ towards the desired range when (5-42) is minimized.

There are many functions that could be used as a penalty function. The most common one used is a quadratic function. In some problems a quadratic penalty may be too "soft" in which case one may want to use a quartic penalty or an exponential penalty. To illustrate the relative penalty these impose, consider a situation in which a parameter, p, which is known to be positive, is to be restricted to a value less than p_{max}. Appropriate quadratic, quartic and exponential penalty functions would be

$$\phi_2 = \left(\frac{p}{p_{max}}\right)^2 \tag{5-43}$$

$$\phi_4 = \left(\frac{p}{p_{max}}\right)^4 \tag{5-44}$$

$$\phi_e = \exp(p - p_{max}) \tag{5-45}$$

Figure 5-23 shows a comparison of these as a function of the parameter value for $p_{max} = 10$. The exponential function is clearly the best if a very sharp constraint is desired.

General forms for quadratic, quartic and exponential penalty functions representing (5-40) for finite $\underline{g}_{min}$ and $\underline{g}_{max}$, are

$$\phi_2[\underline{g}(\underline{p})] = r_{pc} \sum_{i=1}^{\mu} \left[\frac{g_i - \frac{1}{2}(g_{i_{max}} + g_{i_{min}})}{\frac{1}{2}(g_{i_{max}} - g_{i_{min}})} \right]^2 \tag{5-46}$$

$$\phi_4[\underline{g}(\underline{p})] = r_{pc} \sum_{i=1}^{\mu} \left[\frac{g_i - \frac{1}{2}(g_{i_{max}} + g_{i_{min}})}{\frac{1}{2}(g_{i_{max}} - g_{i_{min}})} \right]^4 \qquad (5\text{-}47)$$

$$\phi_e[\underline{g}(\underline{p})] = r_{pc} \sum_{i=1}^{\mu} [\exp\{c_i(g_i - g_{i_{max}})\} + \exp\{c_i(g_{i_{min}} - g_i)\}] \qquad (5\text{-}48)$$

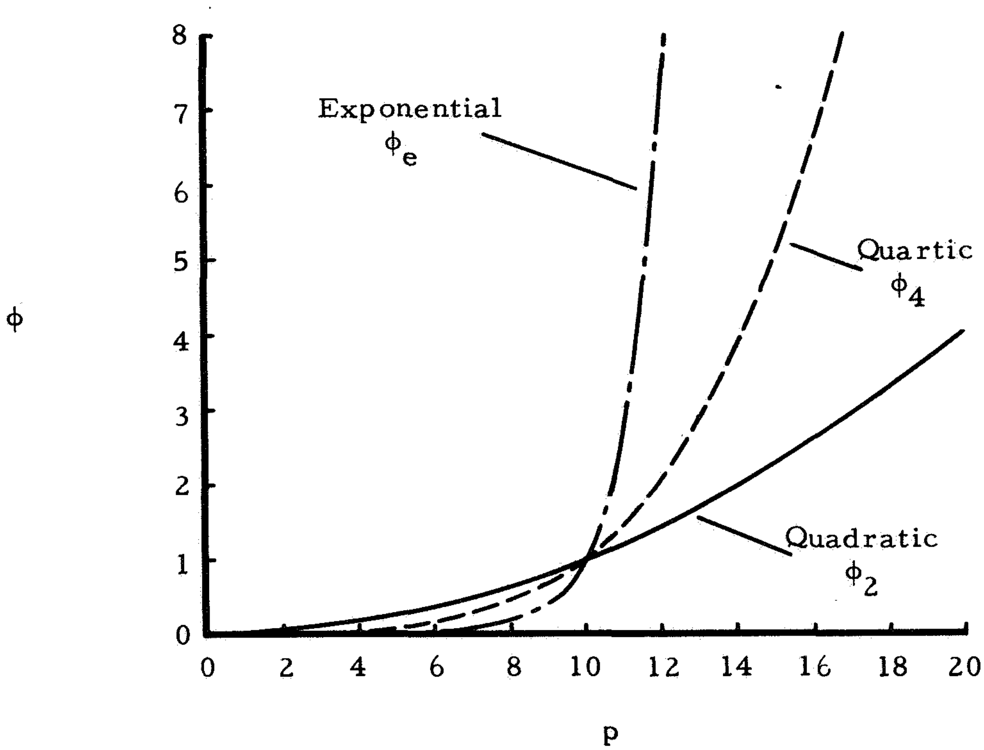


Figure 5-23 Comparison of Quadratic, Quartic and Exponential One-Sided "Soft" Constraints

where μ is the number of elements in $\underline{g}(\underline{p})$, the c_i's are constants that establish how sharp the exponential penalty is, and r_{pc} is a weighting factor for the parameter constraint contribution to the performance index. The weighting factor r_{pc} must be selected so that ϕ is actually small compared to $I(\underline{p})$ when $\underline{g}(p)$ is within the desired range (5-40). Figure 5-24 illustrates the relative penalty imposed using these "soft"

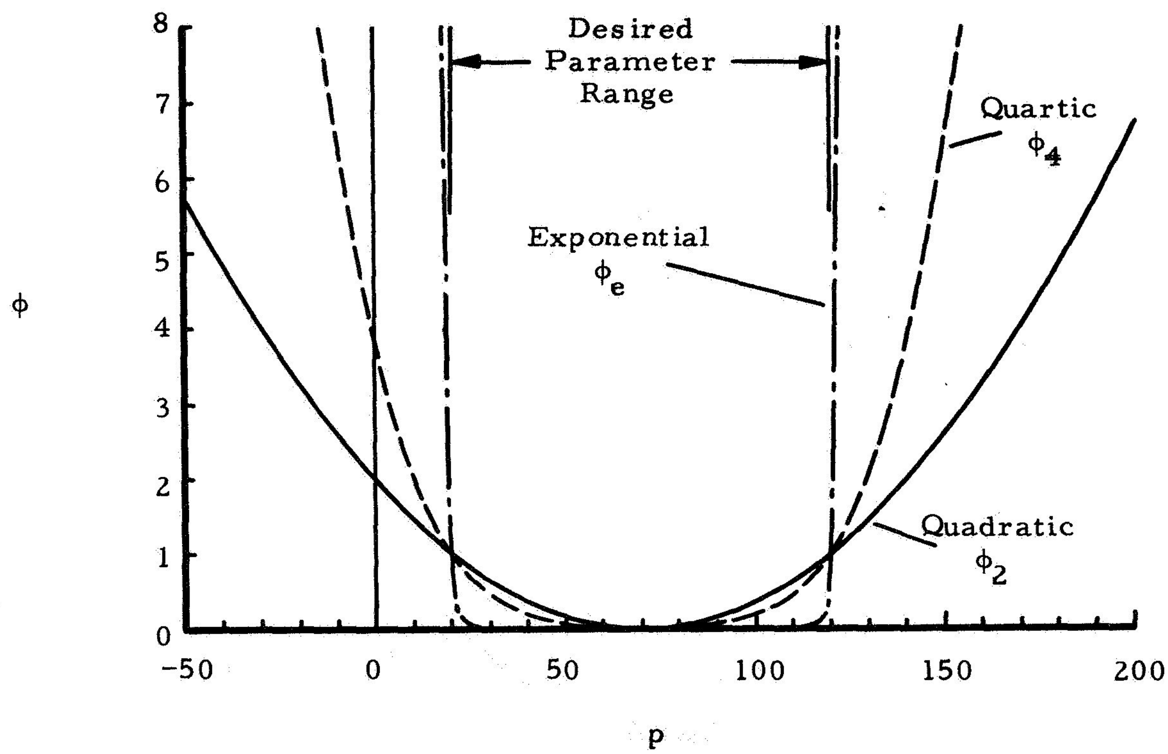


Figure 5-24 Comparison of Quadratic, Quartic and Exponential Two-Sided "Soft" Constraints

constraints for an example where $\underline{g}$ is a scalar parameter, p, and the desired range is $20 \leq p \leq 120$. Using the exponential form can actually provide as sharp of a constraint boundary as desired by choosing a sufficiently large value for c in (5-48).

Arbitrary "soft" constraints can be allowed for in a general computer program for control system design quite easily. Since the augmented performance index (5-42) is linear with respect to ϕ and $I(\underline{p})$ its gradient is also linear with respect to $\nabla\phi$ and $\nabla I(\underline{p})$. The terms ϕ and $\nabla\phi$ can easily be computed in a subroutine and added to $I(\underline{p})$ and $\nabla I(\underline{p})$ in the main program. Then to use a "soft" constraint in a particular design program only involves writing a subroutine for computing ϕ and $\nabla\phi$. If ϕ is one of the forms (5-46) - (5-48), its value and $\nabla\phi$ are analytical functions that are easily programmed and computed. The gradients corresponding to (5-46) - (5-48) are given by

$$\frac{\partial\phi_2[\underline{g}(\underline{p})]}{\partial p_j} = 8r_{pc}\sum_{i=1}^{\mu}\left[\frac{g_i - \frac{1}{2}(g_{i_{max}} + g_{i_{min}})}{(g_{i_{max}} - g_{i_{min}})^2}\right]\frac{\partial g_i}{\partial p_j} \qquad (5\text{-}49)$$

$$\frac{\partial\phi_4[\underline{g}(\underline{p})]}{\partial p_j} = 48r_{pc}\sum_{i=1}^{\mu}\frac{\left[g_i - \frac{1}{2}(g_{i_{max}} + g_{i_{min}})\right]^3}{(g_{i_{max}} - g_{i_{min}})^4}\frac{\partial g_i}{\partial p_j} \qquad (5\text{-}50)$$

$$\frac{\partial\phi_e[\underline{g}(\underline{p})]}{\partial p_j} = r_{pc}\sum_{i=1}^{\mu} c_i\left[\exp\{c_i(g_i - g_{i_{max}})\}\right.$$

$$\left. - \exp\{c_i(g_{i_{min}} - g_i)\}\right]\frac{\partial g_i}{\partial p_j} \qquad (5\text{-}51)$$

where p_j and $\partial\phi/\partial p_j$ are the jth components of $\underline{p}$ and $\nabla\phi$ respectively.

This is the general approach recommended for including parameter constraints in the program listed in Appendix B. The subroutine DELPC is provided for in the main program for this purpose and the weighting factor r_{pc} is named RPC. To include "soft" parameter constraints in a particular design problem RPC must be given some nonzero value and the user must write an appropriate subroutine DELPC (see section B. 1 of Appendix B).

CHAPTER 6

FLIGHT CONTROL SYSTEM APPLICATION

The Model PI is applied in this chapter to the engineering design of three flight control system examples. Chapter 2 is an important prelude to these examples since the emphasis here is on the relationship of realistic design requirements to the synthesis process. The type of design problems illustrated involve the pilot maneuvering loop, as referred to in Chapter 2, and hence involve aircraft handling qualities requirements. First a simplified pitch damper system is designed to provide Satisfactory to Good longitudinal handling qualities for the X-15 aircraft. The second example is the design of a complex lateral-directional stability augmentation system for the X-15 aircraft, which illustrates the Model PI design methods for multivariable systems. Finally, a pitch axis control system is designed for a VTOL aircraft in which the pilot's control stick commands the aircraft's velocity with respect to the ground. The design specifications of this latter example includes handling qualities requirements and velocity step response requirements.

6.1 Simplified Pitch Damper Design for the X-15 Aircraft

An example of a simplified pitch damper for the X-15 aircraft that was designed by linear optimal control in references 27 and 28, is redesigned here using the Model PI. The objective is to produce Satisfactory longitudinal handling qualities at one flight condition. The present problem formulation differs from that in references 27 and 28 in respect to this objective. These previous works were primarily interested in illustrating the form of the linear optimal control solution and did not state any design objective. Here, the emphasis is on meeting realistic design criteria for such a flight control system. Figure 2-2, which was developed basically for this type of vehicle, is used as the handling qualities criteria. Other factors affecting the longitudinal handling qualities, mentioned in section 2.1.1.1, are assumed to be satisfactory.

Only the rigid body, short period longitudinal dynamics at one flight condition are considered. Small perturbations about the equilibrium flight path are described approximately by

$$\begin{aligned} \dot{q} &= M_q q + M_\alpha \alpha + M_{\dot{\alpha}} \dot{\alpha} + M_{\delta_h} \delta_h \\ \dot{\alpha} &= q - L_\alpha \alpha - L_{\delta_h} \delta_h \\ n_z &= \frac{V}{g} (q - \dot{\alpha}) \end{aligned} \tag{6-1}$$

where

- q is the pitch rate
- α is the incremental angle of attack
- n_z is the normal acceleration
- δ_h is the incremental deflection of the horizontal stabilizer

M_q, M_α, $M_{\dot{\alpha}}$, M_{δ_h}, L_α and L_{δ_h} are constant, dimensional stability and control derivatives.

The X-15 pitch axis control is obtained from all movable horizontal stabilizers (51). The combined dynamics of the pitch damper servo and horizontal stabilizer hydraulic actuator is represented in this example by a single first order lag, i.e.

$$\dot{\delta}_h = -\frac{1}{\tau} \delta_h + \frac{1}{\tau} \delta_{h_c} \tag{6-2}$$

where

δ_{h_c} is the incremental command input to the pitch damper servo

τ is the equivalent time constant

Using the value of $\tau = 0.15$ as assumed in reference 28, gives a transfer function

$$\frac{\delta_h(s)}{\delta_{h_c}(s)} = \frac{1}{(1 + 0.15s)} \tag{6-3}$$

The numerical values used in reference 28 for the X-15 at a Mach number of 4.8 and an altitude of 77,000 feet (configuration and weight not specified) are listed in table 6-1.

TABLE 6-1

DIMENSIONAL DERIVATIVES FOR THE X-15 AIRCRAFT
(From reference 28)

Mach Number = 4.8	Altitude = 77,000 feet
$M_q = -0.132 \ sec^{-1}$ $M_\alpha = -17.1 \ sec^{-2}$ $M_{\dot{\alpha}} = -0.046 \ sec^{-1}$ $M_{\delta_h} = -12.2 \ sec^{-2}$	$L_\alpha = 0.277 \ sec^{-1}$ $L_{\delta_h} = 0.037 \ sec^{-1}$

Using these values results in an open loop X-15 pitch rate transfer function of

$$\frac{q(s)}{\delta_h(s)} = \frac{-0.161\left(1 + \frac{s}{0.226}\right)}{1 + \frac{2(0.055)}{(4.13)}s + \left(\frac{s}{4.13}\right)^2} \tag{6-4}$$

which corresponds to a short period mode frequency, ω_{sp}, of 4.13 rad./sec. and a damping ration, ζ_{sp}, of 0.055. These values are interpreted on figure 6-1 in terms of the longitudinal handling qualities criteria as being Unacceptable. This would clearly indicate the need

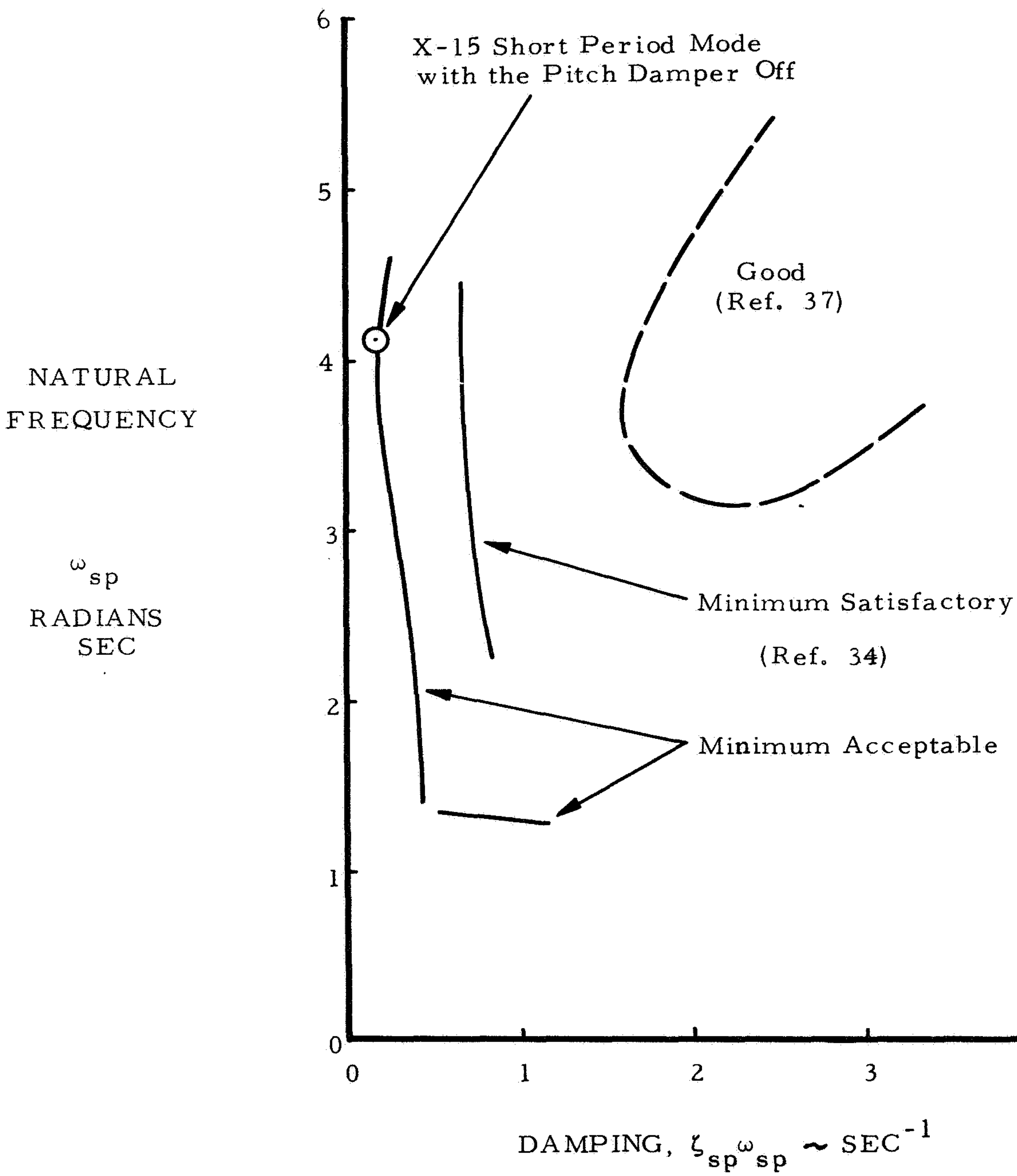


Figure 6-1 Longitudinal Handling Qualities of the X-15 Aircraft (Pitch Damper Off) at Mach No. 4.8 and Altitude of 77,000 Feet)

for a pitch damper system.

The design requirements for a pitch damper in this case are only requirements on the closed-loop short period mode poles. Nothing is stated explicitly about the pole due to the actuator or the system's zeros, both of which affect the closed-loop response. The general requirements on these are implicit in the handling qualities criteria. Criteria such as on figure 6-1 are established based on tests in which the short period mode is dominant. Any other modes such as that due to the actuator must therefore have small residues compared to that of the short period mode in order for the handling qualities to remain valid. How small the actuator mode residue must be, is left to the designer's engineering judgement. Chalk (39) has shown in a study on the effect of L_α and true speed that the zeros and static sensitivities of the short period mode transfer functions affect the handling qualities. One conclusion was the L_α and true speed must be within the best range of values as determined in the study, for criteria such as that on figure 6-1 to be valid. This example meets that requirements so that figure 6-1 is appropriate.

The first step in the Model PI design procedure is to select a model to represent the design specifications. For the type of specifications in this problem it is convenient to work in the s-plane. The criteria of figure 6-1 are transformed into the s-plane on figure 6-2. Also the open-loop pitch response and actuator poles and zero are indicated for reference. The cross-hatched region represents a realistic design objective for the closed-loop short period mode poles. Requiring "Good" handling qualities may place too severe of a requirement on the damper system, but the designer would still try to get as close to the Good region as feasible. Therefore the cross-hatched region is as the design objective with the highest damping preferred.

Two design cases are computed using two different choices for the short period mode model poles. These are shown on figure 6-2. Model 1 possibly represents the most desirable choice, but the damper gain may be too high. Model 2 would correspond to a lower damper

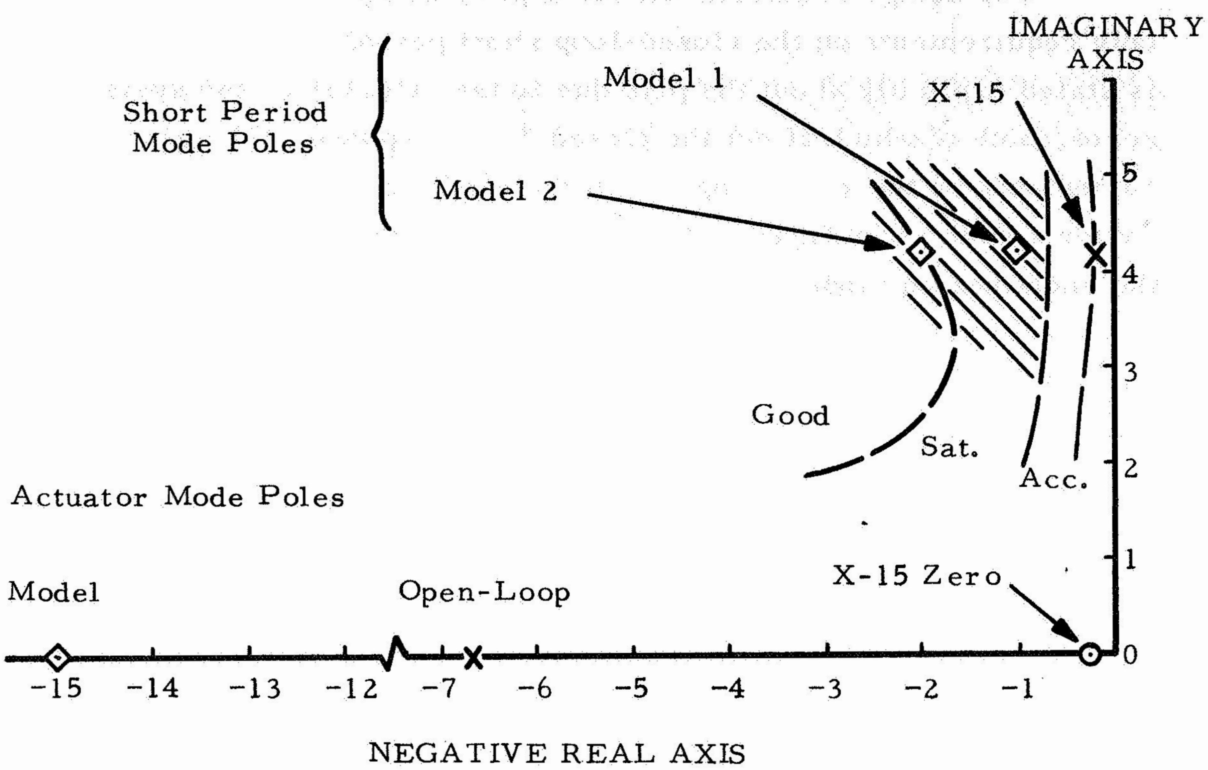


Figure 6-2 Relationship of Poles for Models 1 and 2 to Longitudinal Handling Qualities Criteria

gain and still be a satisfactory solution. In each case the model includes a pole for the actuator mode that is sufficiently far to the left to have little effect on the handling qualities. The poles for the two models are

Model 1

$$s_{sp} = -1.0 \pm j4.2 \qquad s_a = -15.0 \tag{6-5}$$

Model 2

$$s_{sp} = -2.0 \pm j4.2 \qquad s_a = -15.0 \tag{6-6}$$

where s_{sp} and s_a refer to the short period mode and actuator mode poles respectively.

The damper system configuration considered used pitch rate feedback and a feedback around the servo actuator, as shown in figure 6-3. The two feedback gains k_q and k_δ are the free design parameters. These are to be selected so that the closed-loop short period mode poles are close to those of the model and the actuator mode pole is relatively near that of the model. The closed-loop transfer function is

$$\frac{q(s)}{\delta_{h_p}(s)} = \frac{-81.4\,(s + .226)}{s^3 + a_2 s^2 + a_1 s + a_0} \tag{6-7}$$

where

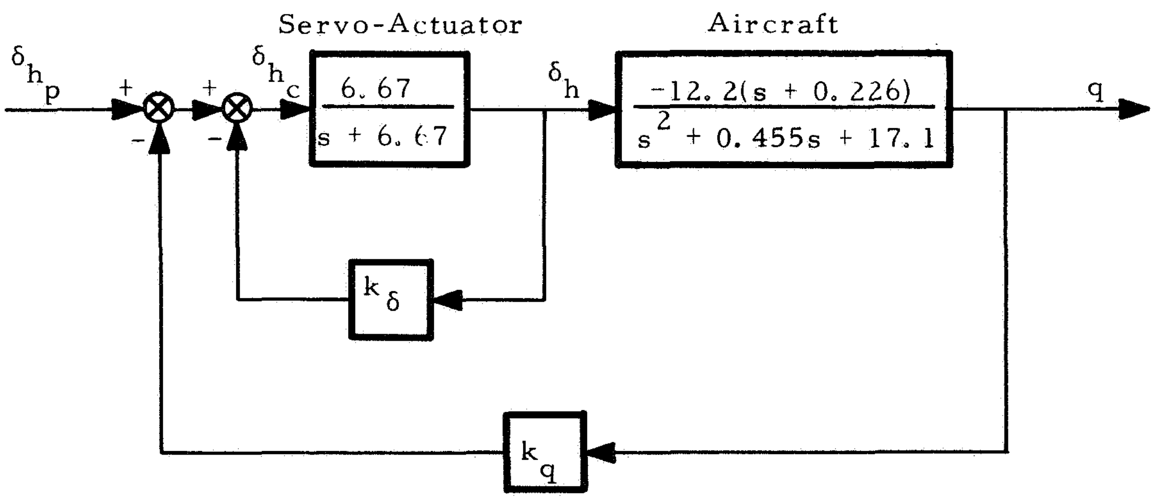


Figure 6-3 Block Diagram of Pitch Damper Configuration

$$a_0 = 114 + 114k_\delta - 18.4k_q$$
$$a_1 = 20.13 + 3.03k_\delta - 81.4k_q \qquad (6\text{-}8)$$
$$a_2 = 7.13 + 6.67k_\delta$$

This design problem is an example of the special situation discussed in section 3.2.2 in which the specifications are only on the closed-loop poles and the zero is unaffected by the free parameters. In such a case, the special form of the Model PI,

$$PI = \int_0^\infty \| \underline{x}(t) \|^2_{\tilde{Q}} \, dt \qquad (6\text{-}9)$$

can be used. The model coefficient vectors for the two models are obtained by forming their characteristic equations from (6-5) and (6-6) respectively.

Model 1

$$s^3 + 17.0 s^2 + 48.64 s + 279.6$$
$$\underline{\alpha}' = [279.6 \quad 48.64 \quad 17.0 \quad 1] \qquad (6\text{-}10)$$

Model 2

$$s^3 + 19.0 s^2 + 81.64 s + 324.6$$
$$\underline{\alpha}' = [324.6 \quad 81.64 \quad 19.0 \quad 1] \qquad (6\text{-}11)$$

The problem is now set up in the form discussed in the previous chapter to apply the general computer program for the Mode PI (Appendix B). One need only to write the subroutine COEF to generate the coefficients (6-8) and punch the data cards for (6-10) and (6-11) for the appropriate model.

Solution for Model 1

An initial choice for the free parameters of

k_δ = 0.9 deg/deg

k_q = −0.5 deg/deg per sec

resulted in final values of

$$k_\delta = 1.315 \text{ deg/deg}$$
$$k_q = -0.288 \text{ deg/deg per sec}$$

The corresponding closed-loop transfer function is

$$\frac{q(s)}{\delta_{h_p}(s)} = \frac{-0.0683\left(1 + \frac{s}{0.226}\right)}{\left(1 + \frac{s}{13.9}\right)\left[1 + \frac{2(0.23)}{(4.4)}s + \left(\frac{s}{4.4}\right)^2\right]} \qquad (6\text{-}12)$$

This solution is compared to the model and the design specifications on figure 6-4. The closed-loop short period mode poles are seen

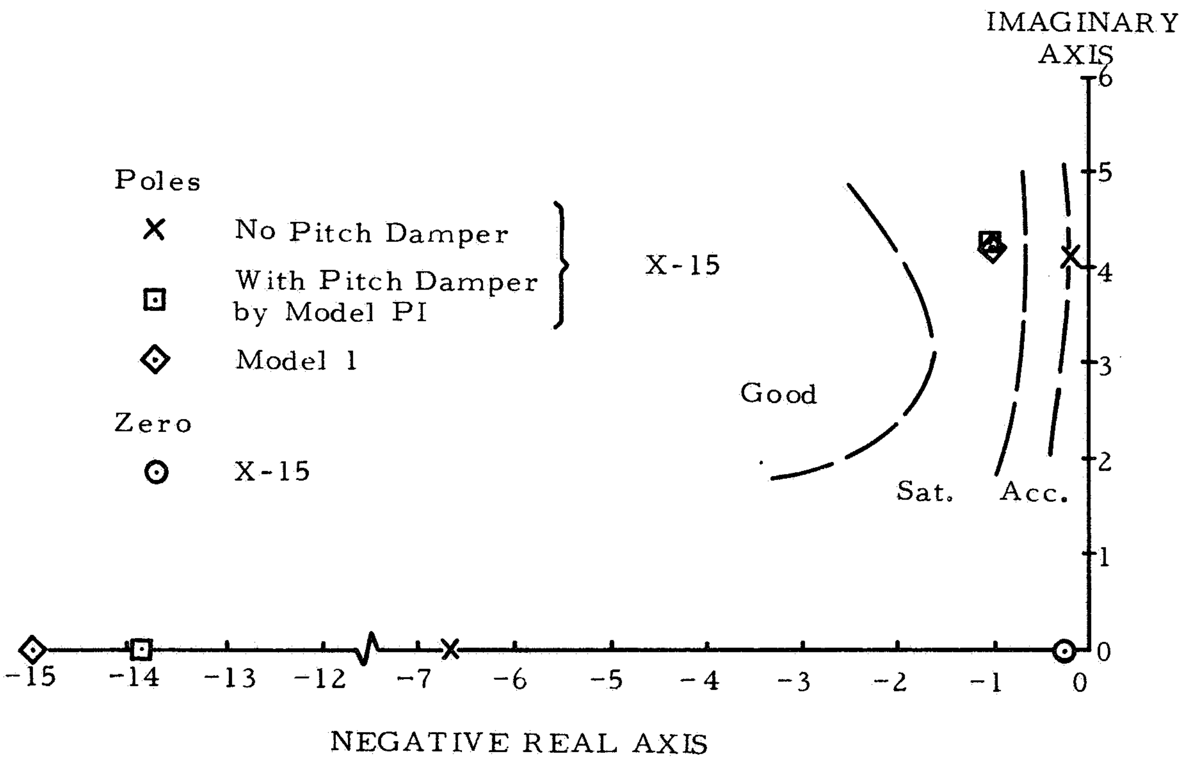


Figure 6-4 Comparison of Model PI Solution Poles for Model 1 to Longitudinal Handling Qualities Criteria

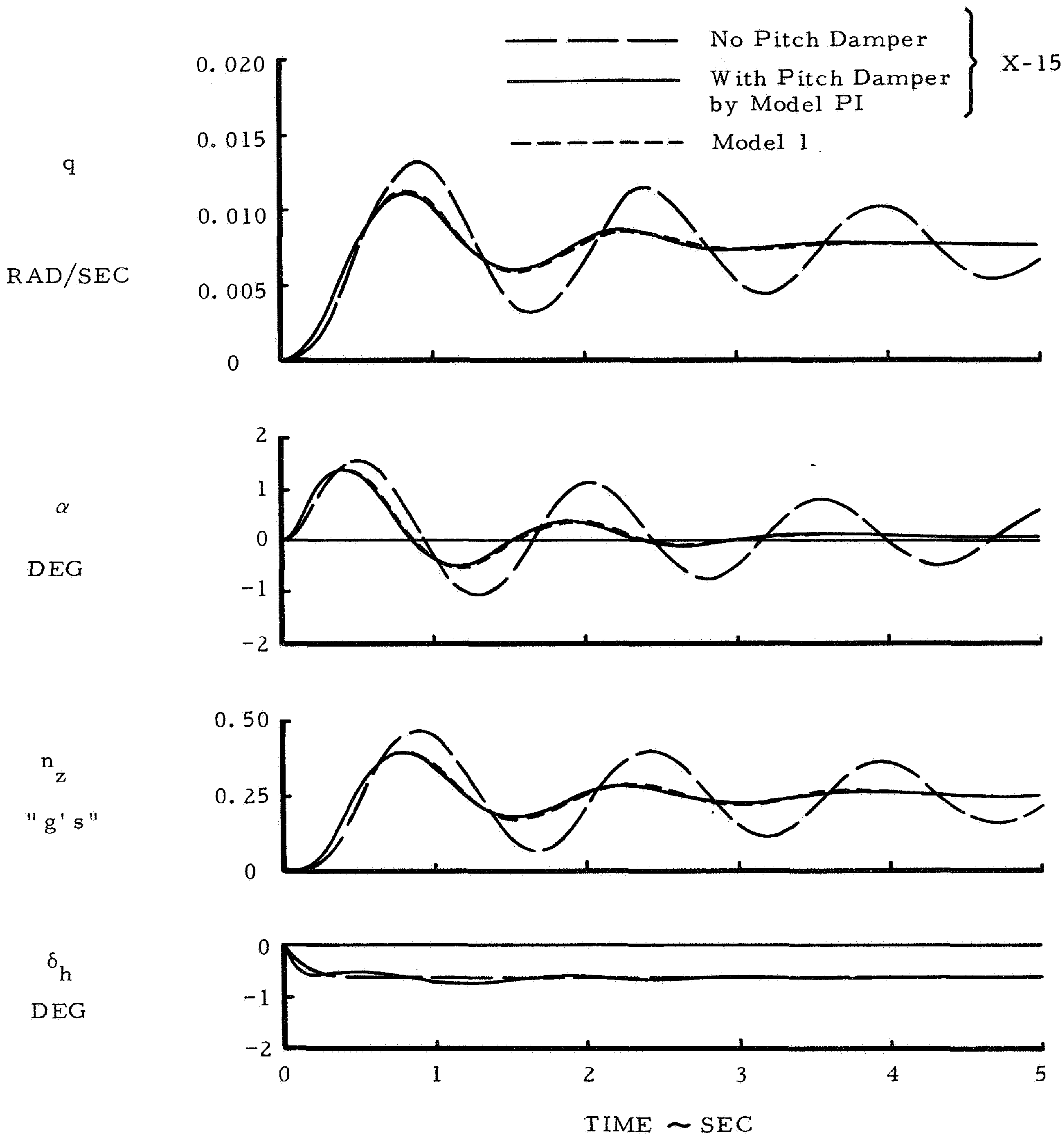


Figure 6-5 Time History Comparison of the Model PI Pitch Damper Design to Model 1 for a $\frac{1}{4}$ "g" Pull-Up

to be very close to those of the model and, more important, in the region corresponding to Satisfactory handling qualities. It's interesting to note that the pitch rate feedback gain for this solution is approximately the same as a gain setting of 4 in the actual X-15 aircraft. The X-15 pitch damper gain setting range is from 1 to 10 corresponding to a gain range of 0.075 to 0.750 deg per deg/sec (51). The gain, k_q, selected using the Model PI is certainly of a realistic magnitude.

Figure 6-5 shows the time response of q, α, n_z and δ_h for a $\frac{1}{4}$ "g" pull-up maneuver for the dampers off X-15, model 1 and the dampers on X-15 designed by the Model PI method. The dampers on case is essentially the same as the model. The model's zeros are taken to be the same as those for the X-15.

Solution for Model 2

An initial choice for the free parameters of

$$k_\delta = 1.0 \ \text{deg/deg}$$
$$k_q = -0.6 \ \text{deg/deg per sec}$$

resulted in final values of

$$k_\delta = 1.620 \ \text{deg/deg}$$
$$k_q = -0.661 \ \text{deg/deg per sec}$$

The corresponding closed-loop transfer function is

$$\frac{q(s)}{\delta_{h_p}(s)} = \frac{-0.0588\left(1 + \frac{s}{0.226}\right)}{\left(1 + \frac{s}{13.9}\right)\left[1 + \frac{2(0.43)}{(4.74)} s + \left(\frac{s}{4.74}\right)^2\right]} \qquad (6\text{-}13)$$

This solution is compared to the second model and the design specifications on figure 6-6. Again the Model PI solution is very close to the model and near the Good handling qualities region. The pitch rate feedback gain in this case corresponds approximately to an actual X-15 pitch damper setting of 9, so that the gains are not unrealistic in magnitude. The time responses for this solution presented in figure 6-7 show results similar to the first design, only more heavily damped.

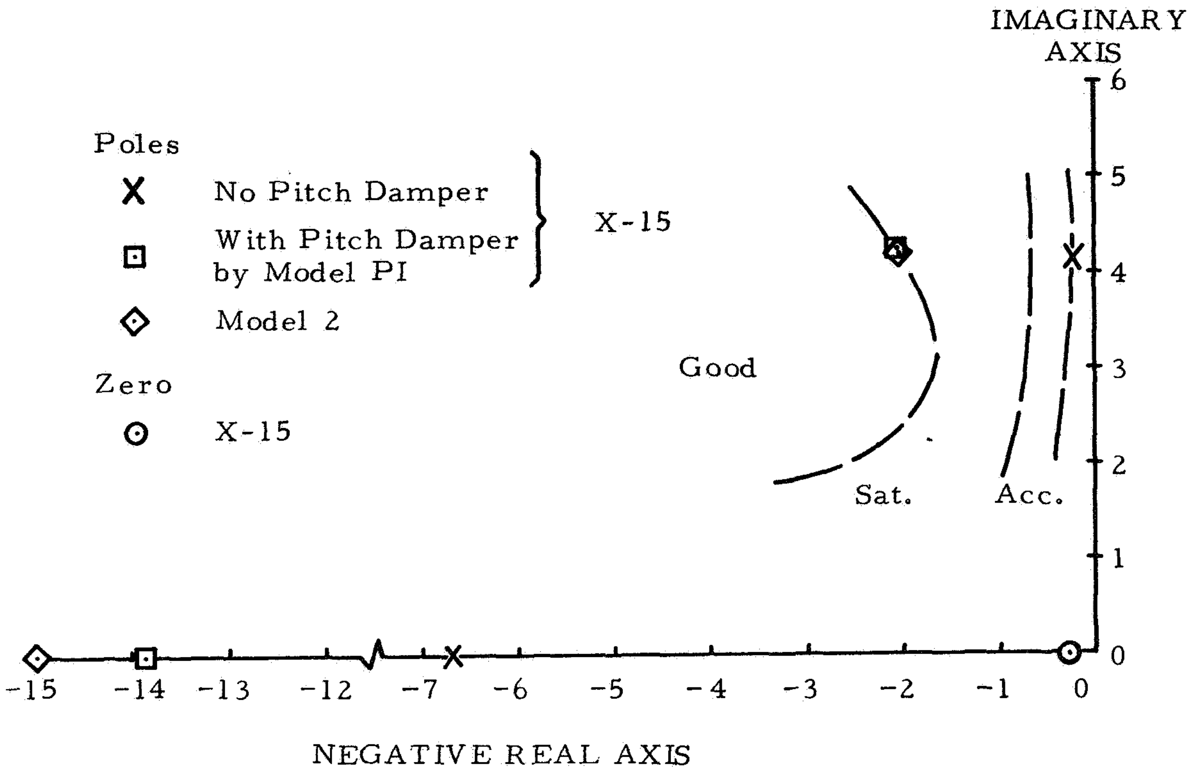


Figure 6-6 Comparison of Model PI Solution Poles for Model 2 to Longitudinal Handling Qualities Criteria

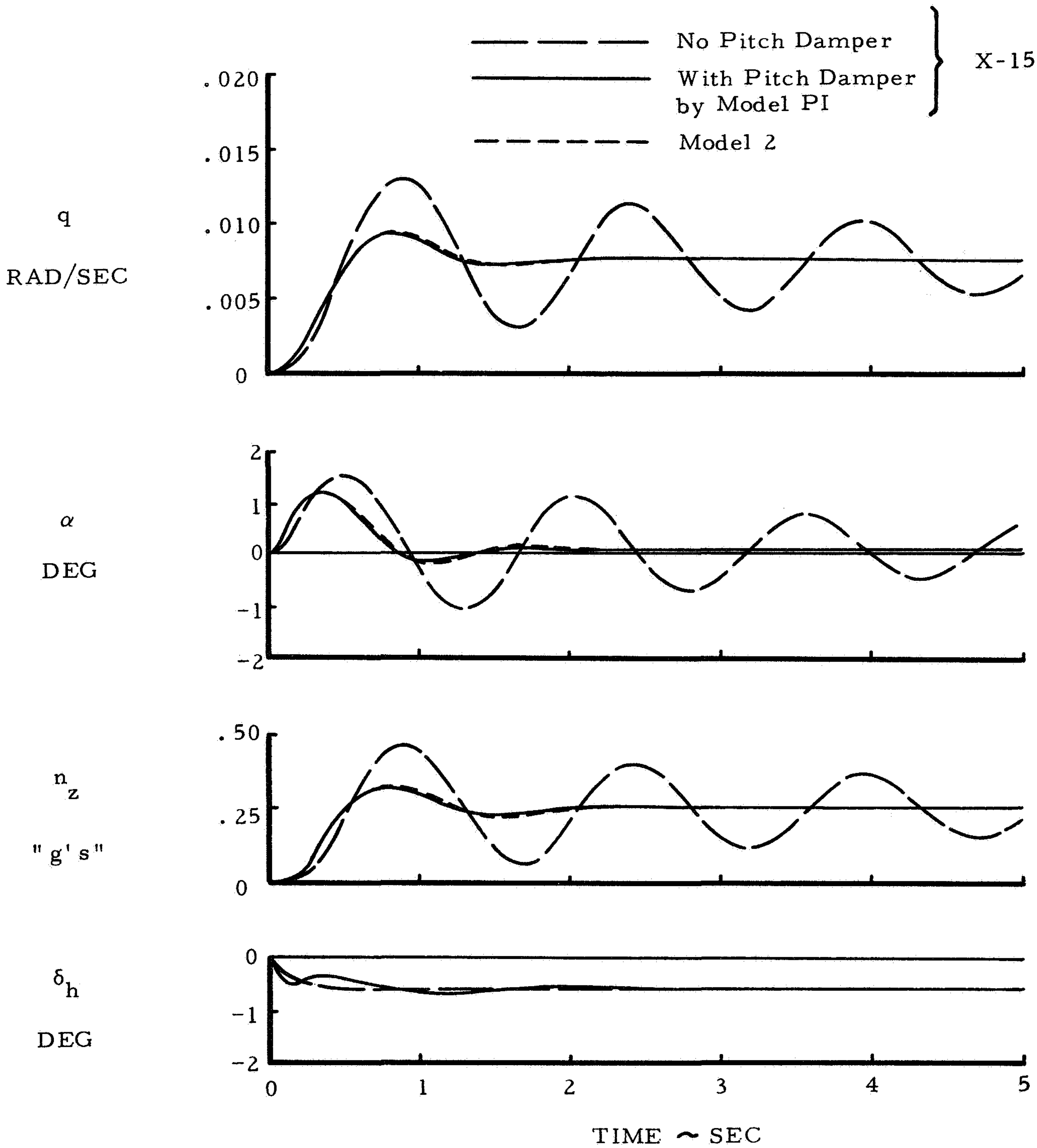


Figure 6-7 Time History Comparison of the Model PI Pitch Damper Design to Model 2 for a $\frac{1}{4}$ "g" Pull-Up

6.2 Lateral-Directional Stability Augmentation System for the X-15 Aircraft

Design of a multivariable flight control system by the Model PI method is illustrated here by an example that was previously treated in references 27 and 28 by linear optimal control theory and in reference 24 by the model-referenced ISE method of parameter optimization. The treatment presented here differs from these previous works in respect to the design object, as it did in the pitch damper example of section 6.1. In references 24, 27, and 28 the design objectives were to match the transient responses of a linear model to an initial condition in side-slip angle and to an initial condition in roll rate. The model used corresponds to the dynamic characteristics of a T-33 trainer modified to have generally good handling qualities. While one might state qualitatively that producing a design with transient responses similar to those of the model for the same initial conditions is generally desirable, one must still check the actual handling qualities characteristics of the resulting design against the appropriate criteria of the form discussed in section 2.1.1.2. The approach taken here in the Model PI design is fundamentally the same as that discussed in Chapter 5. In this example the design specifications are in the form of lateral-directional handling qualities criteria. Starting at this point, models are selected to represent the criteria. Then a SAS design is synthesized using the Model PI. Finally the design is judged by the actual handling qualities criteria.

The discussion at the beginning of section 3.2.3 in Chapter 3 on the formulation of multivariable control system design problems is quite pertinent to this example. Although it is most accurately formulated as a multivariable design problem, one can obtain a fairly good design by treating instead the design of a simplified single input/output system representing only the poles of the actual system. This approximate design approach will be presented first. Then two of the three methods for designing multivariable control systems proposed in section 3.2.3 will be illustrated.

6.2.1 Problem Formulation

Generally an aircraft's lateral-directional motion for small perturbations can be described approximately by

$$
\begin{aligned}
\dot{p} &= L_p p + L_r r + L_\beta \beta + L_{\delta_a} \delta_a + L_{\delta_v} \delta_v \\
\dot{r} &= N_p p + N_r r + N_\beta \beta + N_{\delta_a} \delta_a + N_{\delta_v} \delta_v \qquad (6\text{-}14) \\
\dot{\beta} &= (g/V)\phi - r + Y_\beta \beta + Y_{\delta_v} \delta_v
\end{aligned}
$$

where

p	is the roll rate
r	is the yaw rate
β	is the angle of sideslip
ϕ	is the bank angle
δ_a	is the roll control surface deflection
δ_v	is the vertical stabilizer deflection

L_p, L_r, L_β, L_{δ_v}, N_p, N_r, N_β, N_{δ_a}, N_{δ_v}, Y_β and Y_{δ_v} are dimensional stability and control derivatives.

The numerical values used for this example in references 24, 27, and 28 are listed in table 6-2. They are the characteristics of the X-15 aircraft at a Mach number of 5.5 and at an altitude of 147,000 feet (flight configuration and weight were not specified).

Using the numerical values from table 6-2 in equation (6-14), one can obtain the open-loop X-15 transfer function of roll control surface deflection to roll rates as

(6-15)

$$
\frac{p(s)}{\delta_a(s)} = \frac{23,700\ s\left[1 + \frac{2(0.007)}{(1.71)}s + \left(\frac{s}{1.71}\right)^2\right]}{\underbrace{\left(1 - \frac{s}{0.0033}\right)}_{\text{Spiral}}\underbrace{\left(1 + \frac{s}{0.0645}\right)}_{\text{Roll Subsidence}}\underbrace{\left[1 + \frac{2(0.0078)}{(1.72)}s + \left(\frac{s}{1.72}\right)^2\right]}_{\text{Dutch-roll}}}
$$

TABLE 6-2

LATERAL-DIRECTIONAL DIMENSIONAL DERIVATIVES
FOR THE X-15 AIRCRAFT
(From reference 28)

Mach Number = 5.5		Altitude = 147,000 feet
$L_p = -0.0625\ sec^{-1}$	$N_p = -0.0037\ sec^{-1}$	$Y_\beta = -0.0019\ sec^{-1}$
$L_r = -0.0376\ sec^{-1}$	$N_r = -0.0103\ sec^{-1}$	$Y_{\delta_v} = -0.0025\ sec^{-1}$
$L_\beta = 0.514\ sec^{-2}$	$N_\beta = 2.959\ sec^{-2}$	
$L_{\delta_a} = -5.116\ sec^{-2}$	$N_{\delta_a} = -0.2496\ sec^{-2}$	
$L_{\delta_v} = -2.325\ sec^{-2}$	$N_{\delta_v} = -1.432\ sec^{-2}$	$g/V = 0.00535\ sec^{-1}$

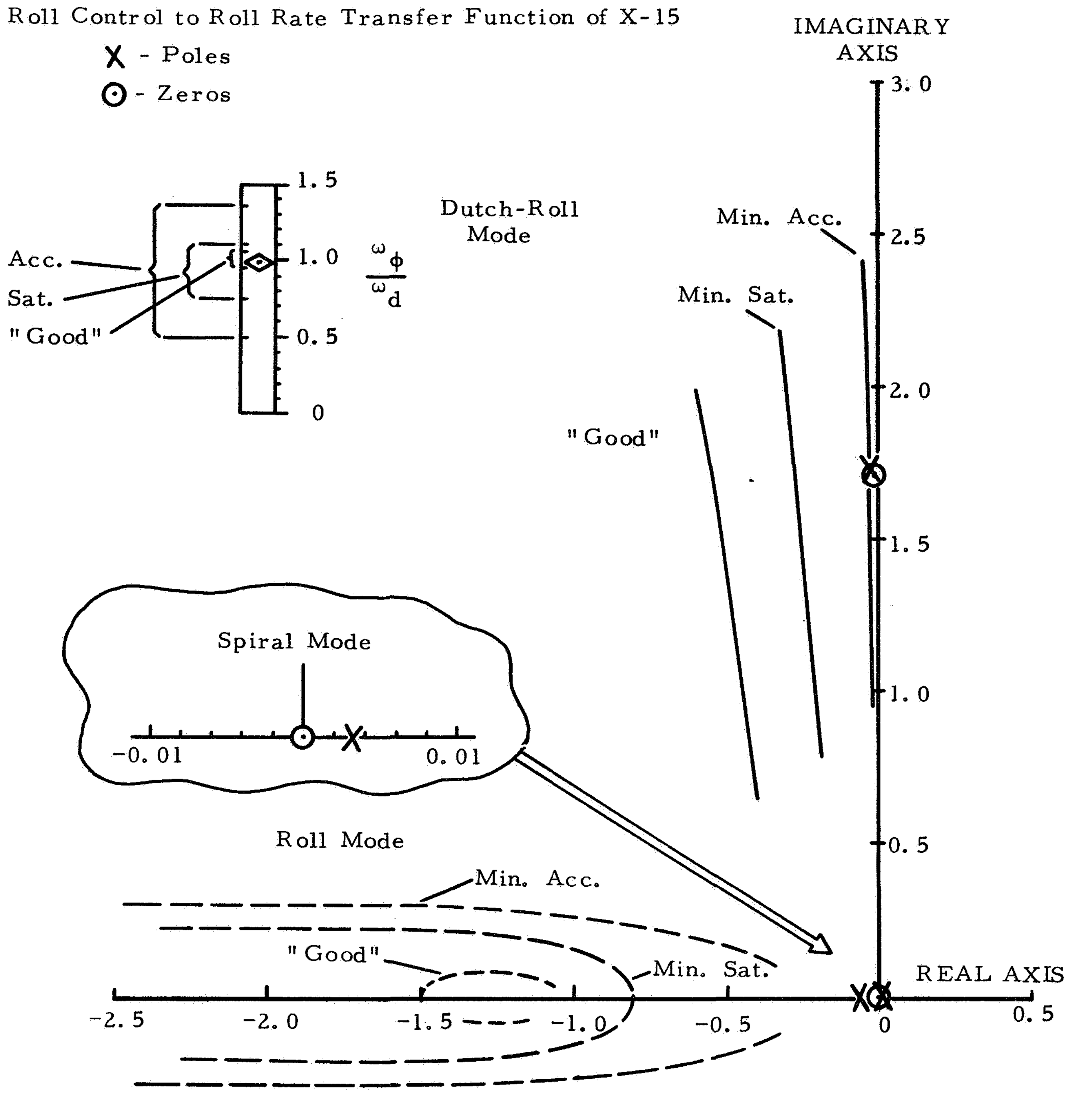


Figure 6-8 Lateral-Directional Handling Qualities of the X-15 Aircraft (SAS Off) at Mach No. 5. 5 and Altitude of 147, 000 Feet

The specific lateral-directional handling qualities criteria for this example were presented in figure 2-3 of Chapter 2. These are repeated in figure 6-8, and the poles and zeros of equation (6-15) have been superimposed. Based on these criteria the X-15 at this flight condition (without a SAS) would be predicted to have Unacceptable roll and Dutch-roll handling qualities. It is apparent that a SAS is needed to increase the roll damping and Dutch-roll mode damping in order to obtain Good or at least Satisfactory handling qualities.

Although the handling qualities criteria used here are described entirely by terms appearing in the $p(s)/\delta_a(s)$ transfer function, they can not be adequately represented in an analytical design method by just a model of the roll command to roll rate transfer characteristics. Part of the criteria requires the complex zeros of (6-15) to cancel or nearly cancel out the Dutch-roll mode poles. A model with this characteristic would have such a small Dutch-roll mode residue that it would essentially neglect that mode. In addition to a model of the roll command to roll rate transfer characteristics, it is necessary to have a model of some transfer characteristics in which the Dutch-roll mode is dominant. A suitable one is the yaw command to sideslip angle transfer function. Before selecting the specific models to use in this case, it is helpful to first form the corresponding system closed-loop transfer functions.

A possible SAS configuration* is shown in figure 6-9. This configuration is the same as that considered in reference 24, which pointed out the need for a high-pass (wash-out) filter to eliminate the yaw rate feedback in a steady turn. In this example, the dynamics of the motion sensors, the SAS servos, and the control surface actuators are neglected. A mathematical block diagram of the configuration to be considered is shown in figure 6-10. The wash-out filter shown, with a time constant of one second, is the same as that used in reference 24. The roll rate, sideslip and yaw rate feedback gains, denoted by k_p, k_β

---

* This is not the actual X-15 SAS configuration (51), which has a yaw rate feedback to the roll SAS servo and does not have a sideslip angle feedback to the yaw SAS servo.

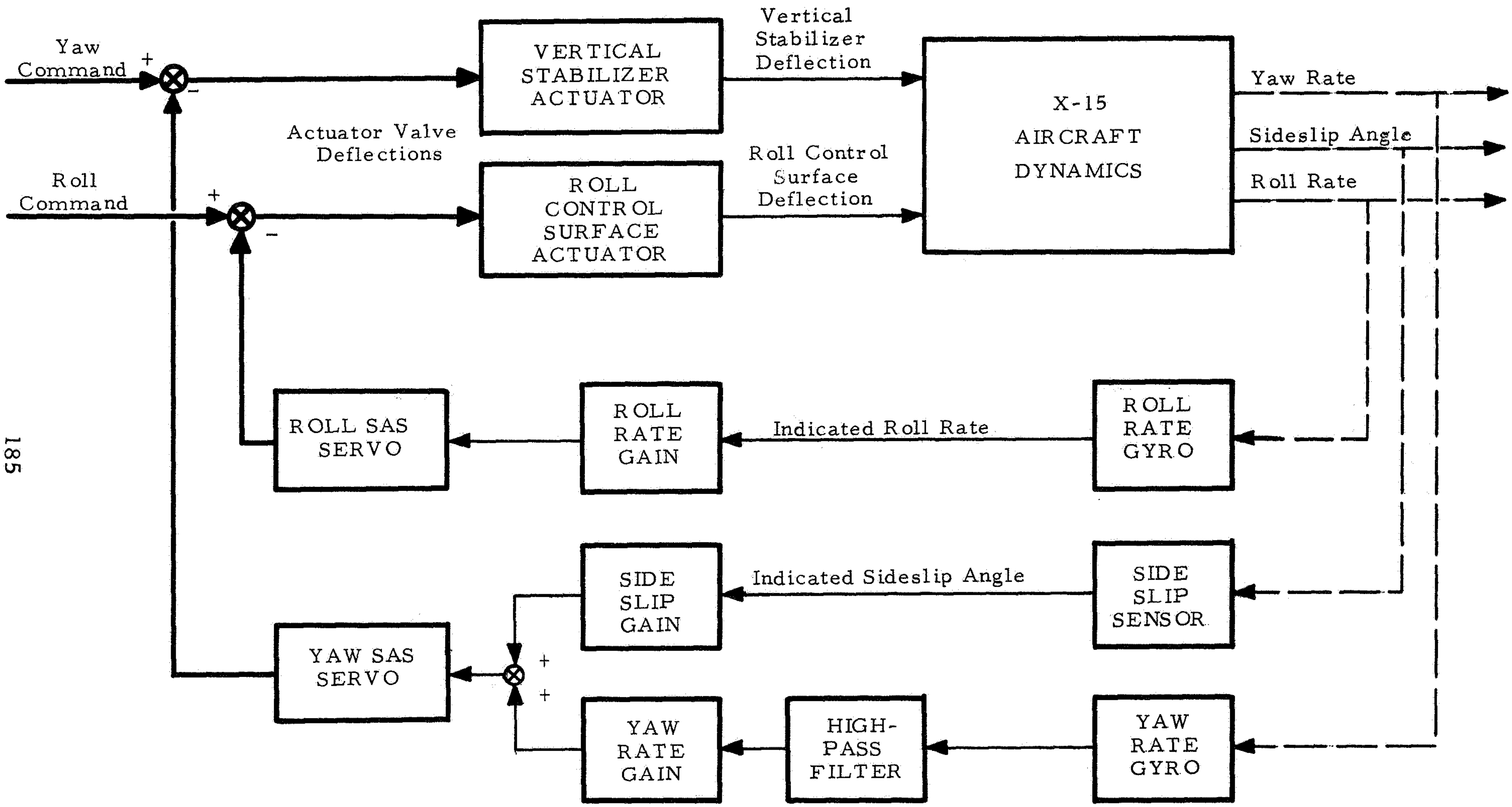
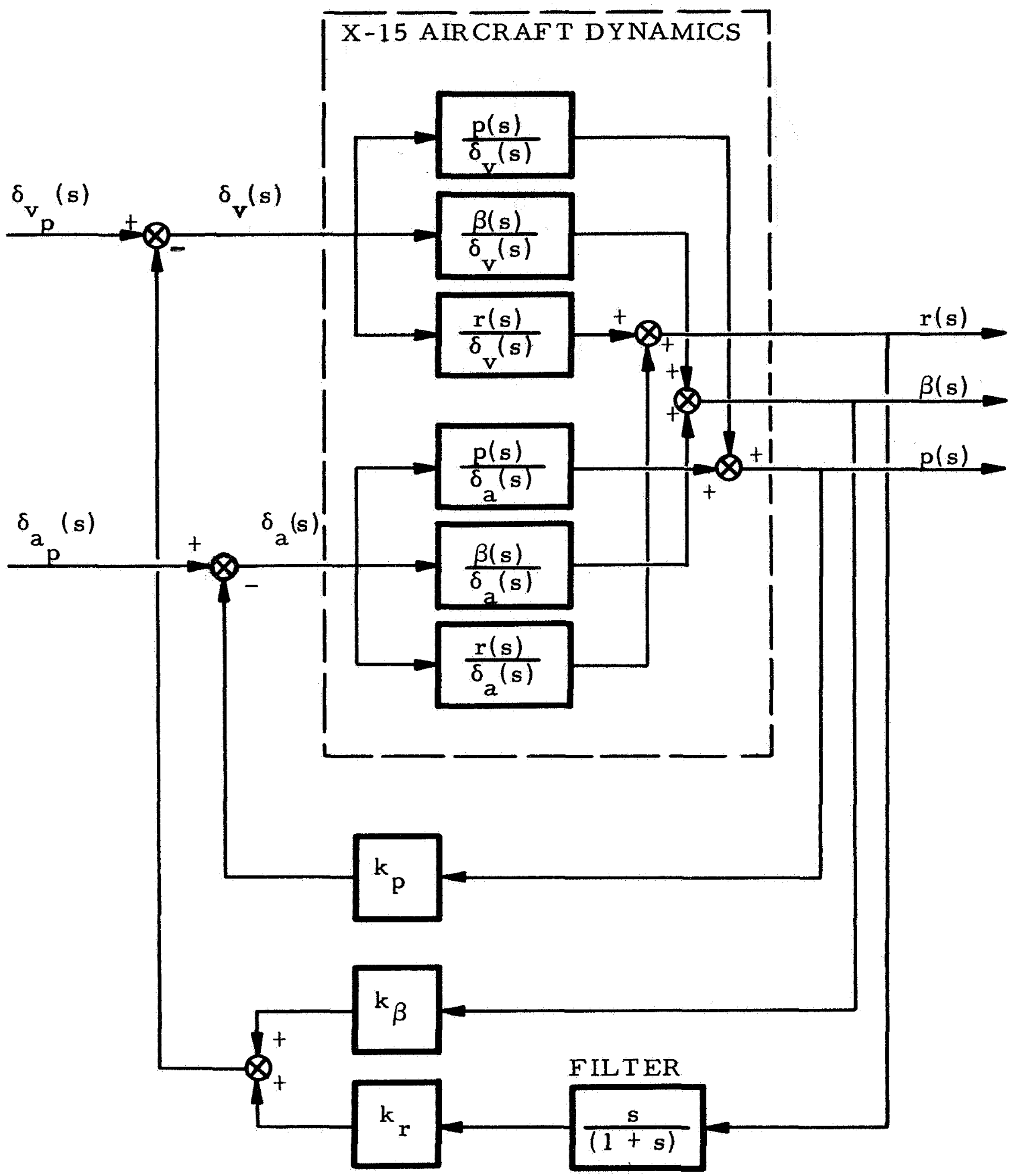


Figure 6-9 Functional Block Diagram of a Lateral-Directional SAS Configuration



k_p, k_β, and k_r are free design parameters

Figure 6-10 Block Diagram of the SAS Configuration Considered and k_r respectively, are the free design parameters. As shown in figure 6-10, the control surface deflection, $\delta_a(s)$ and $\delta_v(s)$ are related to the feedback variables and the commands by

$$\delta_a(s) = \delta_{a_p}(s) - k_p p(s) \tag{6-16}$$

$$\delta_v(s) = \delta_{v_p}(s) - k_r \left(\frac{s}{1+s}\right) r(s) - k_\beta \beta(s) \tag{6-17}$$

where

δ_{a_p} is the pilot roll command

δ_{v_p} is the pilot yaw command

The desired closed-loop transfer functions are for pilot roll command to roll rate and pilot yaw command to sideslip angle. It is straight forward, though admittedly tedious, to obtain these from the Laplace transform of (6-14) and equations (6-16) and (6-17). For the numerical values listed in table 6-2, these are

$$\frac{p(s)}{\delta_{a_p}(s)} = \frac{N_{[\delta_{a_p}, p]}(s)}{D(s)} \tag{6-18}$$

$$\frac{\beta(s)}{\delta_{v_p}(s)} = \frac{N_{[\delta_{v_p}, \beta]}(s)}{D(s)} \tag{6-19}$$

where

$$D(s) = s^5 + a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0$$

$$a_4 = 1.085 - 5.116\, k_p - 1.432\, k_r - 0.0025\, k_\beta$$

$$a_3 = 3.045 - 5.239\, k_p - 0.105\, k_r + 1.429\, k_\beta + 6.746\, k_p k_r + 0.0128\, k_p k_\beta$$

$$a_2 = 3.128 - 15.134\, k_p - 0.000494\, k_r + 1.513\, k_\beta + 0.118\, k_p k_r - 6.733\, k_p k_\beta$$

$$a_1 = 0.180 - 15.011\, k_p - 0.0329\, k_r + 0.068\, k_\beta - 6.746\, k_p k_\beta$$

$$a_0 = -0.000624 - 0.000416\, k_\beta \qquad (6\text{-}20)$$

$$N_{[\delta_{a_p}, p]}(s) = b_{14}s^4 + b_{13}s^3 + b_{12}s^2 + b_{11}s + b_{10}$$

$$b_{14} = -5.116$$

$$b_{13} = -5.239 + 6.746\, k_r + 0.0279\, k_\beta$$

$$b_{12} = -15.134 + 0.118\, k_r - 6.718\, k_\beta$$

$$b_{11} = -15.011 - 6.746\, k_\beta$$

$$b_{10} = 0 \qquad (6\text{-}21)$$

$$N_{[\delta_{v_p}, \beta]}(s) = b_{24}s^4 + b_{23}s^3 + b_{22}s^2 + b_{21}s + b_{20}$$

$$b_{24} = -0.0025$$

$$b_{23} = 1.429 + 0.0128\, k_p$$

$$b_{22} = 1.500 - 6.733\, k_p$$

$$b_{21} = 0.068 - 6.746\, k_p \qquad (6\text{-}22)$$

$$b_{20} = -0.000416$$

Two models can now be established to represent the design specifications for these two transfer relationships. In this case it is relatively simple to select models with the same structure as the closed-loop system transfer functions by referring directly to the handling qualities criteria, figure 6-8. First consider the roll command to roll rate model, i. e.

$$\frac{p_m(s)}{\delta_{a_p}(s)} = \frac{S_{1m} s \left(1 + \frac{s}{\tau_2}\right)\left[1 + \frac{2\zeta_{\phi_m}}{\omega_{\phi_m}} s + \left(\frac{s}{\omega_{\phi_m}}\right)^2\right]}{\left(1 + \frac{s}{\tau_1}\right)\left(1 + \frac{s}{\tau_R}\right)\left(1 + \frac{s}{\tau_S}\right)\left[1 + \frac{2\zeta_m}{\omega_m} s + \left(\frac{s}{\omega_m}\right)^2\right]} \tag{6-23}$$

Four of the models' poles can be selected in regions corresponding to "Good" handling qualities in figure 6-8, and the complex zeros can be selected sufficiently close to the Dutch-roll poles. The wash-out filter causes an extra pole and zero not covered explicitly by the criteria in figure 6-8. However the residue of this extra mode must be kept small in order for it not to affect the handling qualities so it is wise to select the zero relatively close to the pole. Following these guide lines the model is selected to be

$$\frac{p_m(s)}{\delta_{a_p}(s)} = \frac{-624\, s \left(1 + \frac{s}{3.5}\right)\left[1 + \frac{2(0.63)}{(1.98)} s + \left(\frac{s}{1.98}\right)^2\right]}{\underbrace{\left(1 + \frac{s}{4.0}\right)}_{\text{Filter}}\underbrace{\left(1 + \frac{s}{1.4}\right)}_{\text{Roll Subsidence}}\underbrace{\left(1 + \frac{s}{0.005}\right)}_{\text{Spiral}}\underbrace{\left[1 + \frac{2(0.6)}{(2.0)} s + \left(\frac{s}{2.0}\right)^2\right]}_{\text{Dutch-roll}}} \tag{6-24}$$

where the sensitivity selected is such that the model would produce the same rolling moment due to roll control surface deflection as the X-15. The model's poles and zeros are superimposed on the handling qualities criteria in figure 6-11. An aircraft with these characteristics would be predicted to have "Good" handling qualities.

Roll Control to Roll Rate Transfer Function, Equation (6-24)

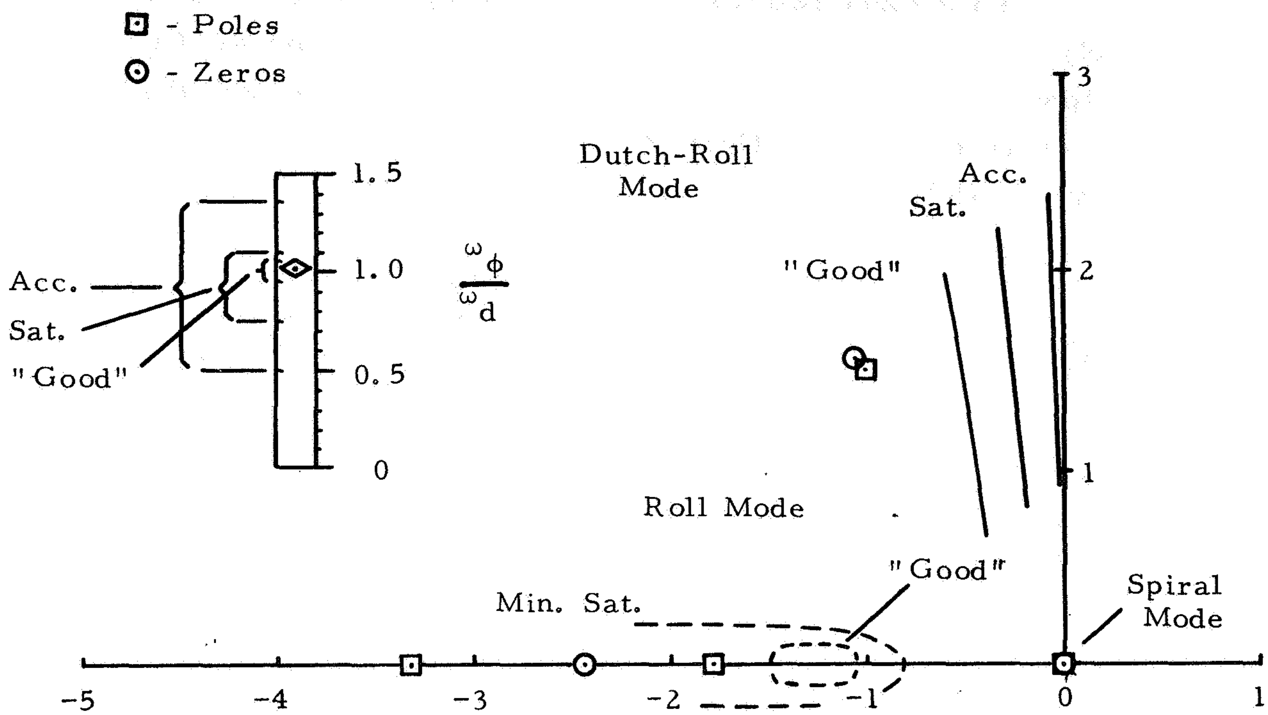


Figure 6-11 Estimated Handling Qualities of the Roll Control Model

It is only necessary to select zeros to establish the yaw command to sideslip angle model. One can use the closed-loop system transfer function numerator (6-21) and (6-22) as a guide. Although not apparent from (6-22) the quantity (1 + s) is a factor of the numerator of (6-21), due to the wash-out filter, and thus one model zero should correspond to this factor. Since b_{20} is very small compared to the other coefficients, the corresponding coefficient in the model could be neglected giving a zero at the origin. Factoring out (1 + s) and neglecting b_{20} in (6-21) would reduce the numerator to approximately

$$N_{[\delta_{v_p}, \beta]}(s) \approx -0.0025\, s(1 + s)\left[s^2 - 400\,(c_1 s + c_0)\right] \qquad (6\text{-}25)$$

where c_1 and c_0 are functions of k_p and are at least one order of magnitude less than 400, so that the quadratic term would have factors of approximately $(s-400\ c_1)$ and $(s + c_0/c_1)$. Since one zero will be far into the right half plane the model should have a zero in that vicinity, e.g. at about +400. The remaining zero should be close to the roll mode pole so that the Dutch-roll mode is dominant. Then the model selected for the yaw command to sideslip angle transfer function is

$$\frac{\beta_m(s)}{\delta_{v_p}(s)} = \frac{12.06\ s(1+s)\left(1+\frac{s}{1.35}\right)\left(1-\frac{s}{400}\right)}{\left(1+\frac{s}{4.0}\right)\left(1+\frac{s}{1.4}\right)\left(1+\frac{s}{0.005}\right)\left[1+\frac{2(0.6)}{(2.0)}s+\left(\frac{s}{2.0}\right)^2\right]} \qquad (6\text{-}26)$$

The sensitivity used in (6-26) is such that the model would produce the same side force due to the vertical stabilizer deflection as the X-15.

The geometrical representations of the models $p_m(s)/\delta_{a_p}(s)$ and $\beta_m(s)/\delta_{v_p}(s)$ are

$$\{\underline{\tilde{\alpha}}\ ;\ \underline{\tilde{x}}_{1m_0}\}$$
$$\{\underline{\tilde{\alpha}}\ ;\ \underline{\tilde{x}}_{2m_0}\} \qquad (6\text{-}27)$$

respectively, where

$$\underline{\tilde{\alpha}}' = [0.112 \quad 22.575 \quad 35.143 \quad 22.609 \quad 7.805 \quad 1] \qquad (6\text{-}28)$$

and

$$\underline{\tilde{x}}'_{1m_0} = [\underline{x}'_{1m_0} \ \vdots \ -\underline{x}'_{1m_0}\underline{\alpha}]$$

$$\underline{x}'_{1m_0} = [0 \quad -5.116 \quad 9.23 \quad -20.73 \quad 64.7] \qquad (6\text{-}29)$$

$$\tilde{\underline{x}}'_{2m_0} = [\underline{x}'_{2m_0} \;\vdots\; -\underline{x}'_{2m_0}\alpha]$$

$$\underline{x}'_{2m_0} = [0 \quad -0.0025 \quad 1.014 \quad -5.508 \quad 21.52] \tag{6-30}$$

The elements of $\underline{x}_{1m_0}$ and $\underline{x}_{2m_0}$ were computed from the formula (3-8) for (6-24) and (6-26) respectively.

6.2.2 An Approximate Design Method

The design specifications in this example are primarily requirements on the closed-loop poles, with the exception of the ω_ϕ/ω_d requirement. Therefore it may be possible to obtain an approximate design by considering only the dominant poles of the closed-loop system and model, i.e. neglect the spiral mode and all zeros during the design process. Consider the hypothetical system and model given respectively by

$$\frac{y(s)}{u(s)} = \frac{a_1}{s^4 + a_4 s^3 + a_3 s^2 + a_2 s + a_1} \tag{6-31}$$

where a_1, a_2, a_3 and a_4 are functions of k_p, k_β and k_r given by (6-20) and

$$\frac{y_m(s)}{u(s)} = \frac{1}{\left(1 + \frac{s}{4.0}\right)\left(1 + \frac{s}{1.4}\right)\left[1 + \frac{2(0.6)}{(2.0)} s + \left(\frac{s}{2.0}\right)^2\right]} \tag{6-32}$$

The denominator of (6-31) represents essentially the denominator of (6-18) when the spiral mode is neglected and the denominator of (6-32) is the model denominator neglecting the spiral mode. The values of k_p, k_β and k_r that give the best match of this hypothetical system to the model, in the Model PI sense, should also produce pole locations for the actual closed-loop system (6-18) and (6-19) that are close to the model poles shown on figure 6-11 except possibly the spiral mode pole.

The model's extended coefficient vector for (6-32) is

$$\underline{\alpha}' = [22.4 \quad 35.03 \quad 22.57 \quad 7.8 \quad 1] \qquad (6\text{-}33)$$

This problem is now set up as a single input/output system design and in the form discussed in Chapter 5 for applying the general computer program. The subroutine COEF is written to generate a_1, a_2, a_3 and a_4 from (6-20).

Model PI Solution for the Approximate Design Approach

An initial choice for the free parameters of

$k_p = -0.01$ deg δ_a per deg/sec p

$k_\beta = 0.01$ deg δ_v per deg β

$k_r = -0.1$ deg δ_v per deg/sec r

rested in final values of

$k_p = -0.429$ deg δ_a per deg/sec p

$k_\beta = 4.283$ deg δ_v per deg β

$k_r = -2.680$ deg δ_v per deg/sec r

The corresponding transfer function for the hypothetical system (6-31) is

$$\frac{y(s)}{u(s)} = \frac{1}{\left(1 + \frac{s}{3.32}\right)\left(1 + \frac{s}{1.78}\right)\left[1 + \frac{2(0.555)}{(1.81)} s + \left(\frac{s}{1.81}\right)^2\right]} \qquad (6\text{-}34)$$

The poles of (6-34) are seen to be reasonably close to the model poles (6-32). If the above final values for the free parameters are used in (6-20) and (6-21) to compute the roll command to roll rate transfer function of the actual system, the result is

$$\frac{p(s)}{\delta_{a_p}(s)} = \frac{18,248\ s\left(1+\frac{s}{2.43}\right)\left[1+\frac{2(0.560)}{(1.88)}s+\left(\frac{s}{1.88}\right)^2\right]}{\left(1+\frac{s}{3.32}\right)\left(1+\frac{s}{1.78}\right)\left(1-\frac{s}{0.00012}\right)\left[1+\frac{2(0.552)}{(1.82)}s+\left(\frac{s}{1.82}\right)^2\right]} \tag{6-35}$$

The poles and zeros of (6-35) are superimposed in the handling qualities criteria in figure 6-12.

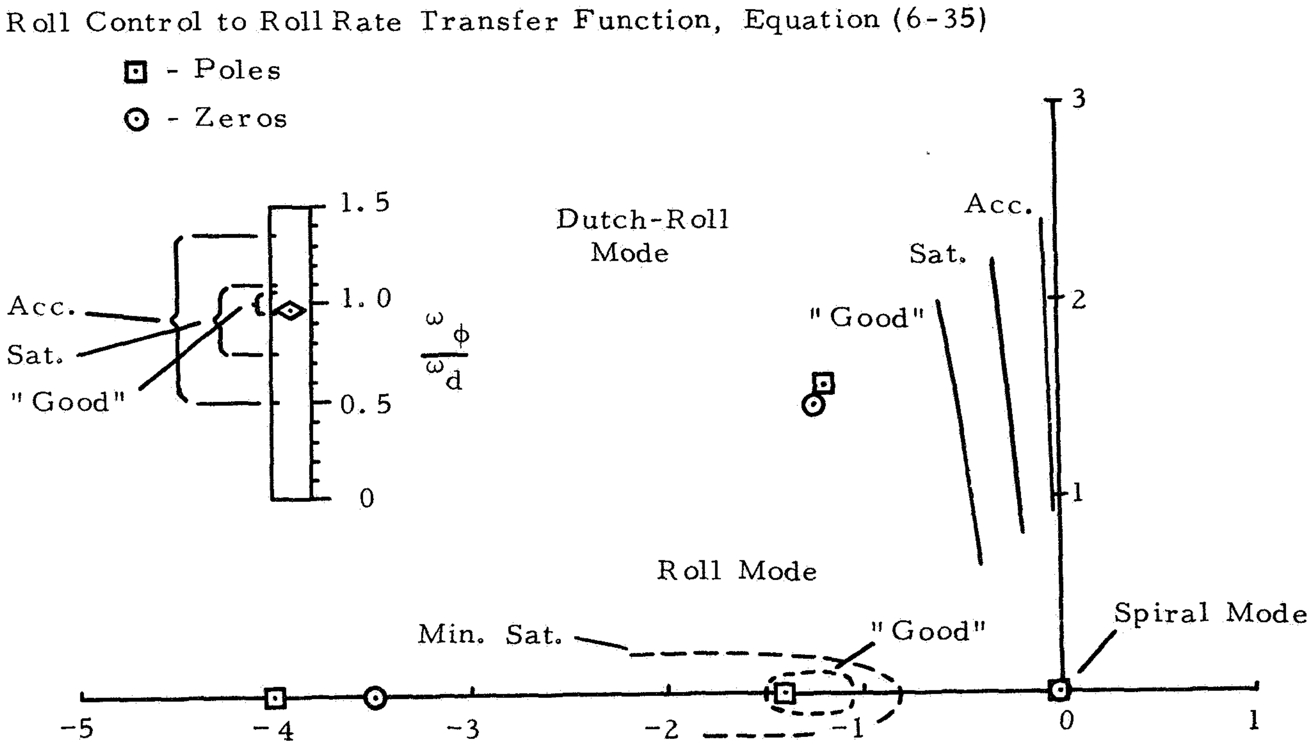


Figure 6-12 Estimated Handling Qualities of the Model PI Design Using an Approximate Design Method

From analyzing figure 6-12 one would probably conclude that this design should produce at least Satisfactory handling qualities. The roll mode pole is not within the "Good" handling qualities region but is well within the Satisfactory region and close to the "Good region. The Dutch-roll mode has "Good" handling qualities characteristics including the ω_ϕ/ω_d ratio. The effect of the system's closed-loop zeros was not included in this approximate design approach, but the closed-loop zeros are dependent on the values of the free design parameters as indicated by equation (6-21). The particular functional dependence of the closed-loop zeros on the free parameters in this example happens to be such that the same values of the free parameters that produce good pole locations also produces good zero locations. One should not expect this to happen in general.

The approximate approach based only on the system and model poles resulted in a good design, in this case, and one would not have to proceed with the design process any further using a more complete description of the system and model. Even in cases where this approximate method does not in itself produce a satisfactory design, it is a good first step in the synthesis process whenever the specifications include specific requirements on closed-loop pole locations.

6.2.3 Design by the Second Method for Multivariable Systems

The second method for multivariable system design using the Model PI is described in Chapter 3, section 3.2.3.2. To apply this method it is necessary to define a weighted average of the two models (6-27) that corresponds to (3-103) in Chapter 3. In this example the two models should be weighted equally since the roll mode characteristics, which are dominant in $p_m(s)/\delta_{a_p}(s)$, and the Dutch-roll mode characteristics, which are dominant in $\beta_m(s)/\delta_{v_p}(s)$, are equally important. Therefore the priority number c should be 0.5. The pseudo IC vector of the weighted averaged model is then given by *

---

* Equation (3-104) reduces to (6-36) because $\underline{x}_{m_0} = \widetilde{W}\tilde{\underline{x}}_{m_0}$ when the system and model are of the same order.

$$\underline{\tilde{x}}'_{m_0} = [\underline{x}'_{m_0} \;\vdots\; -\underline{x}'_{m_0}\underline{\alpha}]$$

$$\underline{x}_{m_0} = 0.5 \frac{\underline{x}_{1m_0}}{\|\underline{x}_{1m_0}\|} + 0.5 \frac{\underline{x}_{2m_0}}{\|\underline{x}_{2m_0}\|} \tag{6-36}$$

where $\underline{x}_{1m_0}$ and $\underline{x}_{2m_0}$ are given by(6-29) and (6-30) respectively. Using the numerical values of $\underline{x}_{1m_0}$ and $\underline{x}_{2m_0}$ gives

$$\underline{x}'_{m_0} = [0 \quad -0.0376 \quad +0.0899 \quad -0.275 \quad +0.954] \tag{6-37}$$

The model's extended coefficient vector is, of course, given by (6-28).

The subroutine CALSYS MOD 2 forms the weighted averaged system from (6-18) and (6-19) which are entered into the program via subroutines COEF1 and COEF2 respectively. The relationships (6-20) and (6-21) are written into COEF1 and the relationships (6-22) are written into COEF2. The value of CC in COEF1 is given as CC = c = 0.5 and in COEF2 as CC = (1-c) = 0.5. The pseudo IC vector (6-37) is entered as input data (see Appendix B).

Model PI Solution for Second Method

Using a pseudo IC weighting factor in the Model PI of 4.0×10^{-6} and an initial choice for the free parameters of

$k_p = -0.429$ deg δ_a per deg/sec p

$k_\beta = 4.283$ deg δ_v per deg β

$k_r = -2.680$ deg δ_v per deg/sec r

which were the final values of the approximate design in section 6.2.2, resulted in final values of

$k_p \;=\; -0.286 \text{ deg } \delta_a \text{ per deg/sec } p$

$k_\beta \;=\; 4.209 \text{ deg } \delta_v \text{ per deg } \beta$

$k_r \;=\; -3.086 \text{ deg } \delta_v \text{ per deg/sec } r$

Using these values in equations (6-20) - (6-22) gives closed-loop transfer functions of

$$\frac{p(s)}{\delta_{a_p}(s)} = \frac{18,273\, s\left(1 + \frac{s}{3.24}\right)\left[1 + \frac{2(.566)}{(1.62)} s + \left(\frac{s}{1.62}\right)^2\right]}{\left(1 + \frac{s}{3.84}\right)\left(1 + \frac{s}{1.37}\right)\left(1 - \frac{s}{0.00018}\right)\left[1 + \frac{2(0.555)}{(1.57)} s + \left(\frac{s}{1.57}\right)^2\right]} \tag{6-38}$$

and

$$\frac{\beta(s)}{\delta_{v_p}(s)} = \frac{0.175\left(1 - \frac{s}{0.00021}\right)(1 + s)\left(1 + \frac{s}{1.396}\right)\left(1 - \frac{s}{573}\right)}{\left(1 + \frac{s}{3.84}\right)\left(1 + \frac{s}{1.37}\right)\left(1 - \frac{s}{0.00018}\right)\left[1 + \frac{2(0.555)}{(1.57)} s + \left(\frac{s}{1.57}\right)^2\right]} \tag{6-39}$$

This design is compared to the handling qualities criteria in figure 6-13 which shows it to have "Good" handling qualities characteristics. The only significant difference between this design and that obtained by the approximate approach of the previous section is the roll mode pole location. In this case the roll mode pole is within the "Good" handling qualities region. However the boundaries are not sharply defined and the difference in the roll mode poles is so slight that the difference in handling qualities may hardly be noticeable.

Roll Control to Roll Rate Transfer Function, Equation (6-38)

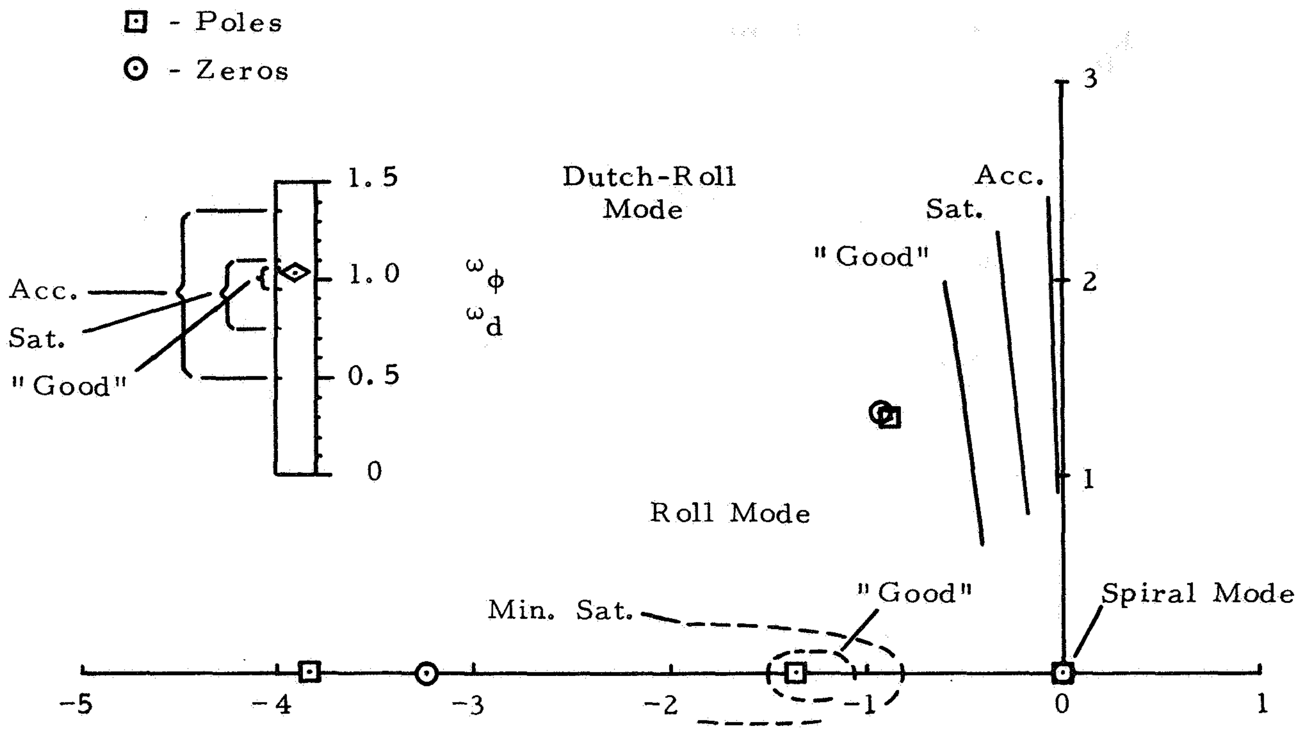


Figure 6-13 Estimated Handling Qualities of the Model PI Design Using the Second Method for Multivariable Systems

6.2.4 Design by the Third Method for Multivariable Systems

The third design method for multivariable systems described in Chapter 3, section 3.2.3.3 is to apply the single input/output method alternately to the design of the closed-loop " systems" represented by (6-18) and (6-19) with equations (6-24) and (6-26) as the respective models. The design example is already in the proper form to apply this approach directly. All that is necessary is to write the subroutine COEF for each "system" and provide the input data cards for the appropriate model.

First consider selecting the free parameter values k_p, k_β, and k_r that provide the best Model PI match of the system's p/δ_{a_p} transfer characteristics to the model's p_m/δ_{a_p} transfer characteristics represented by its extended coefficient vector (6-28) and pseudo IC vector

(6-29). The subroutine COEF for this step is written to generate a_0, a_1, a_2, a_3 and a_4 from (6-20) and b_{10}, b_{11}, b_{12}, b_{13} and b_{14} from (6-21). A pseudo IC weight factor of 4.0×10^{-6} is used in the Model PI throughout.

Model PI Solution for p/δ_{a_p} with k_p, k_β, and k_r Free

Starting from an initial choice for the free parameters

$$k_p = -0.429 \text{ deg } \delta_a \text{ per deg/sec } p$$

$$k_\beta = 4.283 \text{ deg } \delta_v \text{ per deg } \beta$$

$$k_r = -2.680 \text{ deg } \delta_v \text{ per deg/sec } r$$

which were the final values of the approximate design approach, resulted in final values for this step of

$$k_p = -0.270 \text{ deg } \delta_a \text{ per deg/sec } p$$

$$k_\beta = 4.206 \text{ deg } \delta_v \text{ per deg } \beta$$

$$k_r = -2.974 \text{ deg } \delta_v \text{ per deg/sec } r$$

The next step is to hold k_p fixed at the value −0.270 and then determine the Model PI solution for matching the system's β/δ_{v_p} transfer characteristics to the model's β_m/δ_{v_p} characteristics with just k_β and k_r as free parameters. The roll rate feedback gain is held constant because it is known to be the least effective of the three parameters in changing the sideslip response to a yaw command. The subroutine COEF in this step generates the a_0, a_1, a_2, a_3 and a_4 but with $k_p = -0.270$, and b_{20}, b_{21}, b_{22}, b_{23}, and b_{24} from (6-22). Equations (6-28) and (6-30) represent the model to be used here.

Model PI Solution for β/δ_{v_p} with k_β and k_r Free

Starting with an initial choice of k_β and k_r given by the final values of the above step, resulted in final values for this step of

$$k_\beta = 4.562 \text{ deg } \delta_v \text{ per deg } \beta$$

$$k_r = -3.084 \text{ deg } \delta_v \text{ per deg/sec r}$$

These values together with the previous value for k_p should be a good compromise between matching the two models. To check this, the first step is repeated with k_p free and k_β and k_r fixed at these latter values. The procedure for doing this should be clear from the above steps. The resulting Model PI design did not change k_p to within three significant figures. Therefore the final solution by third method for multivariable system design is the following:

Model PI Solution for Third Method

$$k_p = -0.270 \text{ deg } \delta_a \text{ deg/sec p}$$

$$k_\beta = 4.562 \text{ deg } \delta_v \text{ per deg } \beta$$

$$k_r = -3.084 \text{ deg } \delta_v \text{ per deg/sec r}$$

Using these values in equation (6-20) - (6-22) gives closed-loop transfer functions of

$$\frac{p(s)}{\delta_{a_p}(s)} = \frac{18,153\, s\left(1 + \frac{s}{3.01}\right)\left[1 + \frac{2(0.582)}{(1.70)} s + \left(\frac{s}{1.70}\right)^2\right]}{\left(1 + \frac{s}{3.68}\right)\left(1 + \frac{s}{1.31}\right)\left(1 - \frac{s}{0.00019}\right)\left[1 + \frac{2(0.575)}{(1.64)} s + \left(\frac{s}{1.64}\right)^2\right]} \tag{6-40}$$

$$\frac{\beta(s)}{\delta_{v_p}(s)} = \frac{0.165\left(1 - \frac{s}{0.00022}\right)(1 + s)\left(1 + \frac{s}{1.319}\right)\left(1 - \frac{s}{573}\right)}{\left(1 + \frac{s}{3.68}\right)\left(1 + \frac{s}{1.31}\right)\left(1 - \frac{s}{0.00019}\right)\left[1 + \frac{2(0.575)}{(1.64)} s + \left(\frac{s}{1.64}\right)^2\right]} \tag{6-41}$$

This design, as shown on figure 6-14, would also be predicted to have "Good" handling qualities characteristics.

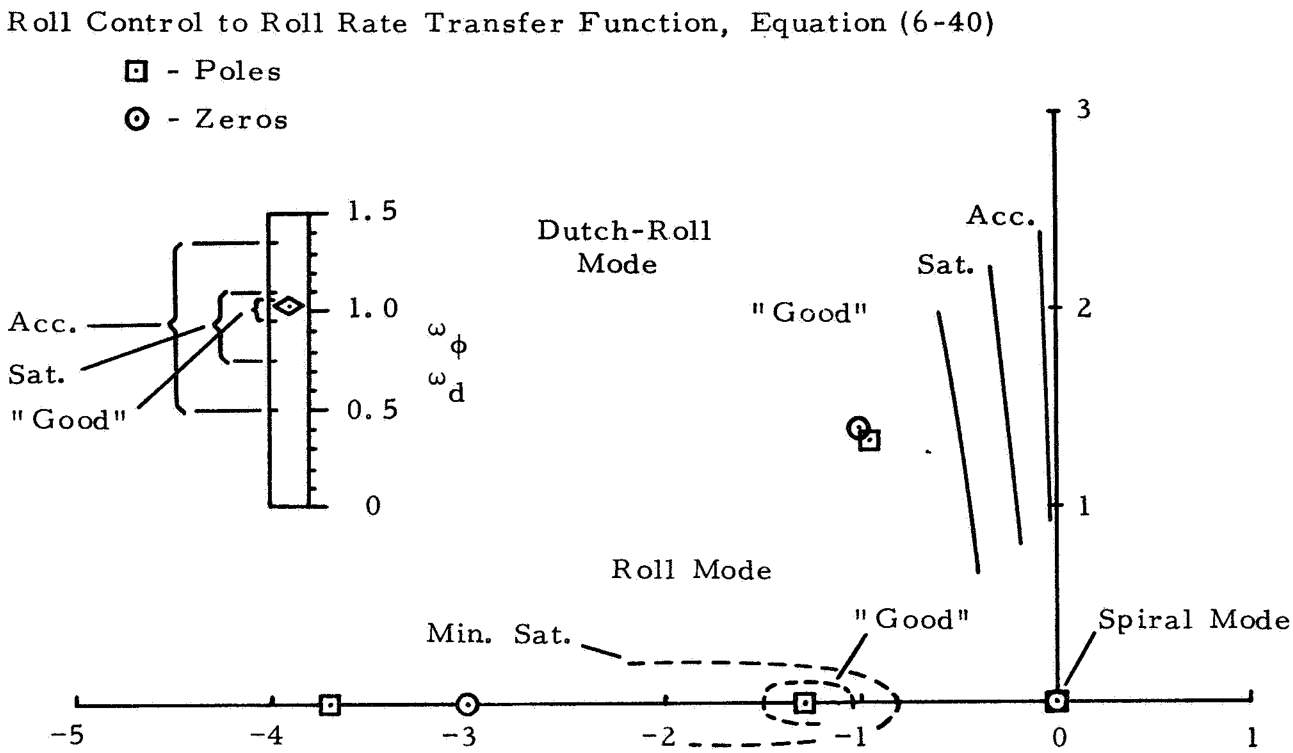


Figure 6-14 Estimated Handling Qualities of the Model PI Design Using the Third Method for Multivariable Systems

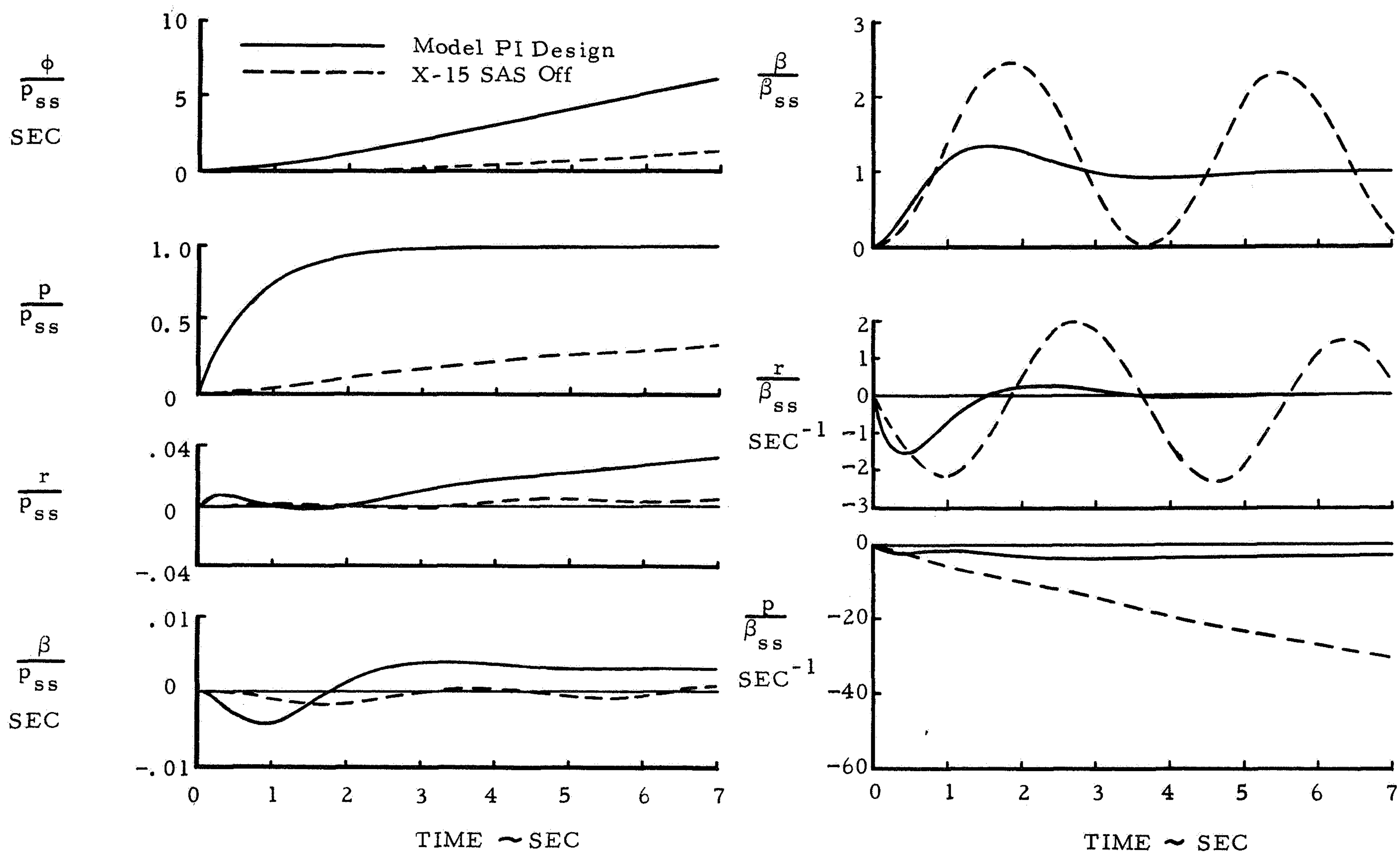
Comparing figures 6-13 and 6-14 one sees that there is no significant difference between the design obtained from the "second" and "third" methods for multivariable system design.

Obtaining the final solution in this example by the "third" method actually involved four applications of the single input/output Model PI design procedure, with the first application being the approximate design approach of section 6.2.2. The second application established the value of k_p, the third application established the values of k_β and k_r, and the last application verified that these values were satisfactory for the p/δ_{a_p} as well as the β/δ_{v_p} transfer characteristics.

6.2.5 Comparison and Discussion of Results

The first and most important result is that the Model PI produced conceptual designs for a stability augmentation system that would provide Satisfactory to "Good" lateral-directional handling qualities for the X-15 at the flight condition considered. That was the objective of the example. Figures 6-12, 6-13, and 6-14 show that the design specification of obtaining at least Satisfactory handling qualities is satisfied by all three of these Model PI designs. The improvement in the dynamic response of the X-15 with a stability augmentation system is illustrated in figure 6-15 for the design obtained by the third method. The other two designs give quite similar improvements.

It isn't possible to compare these designs directly to the actual X-15 SAS because the feedback configurations are different. But the roll and yaw rate feedback gain ranges used in the actual X-15 SAS should indicate the general gain levels that would be reasonable to implement. The roll rate feedback gain on the X-15 SAS (51) ranges from 0 to −0.5 deg δ_a per deg /sec p, and the yaw rate feedback gain ranges from 0 to −0.3 deg δ_v per deg/sec r. The Model PI designs have roll rate gains in the range of −0.27 to −0.43 deg δ_a per deg/sec p which is within the X-15 SAS range and thus are reasonable to implement. The yaw rate feedback gains obtained here, ranging from about −2.7 to −3.1 deg δ_v per deg r, are an order of magnitude greater than the corresponding X-15 SAS range, which indicates that they are probably too high or at least higher than necessary. This is not a fault of the design procedure but rather the model selected. Referring back to figure 6-11 one can see that the Dutch-roll mode damping selected for the model was very conservative. A much lower damping ratio, as low as 0.2, could be used for the model and still have Satisfactory Dutch-roll handling qualities characteristics, providing the other conditions are met. Since the yaw rate gain affects the Dutch-roll damping primarily, a lower value would result from the Model PI design if a lower damping ratio were used in the model. Repeating the Model PI design procedure with a less conservative model would be a better approach for obtaining a



(a) A Step Roll Command Input

(b) A Step Yaw Command Input

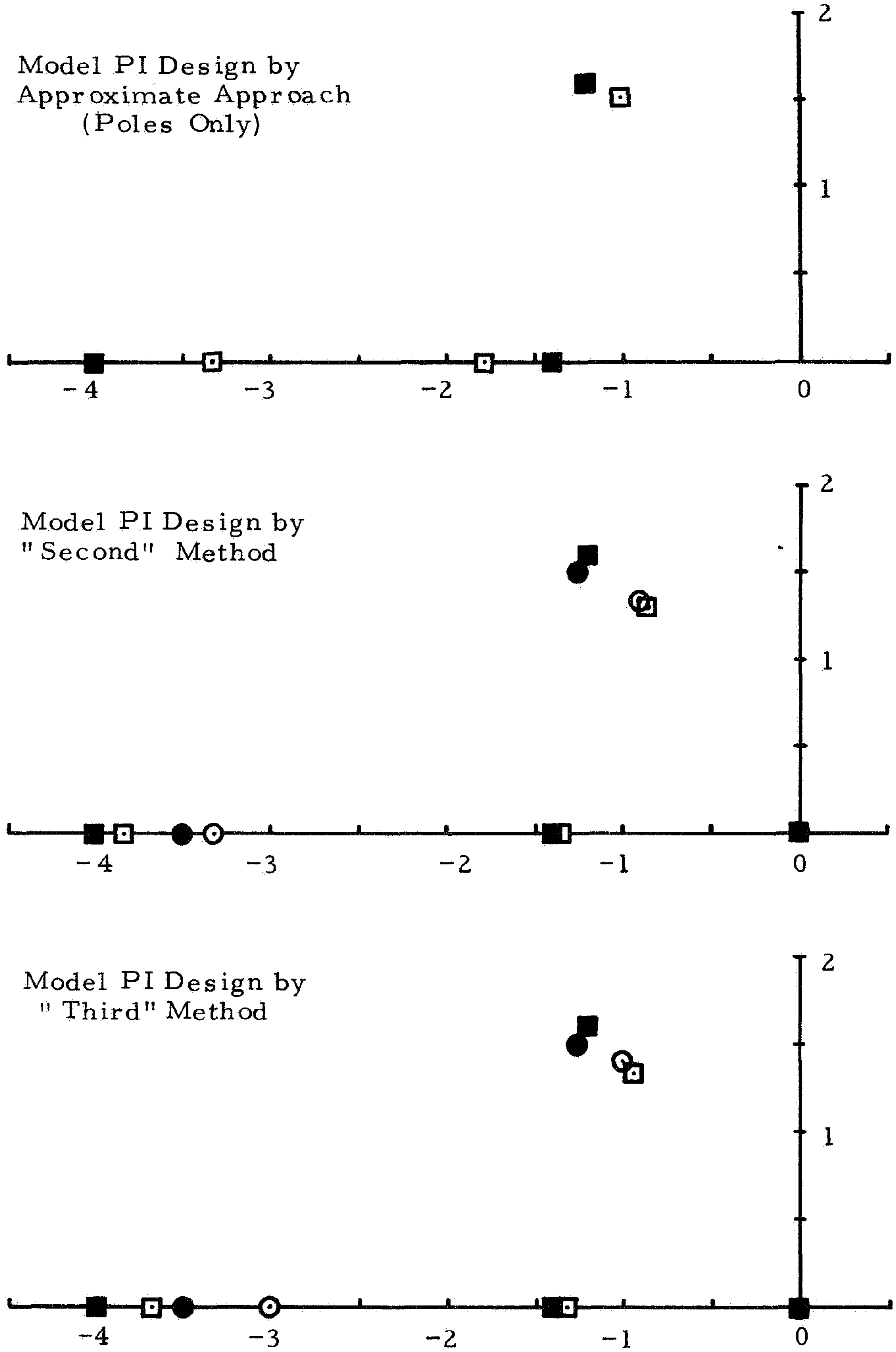
Figure 6-15 Time History Comparison of Model PI Design (Third Method) to the X-15 SAS Off realistic value for the yaw rate gain, than placing a constraint on the magnitude of the yaw rate gain. By using the first design as a guide, one could select a new model with a lower Dutch-roll damping that could quite likely be matched very closely by the Model PI design. This gives the designer more control over the resulting poles and zeros locations than merely placing a constraint on a free parameter.

The sideslip angle feedback gains resulting from the Model PI approach for the model chosen, which are around 4 deg δ_v per deg β, are probably too high. The X-15 SAS yaw servo is authority limited to ±7.5 degrees of vertical stabilizer deflection (51) so that less than 2 degrees of sideslip would saturate the yaw servo with this gain. Using a less conservative model as discussed above would also result in a lower sideslip gain.

This discussion of the practical limitations that might arise in implementing the conceptual designs is included to emphasize the synthesis nature of the Model PI method. In complex problems, one should plan on several design iterations to obtain an acceptable, practical design. This is true with other analytical design methods also. For example, the designs in references 24, 27 and 28 for the same problem considered here, also have feedback gains that would be too high for practical implementation. To obtain practical designs one would have to repeat the procedures used in the particular reference with an appropriate modification. In the linear optimal control approach used in references 27 and 28, the resulting feedback design is much more complicated than necessary, and the designer would want to simplify it, which involves additional synthesis. Although these analytical techniques do not provide an automatic design they are much simpler to use than conventional techniques on complex design problems such as this example. Anyone familiar with conventional techniques can appreciate the tedious effort that would be involved in designing this multivariable system by root locus, Nyqyust or Bode techniques.

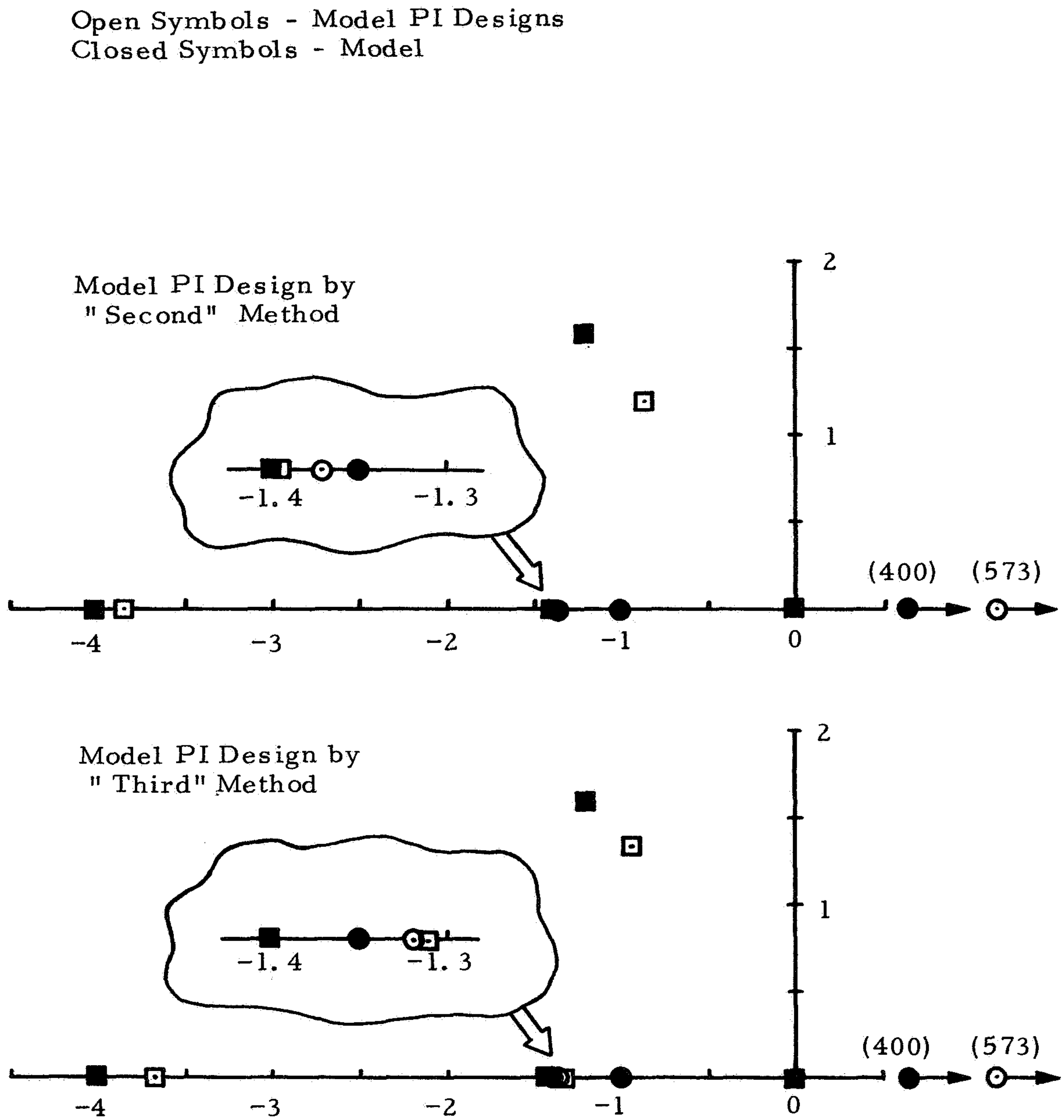
The effectiveness of the Model PI in the three design approaches illustrated is indicated in the following figures that compare the resulting designs to the models used. Figure 6-16 compares the poles and

and zeros of the p/δ_{a_p} and β/δ_{v_p} transfer functions for each design to those of the models (only the poles are compared for the approximate approach, section 6. 2. 2). In each case the Model PI design poles and zeros are close to the respective model poles and zeros. The time responses of the X-15 with the Model PI SAS designs for step δ_{a_p} and δ_{v_p} inputs are compared to the corresponding models' time responses in figures 6-17 to 6-19. Part a in each figure is for a δ_{a_p} step input and part b is for a δ_{v_p} stepinput. The spiral mode is actually unstable in every design but the divergence is so slow (the time constants are over one hour) that it is of no significance. The "steady-state" values of roll rate and sideslip used to normalize the responses in these figures are the values that would result if the spiral mode were neglected. These time responses show that in each case the Model PI produced a design that matches the model's response fairly closely. They also show that in terms of time responses the three designs are not significantly different.



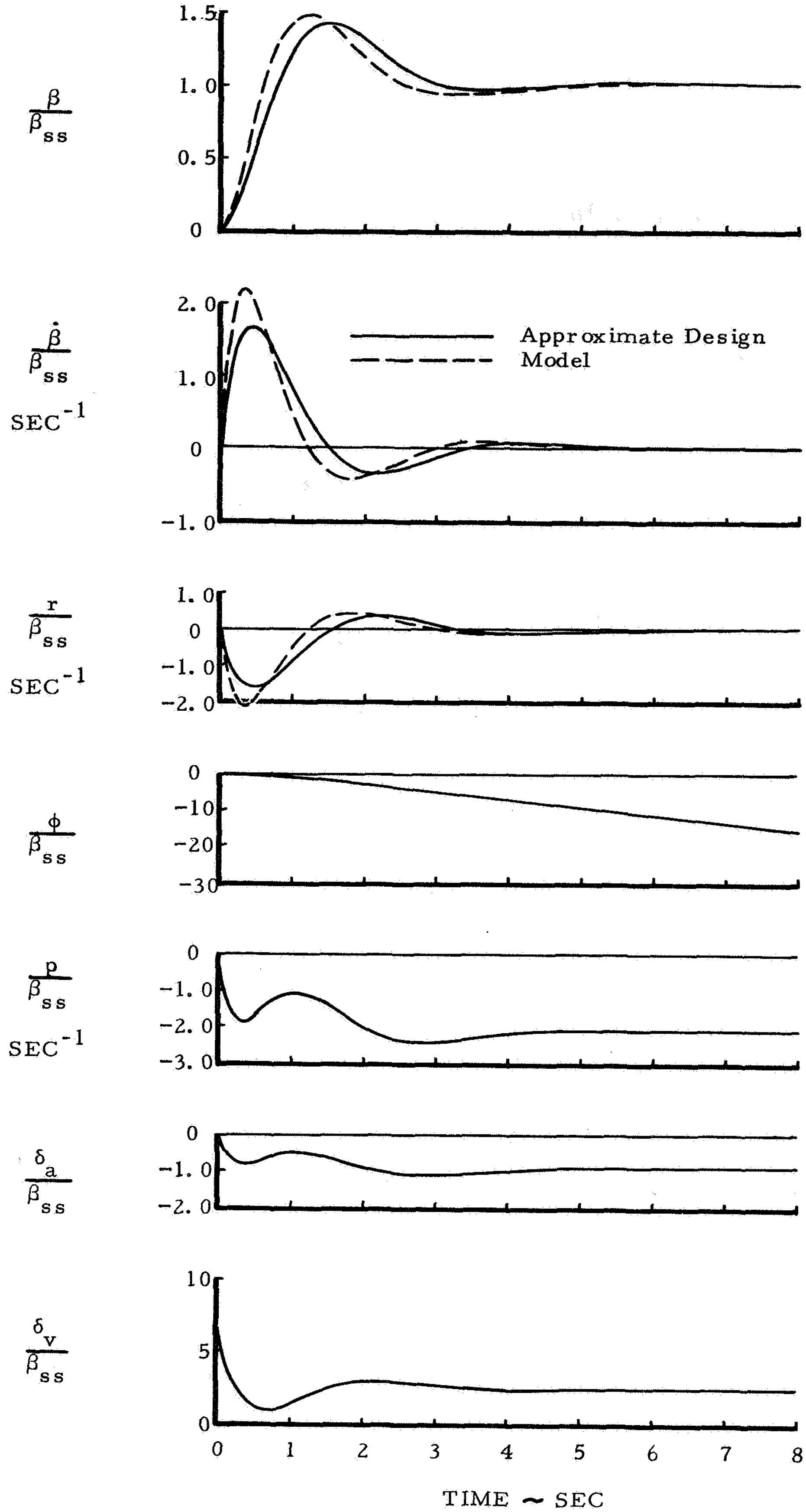
(a) Roll Command to Roll Rate Transfer Function

Figure 6-16 Comparison of the Model and Model PI Design Poles and Zeros for the Three Design Approaches



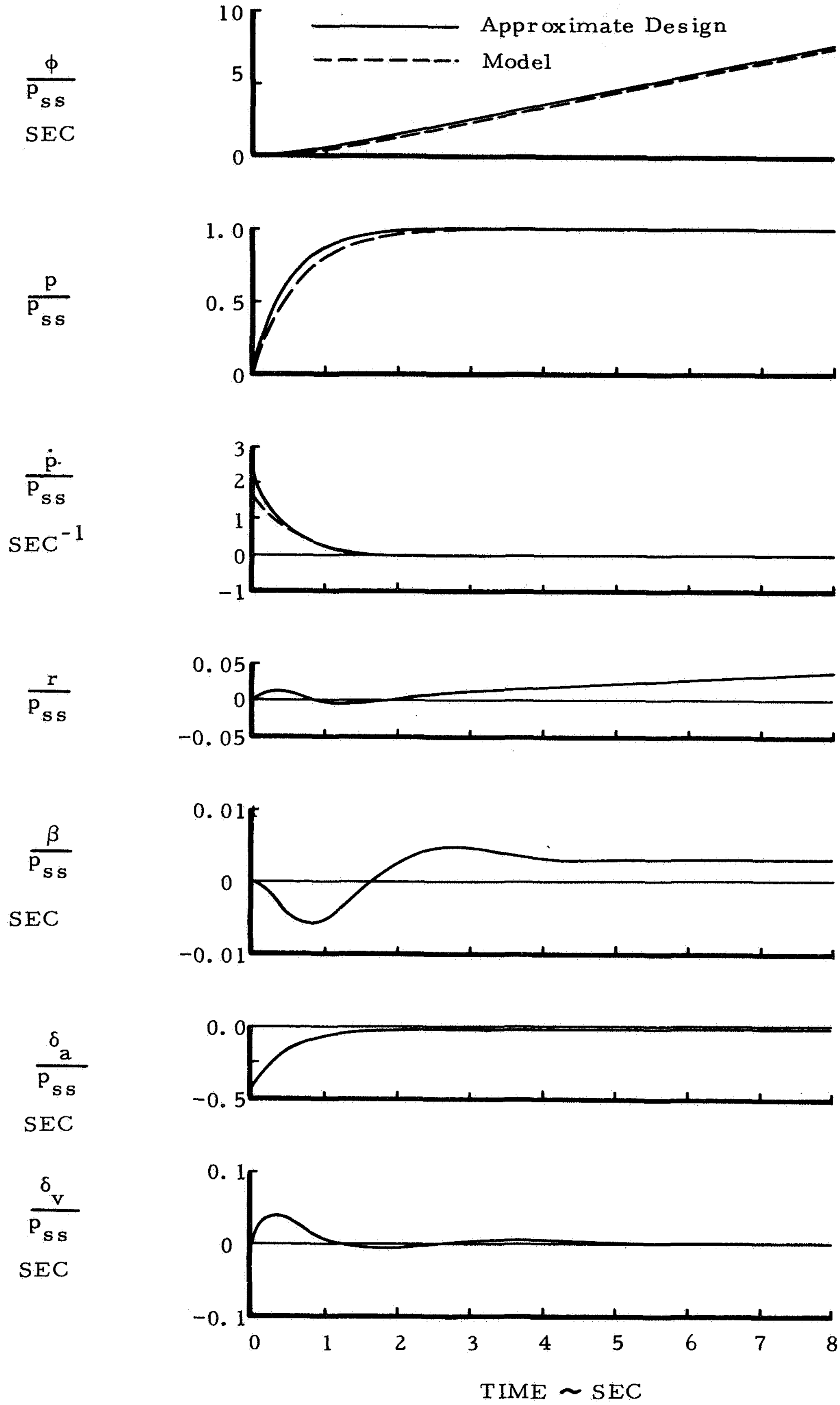
(b) Yaw Command to Sideslip Angle Transfer Function

Figure 6-16 Concluded



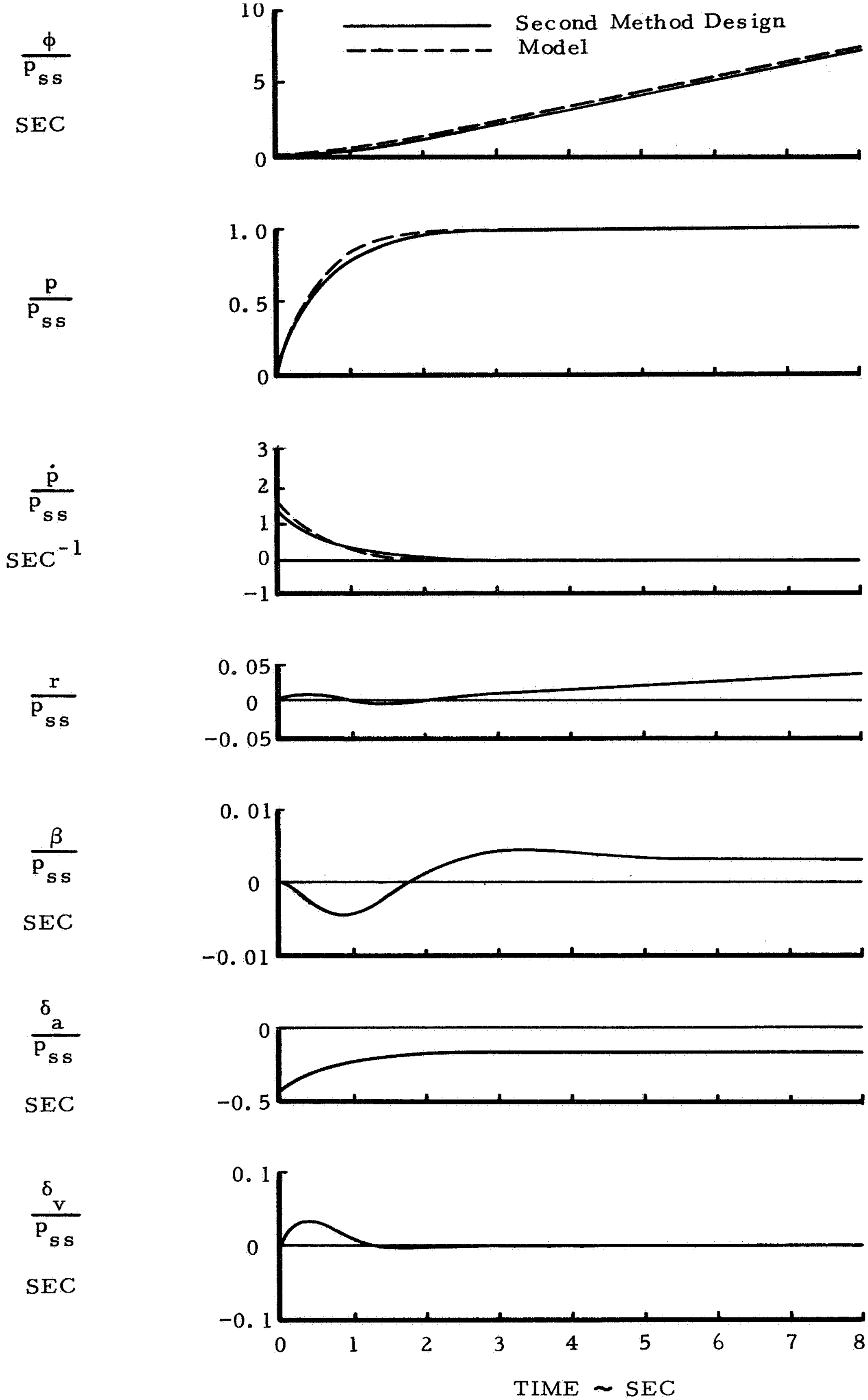
(b) A Step Yaw Command Input

Figure 6-17 Concluded



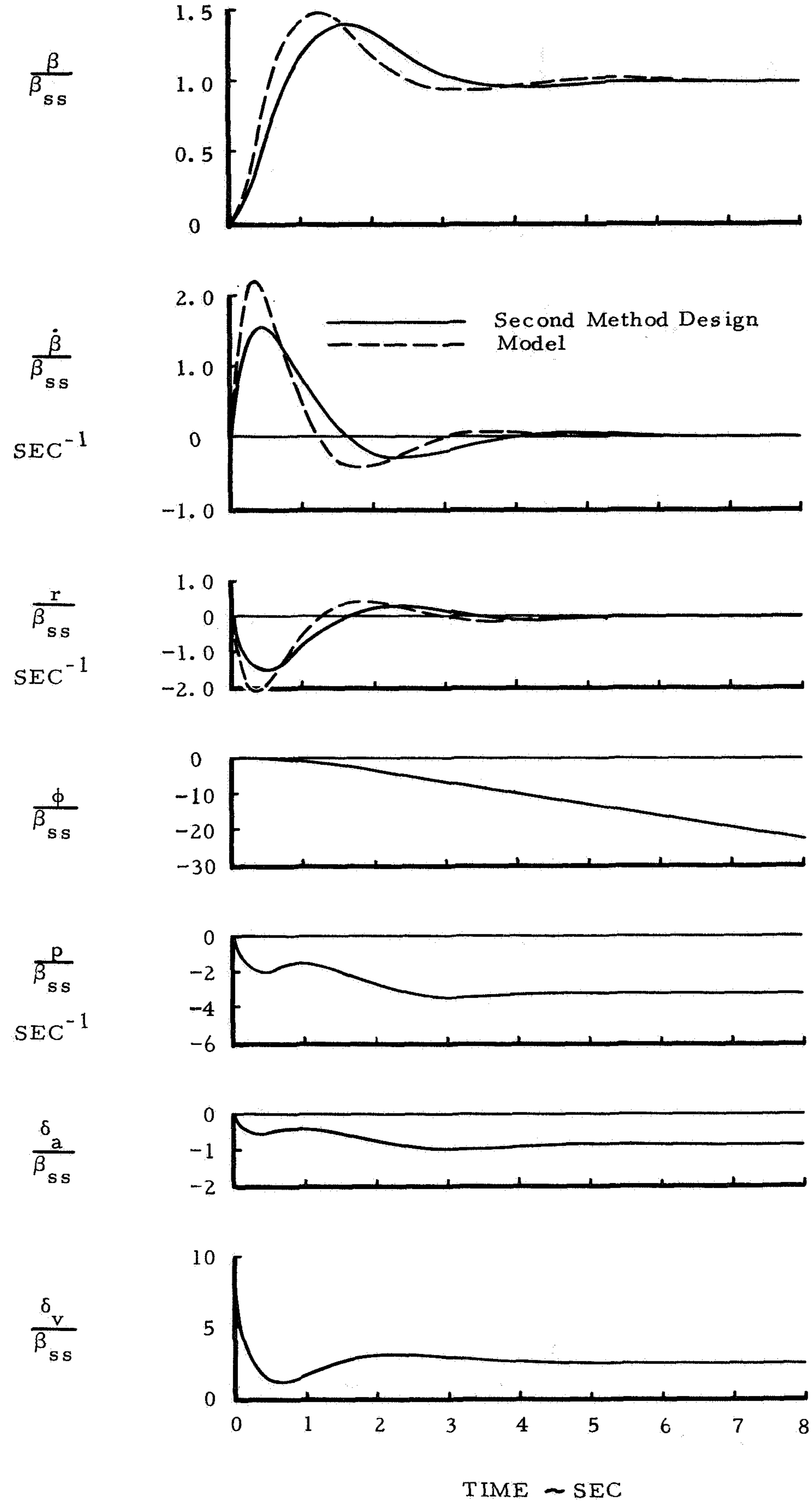
(a) A Step Roll Command Input

Figure 6-17 Time History Comparison of the Model and Model PI Design (Approximate Method)



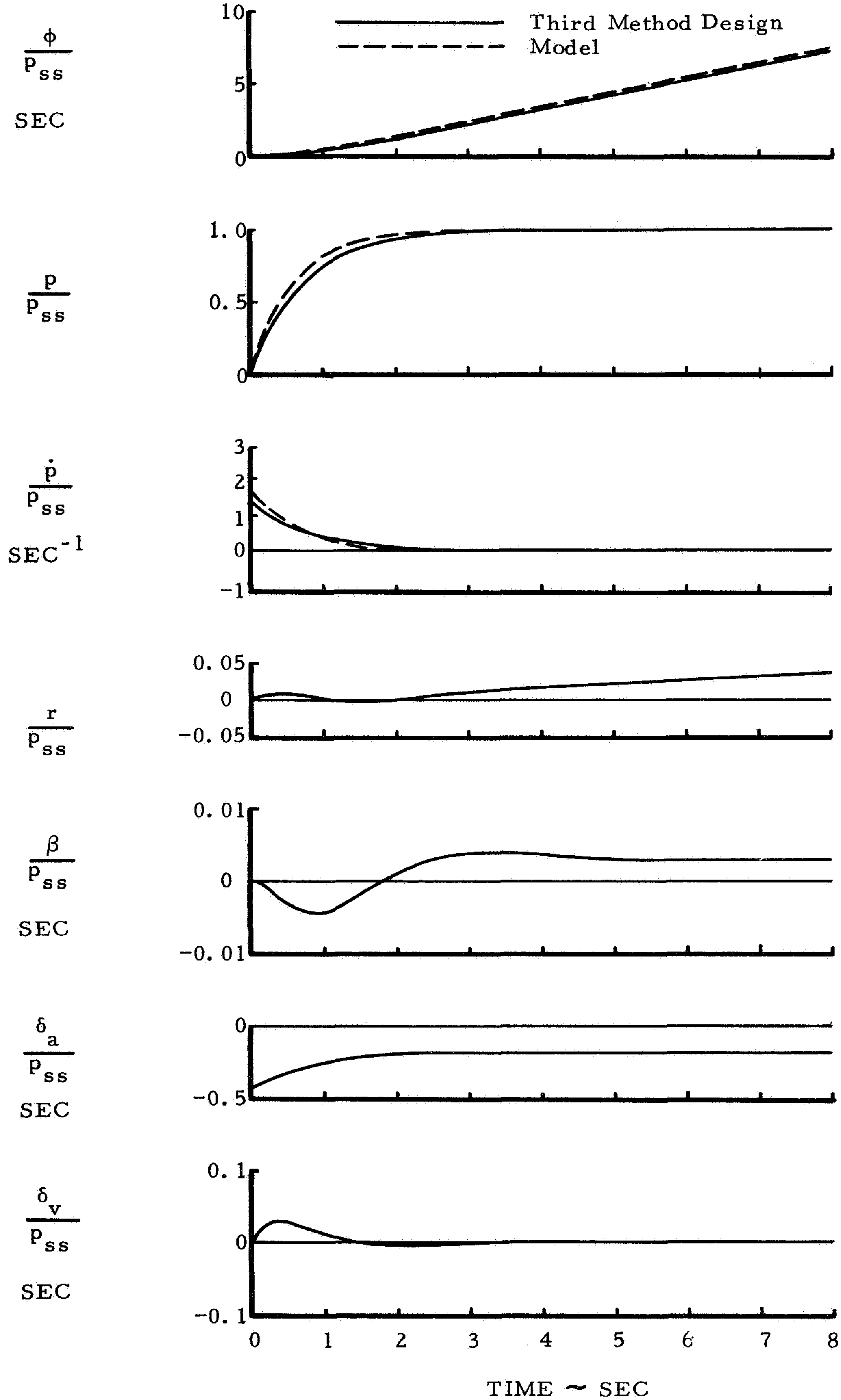
(a) A Step Roll Command Input

Figure 6-18 Time History Comparison of the Model and Model PI Design (Second Method)



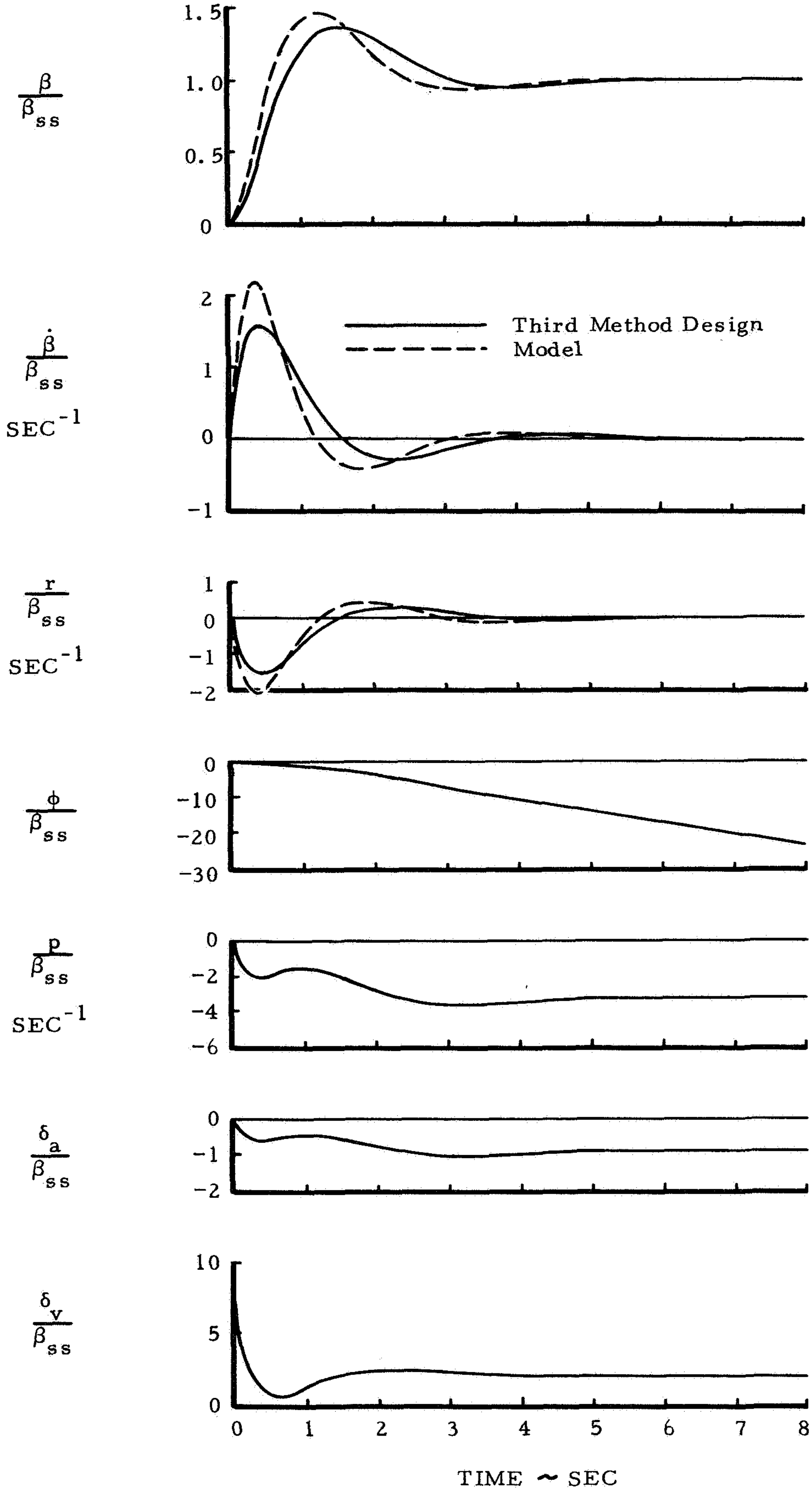
(b) A Step Yaw Command Input

Figure 6-18 Concluded



(a) A Step Roll Command Input

Figure 6-19 Time History Comparison of the Model and Model PI Design (Third Method)



(b) A Step Yaw Command Input

Figure 6-19 Concluded

6.3 A Velocity Command Flight Control System for a VTOL Aircraft

The design of a longitudinal flight control system (FCS) for VTOL aircraft in which the pilot directly commands linear velocity with respect to the ground is presented in reference 52. Having the steady state velocity proportional to the control stick displacement should significantly reduce the pilot control task during hover and low speed flight particularly for IFR operation. "Hands-off" flight, i.e. zero velocity command, automatically produces a hover. A constant glide slope, which is necessary for an IFR landing approach, is established by commanding constant horizontal and vertical velocities. Deviations from the desired glide slope could be easily made by slight adjustments of one control stick position.

This example is a redesign of the velocity command FCS in reference 52 for the hover flight condition. The design specifications used here are somewhat different from those in reference 52 in that the longitudinal handling qualities criteria discussed in Chapter 2, section 2.1.1.3 are included here as part of the specifications. This example is a good illustration of the way engineering specifications may force the designer to compromise between two desirable but conflicting characteristics and of the way such compromises enter into the Model PI design method. As a result of using different specifications than those used in reference 52, one should expect the Model PI design obtained in section 6.3.2 to differ from that in reference 52. In order to compare design techniques, the velocity command FCS is also designed in section 6.3.3 by the Model PI method using specifications comparable to those used in reference 52.

6.3.1 Problem Formulation

A functional block diagram of the system to be considered is shown in figure 6-20. The specific VTOL aircraft considered is a Boeing Vertol CH-46C tandem-rotor helicopter. The feedback control system consists of an inner-loop for pitch attitude stabilization and an outer-loop for obtaining the desired velocity command response. The vehicle attitude and velocity with respect to a local vertical reference

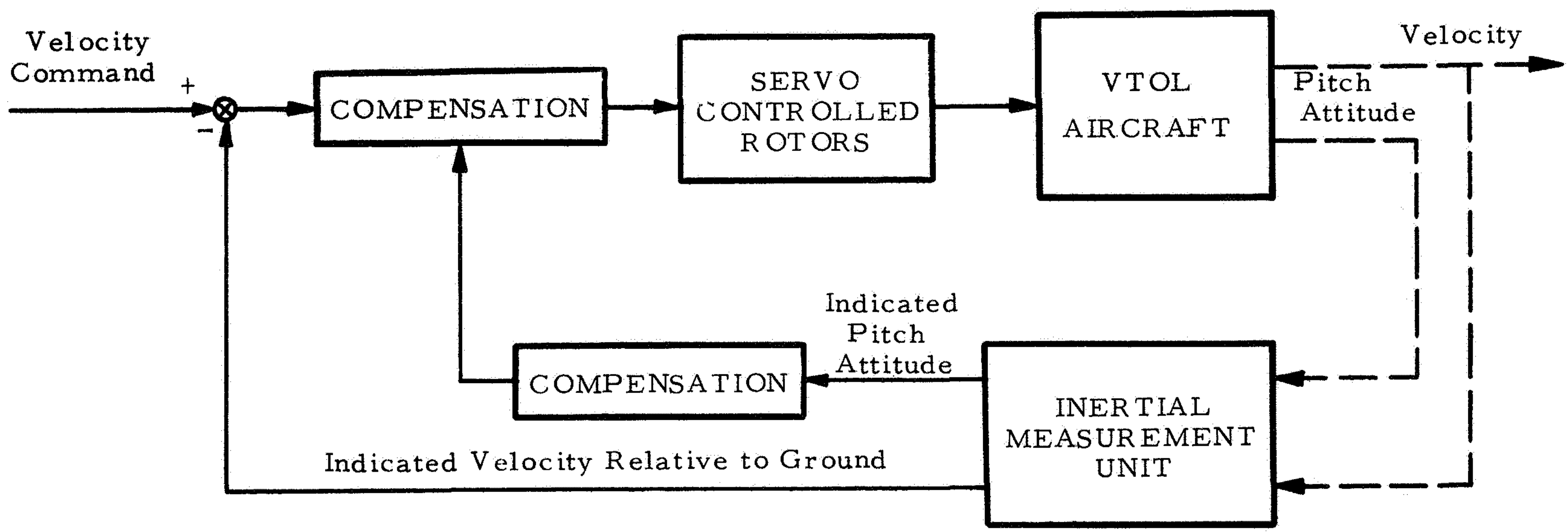


Figure 6-20 Functional Block Diagram of a Velocity Command Flight Control System for a VTOL Aircraft

frame are sensed by an inertial measurement unit. These signals are passed through appropriate compensation networks to a servo that varies the total collective pitch of the two rotors differentially, i.e. increases total collective pitch on one and decreases it on the other, to produce a pitching moment. A forward velocity command input produces a nose down pitching moment tilting the helicopter, and hence the thrust vector, forward. The forward component of thrust accelerates the helicopter to the desired velocity. By the time the desired velocity is reached, the pitch attitude must reduce in magnitude to the new trim value.

One cannot expect to design an extremely fast velocity command response system because it would require rapidly pitching over to a large negative attitude to give the large forward component of thrust that would be necessary to produce a fast velocity response. The short term response would appear to the pilot as essentially an attitude control system, so that one must restrict the pitch control sensitivity and damping to values that give at least Acceptable handling qualities as defined by criteria such as those shown in figure 2-4 and repeated here in figure 6-21. The abscissa, control power/inertia, is equivalent to the pitching acceleration per inch of control stick deflection and is interpreted by the pilot as the pitch control sensitivity. Since the control stick deflection is proportional to the velocity command signal, the abscissa can also be written as pitching acceleration per unit velocity command as indicated by the second scale in figure 6-21. This assumes a scale constant of 14 feet per second of velocity command per inch of control stick deflection, which is the value used in reference 52 at the hover condition. The design specifications used in this example required that the pitch acceleration per unit velocity command and the damping/inertia parameters lie within the Acceptable handling qualities boundary in figure 6-21 and, if possible, within the Desirable boundary.

The assumed specifications for the time response to a step velocity command input are given in figure 6-22. Two velocity response envelopes are shown, corresponding to what are assumed to

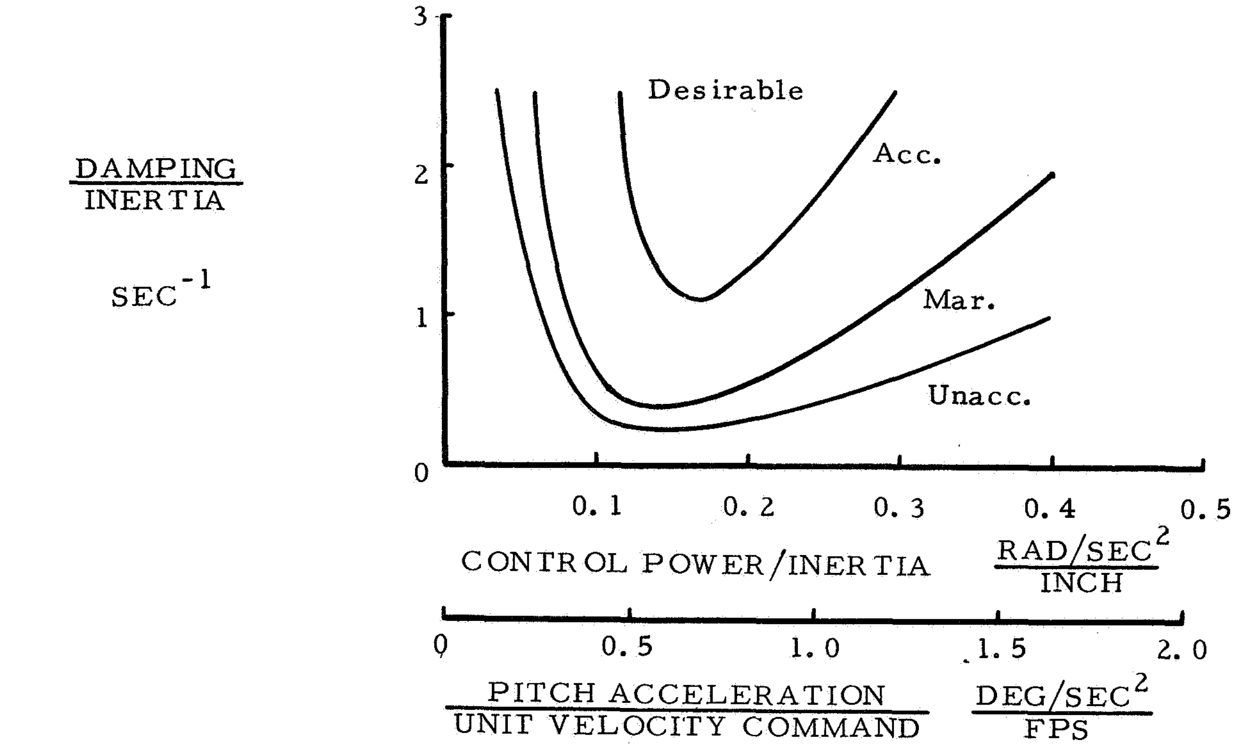


Figure 6-21 Longitudinal Handling Qualities Criteria for VTOL Aircraft (Reference 35)

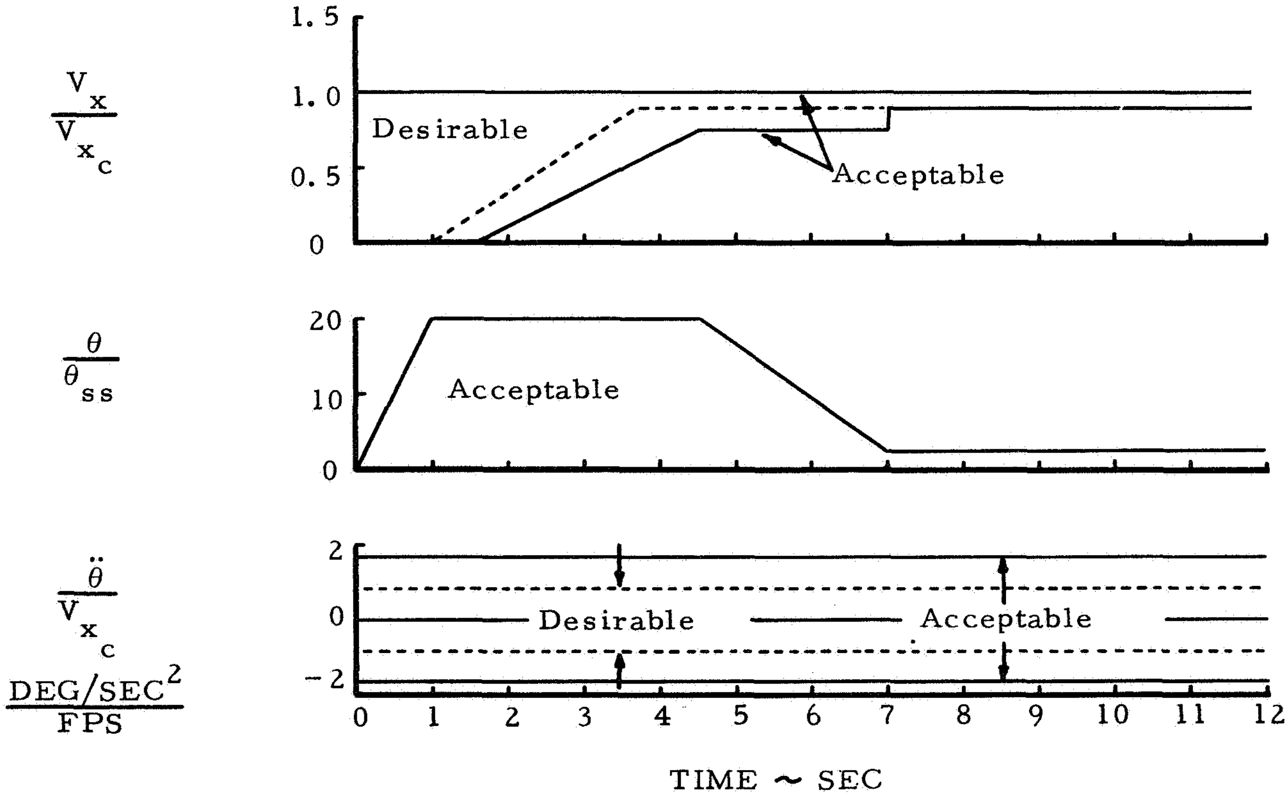


Figure 6-22 Step Response Specification for the Velocity Command FCS Design Example

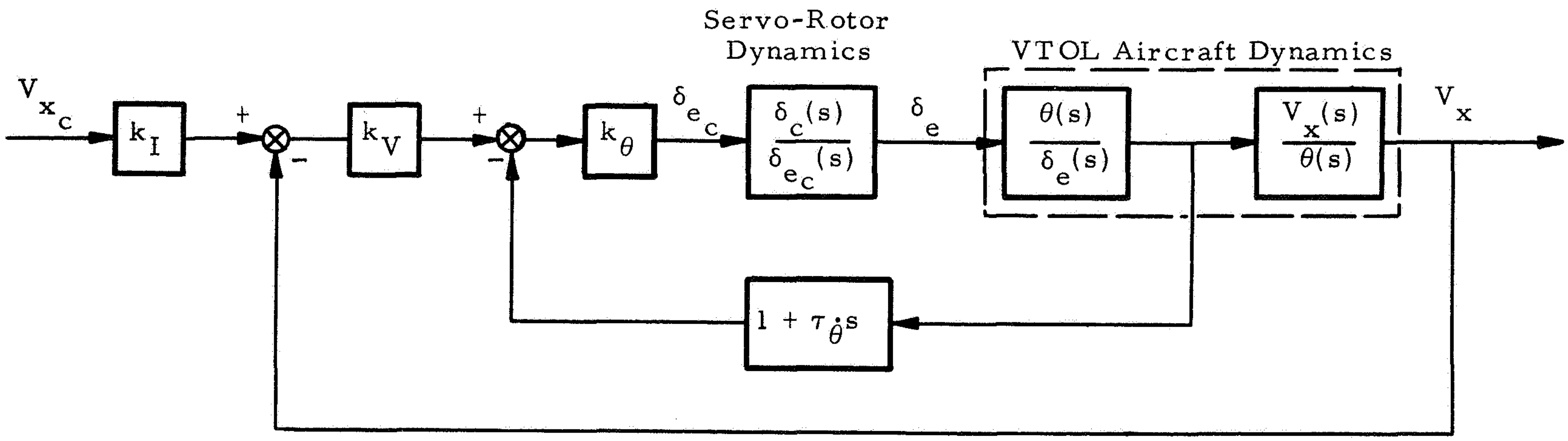
to be Desirable and Acceptable ranges of step responses. The closed-loop step response must lie within the Acceptable envelope and preferably within the Desired envelope. The specification on the $\ddot{\theta}/V_{x_c}$ response represents the requirements of figure 6-21. The Desirable and Acceptable levels of $\ddot{\theta}/V_{x_c}$ indicated on figure 6-22 correspond to values of 0.82 and 1.64 deg/sec^2 per fps, which one can conclude to be Desirable and maximum Acceptable values respectively from figure 6-21. An Acceptable response envelope is also shown for the pitch attitude normalized by the steady-state value. This is essentially a limit on the maximum attitude change allowed for a step command in velocity. It is necessary to pitch over to an attitude many times the steady state (new trim) value to obtain a sufficiently large forward component of thrust, but there is a limit on how far the helicopter should pitch over. The limit shown in figure 6-22 on θ/θ_{ss} was chosen somewhat arbitrarily in this example for lack of the specific information for this vehicle, but it is sufficient to illustrate the point.

To reiterate, the design specifications for this example are given by the Acceptable boundaries in figure 6-21 and 6-22. The Desirable boundaries define preferred goals within the Acceptable regions.

A possible feedback configuration for a velocity command FCS is shown in figure 6-23. It consists of a first order lead in the pitch attitude feedback path and a unity velocity feedback. The free design parameters are the pitch attitude gain, k_θ, the first order lead time constant, $\tau_{\dot{\theta}}$, and the velocity gain, k_v. An input gain, k_I, is included to adjust the closed-loop static sensitivity independent of the loop gain, if necessary. Since the static sensitivity also affects the handling qualities, the input gain will be restricted to a 15% change from unity. The necessary open-loop transfer functions for the VTOL considered at the hover flight condition* are

---

* These transfer functions are from (Appendix A of reference 52. A pole and a zero of $\theta(s)/\delta_c(s)$ that nearly cancel are omitted from (6-43).



$$\frac{\theta(s)}{\delta_e(s)} = \frac{2.543\left(1 + \frac{s}{0.02056}\right)}{\left(1 + \frac{s}{0.8669}\right)\left[1 - \frac{2(0.226)}{(0.429)}s + \left(\frac{s}{0.429}\right)^2\right]}, \quad \frac{\text{DEG}}{\text{INCH}}$$

$$\frac{V_x(s)}{\theta(s)} = \frac{-26.773\left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{\left(1 + \frac{s}{0.02056}\right)}, \quad \frac{\text{FPS}}{\text{DEG}} \qquad \frac{\delta_e(s)}{\delta_{e_c}(s)} = \frac{1}{\left(1 + \frac{s}{15.2}\right)\left(1 + \frac{s}{14.3}\right)}$$

Figure 6-23 Block Diagram of a Velocity Command FCS Feedback Configuration

Servo-Rotor Dynamics:

$$\frac{\delta_e(s)}{\delta_{e_c}(s)} = \frac{1}{\left(1 + \frac{s}{15.2}\right)\left(1 + \frac{s}{14.3}\right)} \qquad (6\text{-}42)$$

Aircraft Dynamics:

$$\frac{\theta(s)}{\delta_e(s)} = \frac{2.543 \left(1 + \frac{s}{0.02056}\right)}{\left(1 + \frac{s}{0.8669}\right)\left[1 - \frac{2(0.226)}{(0.429)} s + \left(\frac{s}{0.429}\right)^2\right]} \qquad (6\text{-}43)$$

$$\frac{V_x(s)}{\theta(s)} = \frac{-26.773 \left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{\left(1 + \frac{s}{0.02056}\right)} \qquad (6\text{-}44)$$

where

δ_e is the differential total collective pitch of the two rotors, measured in terms of inches of control stick deflection.

δ_{e_c} is the servo command signal for δ_e, in terms of inches of control stick deflection.

θ is the pitch attitude, in degrees.

V_x is the longitudinal velocity with respect to the ground, in feet per second (fps).

Since the poles of the servo-rotor dynamics are at least an order of magnitude away from the poles of aircraft dynamics, the servo-rotor dynamics will be neglected during the parameter optimization process. Once a design is obtained it will be checked with the servo-rotor dynamics included. The approximate closed-loop transfer function of V_{x_c} to V_x, neglecting the servo-rotor dynamics, is

$$\frac{V_x(s)}{V_{x_c}(s)} = \frac{b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0} \qquad (6\text{-}45)$$

where

$$a_2 = 0.6725 + 19.70 k_\theta \tau_{\dot\theta} + 0.1970 k_v k_\theta$$

$$a_1 = 0.0152 + 19.70 k_\theta + 0.4050 k_\theta \tau_{\dot\theta} + 0.4809 k_v k_\theta$$

$$a_0 = 0.1593 + 0.4050 k_\theta - 10.844 k_v k_\theta \qquad (6\text{-}46)$$

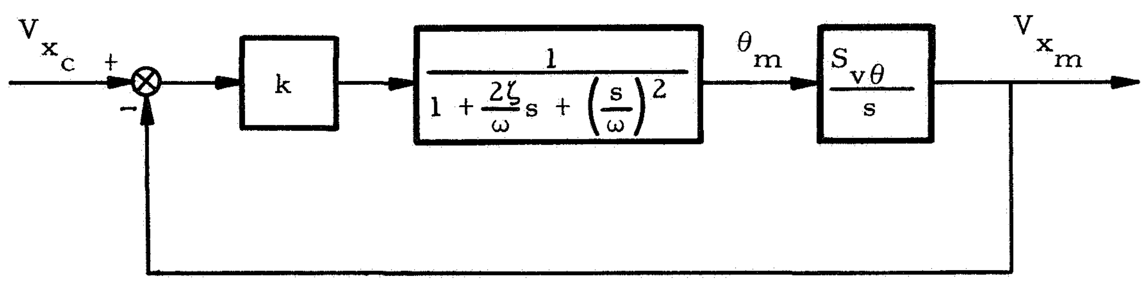
and

$$b_2 = (0.1970 k_v k_\theta)\, k_I$$

$$b_1 = (0.4809 k_v k_\theta)\, k_I$$

$$b_0 = (-10.844 k_v k_\theta)\, k_I \qquad (6\text{-}47)$$

In selecting a model, it is very helpful to consider a rough approximation to the velocity command FCS shown in figure 6-24. From the short term, handling qualities standpoint one can neglect the velocity feedback and consider only the V_{x_c} to θ_m transfer characteristics,



$$S_{v\theta} = -0.55 \quad \text{(Ref. 52)}$$

Figure 6-24 Block Diagram of a Model for the Velocity Command FCS

$$\frac{\theta_m(s)}{V_{x_c}(s)} = \frac{k\omega^2}{s^2 + 2\zeta\omega s + \omega^2} \tag{6-48}$$

The damping/inertia parameter in figure 6-28 corresponds to $2\zeta\omega$ in equation (6-48). The maximum magnitude of $\ddot{\theta}_m$ per unit V_{x_c} for a step V_{x_c} input is the pitch acceleration per unit velocity command parameter in figure 6-21. This maximum magnitude occurs at the initial value of $\ddot{\theta}_m$ which can be obtained by applying the initial value theorem to $\ddot{\theta}_m(s)/V_{x_c}(s)$, i. e.

$$\left|\frac{\ddot{\theta}_m}{V_{x_c}}\right|_{Max} = \left|\lim_{s\to\infty}\left[\frac{s^3 k\omega^2}{s(s^2 + 2\zeta\omega s + \omega^2)}\right]\right|$$

$$= |k\omega^2| \qquad \frac{deg/sec^2}{fps} \tag{6-49}$$

One can select a combination of $2\zeta\omega$ and $k\omega^2$ that would correspond to Desirable or at least Acceptable handling qualities from figure 6-28. The velocity step response can be estimated from the closed-loop transfer function

$$\frac{V_{x_m}(s)}{V_{x_c}(s)} = \frac{k\omega^2(-0.55)}{s^3 + 2\zeta\omega s^2 + \omega^2 s + k\omega^2(-0.55)} \tag{6-50}$$

and the non-dimensionalized step response charts in Clark (49). Once $2\zeta\omega$ and $k\omega^2$ are selected only ω^2 remains unspecified in (6-50), and its is chosen to produce the proper velocity step response.

Figure 6-25 shows the sequence of handling qualities characteristics considered in selecting a model. The combinations of parameters labeled 1 and 2 were tried first in an attempt to obtain Desirable handling qualities. However it is impossible to obtain even an Acceptable velocity step response, i. e. satisfying the criteria in

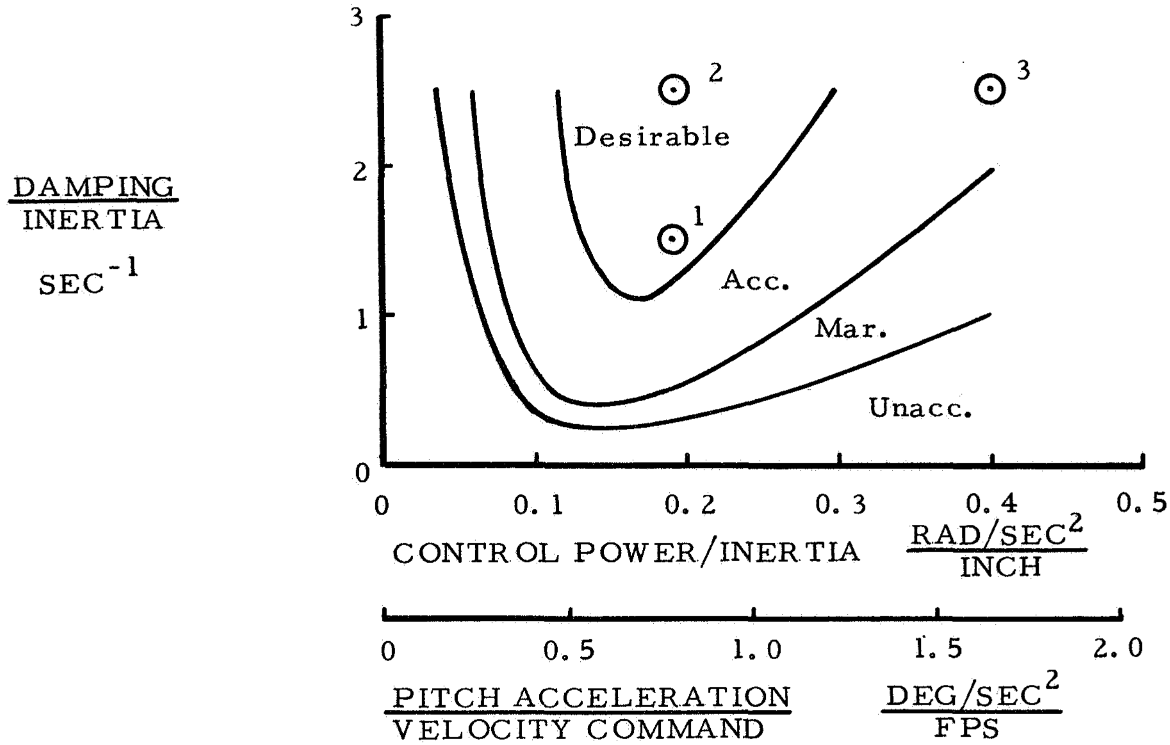


Figure 6-25 Longitudinal Handling Qualities Selected for the Model

figure 6-22, for any value of ω^2 with combinations 1 or 2. The velocity response is either too lightly damped or much too slow. In order to speed up the response and yet maintain adequate damping, the combination labeled 3 was tried. It lies within the Acceptable handling qualities region of figure 6-25 at the upper limit of the Acceptable range of the pitch acceleration per unit velocity command parameter. Using the number 3 combination ($2\zeta\omega = 2.5$ sec^{-1} and $|k\omega^2| = 1.64$ deg/sec^2 per fps) and a frequency of $\omega = 1.6$ rad/sec, results in an approximate model having an Acceptable velocity step response. This establishes the dominant model transfer characteristics as

$$\frac{V_{x_m}(s)}{V_{x_c}(s)} = \frac{1}{\left(1 + \frac{s}{0.7}\right)\left[1 + \frac{2(0.79)}{(1.14)} s + \left(\frac{s}{1.14}\right)^2\right]} \qquad (6\text{-}51)$$

According to the guide for selecting the model structure when using the Model PI, table 3-5 in Chapter 3, the best model structure for this example is one with three poles and two zeros, i.e. the same structure as the closed-loop system transfer function, equation (6-45). One can see from the block diagram in figure 6-23 that the system's closed-loop zeros are just the zeros of the aircraft dynamics. These can also be used as zeros for the model; therefore the model selected to represent the design specifications in the Model PI is

$$\frac{V_{x_m}(s)}{V_{x_c}(s)} = \frac{\left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{\left(1 + \frac{s}{0.7}\right)\left[1 + \frac{2(0.79)}{(1.14)} s + \left(\frac{s}{1.14}\right)^2\right]} \qquad (6\text{-}52)$$

The step response for this model is compared to the step response specifications in figure 6-26, which shows that it lies within the Acceptable region but not within the Desirable region. To speed up the model's velocity response sufficient to lie within the Desirable region would require a larger negative pitch acceleration. But then the pitch acceleration per unit velocity command would exceed the Acceptable limit and the handling qualities would not be Acceptable. The model (6-52) represents a satisfactory compromise between obtaining Desirable handling qualities but Unacceptable step response and Desirable step response but Marginal to Unacceptable handling qualities.

The model's extended coefficient vector and pseudo IC vector corresponding to (6-52) are, respectively,

$$\underline{\tilde{\alpha}}' = [\ 0.903 \quad 2.56 \quad 2.50 \quad 1.0\ ] \qquad (6\text{-}53)$$

$$\tilde{x}'_{m_0} = [\ \underline{x}'_{m_0} \quad -\underline{x}'_{m_0}\underline{\alpha}\ ]$$

$$\underline{x}'_{m_0} = [\ -1.0 \quad -.0165 \quad .0097\ ] \qquad (6\text{-}54)$$

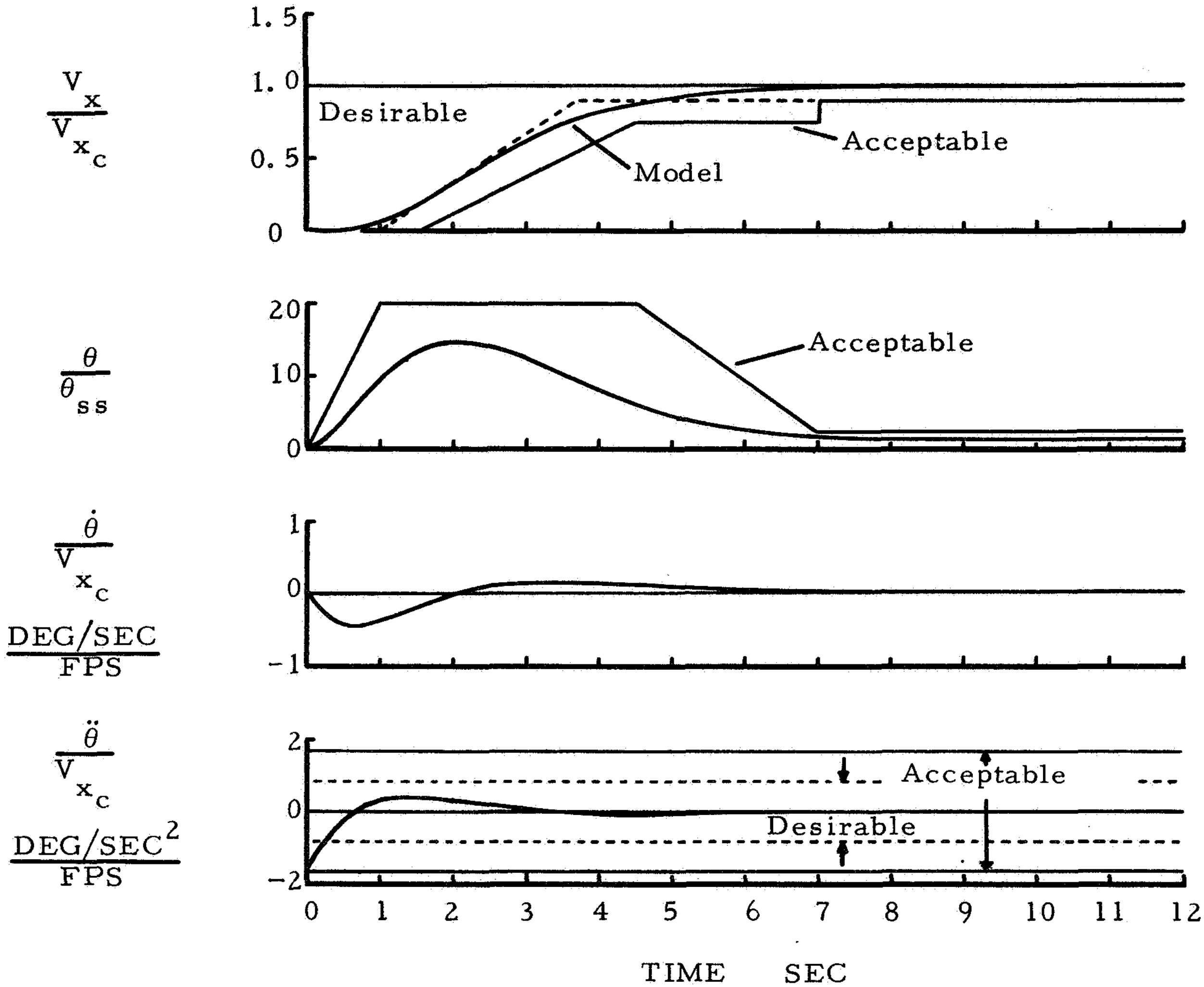


Figure 6-26 Step Response of the Model Used to Represent the Design Specifications

6.3.2 Design by the Model PI Method

The problem is now formulated for applying the general computer program for the Model PI design method. The subroutine COEF is written to generate the coefficients (6-46) and (6-47). The input gain k_I is not considered to be a free parameter during the parameter optimization process is assumed to be unity. It can be adjusted afterwards, if necessary.

Using a pseudo IC weighting factor of 7.0×10^{-2} in the Model PI and an initial choice for the free parameters* of

$$k_\theta = 0.1 \text{ inch/deg}$$

$$\tau_{\dot{\theta}} = 1.0 \text{ sec}^{-1}$$

$$k_v = -0.5 \text{ deg/fps}$$

resulted in a final value of

$$k_\theta = 0.156 \text{ inch/deg}$$

$$\tau_{\dot{\theta}} = 0.826 \text{ sec}^{-1}$$

$$k_v = -0.551 \text{ deg/fps}$$

Using these values in equations (6-45) - (6-47) gives an approximate closed-loop transfer function of

$$\frac{V_x(s)}{V_{x_c}(s)} = \frac{k_I(0.807)\left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{\left(1 + \frac{s}{1.857}\right)\left[1 + \frac{2(0.85)}{(0.787)} s + \left(\frac{s}{0.787}\right)^2\right]} \tag{6-55}$$

The step response corresponding to (6-55) is compared to that of the model and the Acceptable response envelope in figure 6-27. The velocity response is presented for $k_I = 1.0$ and $k_I = 1.15$. One can see that

---

* This initial choice of k_v is approximately the value of k used in (6-50) to establish the dominant model characteristics.

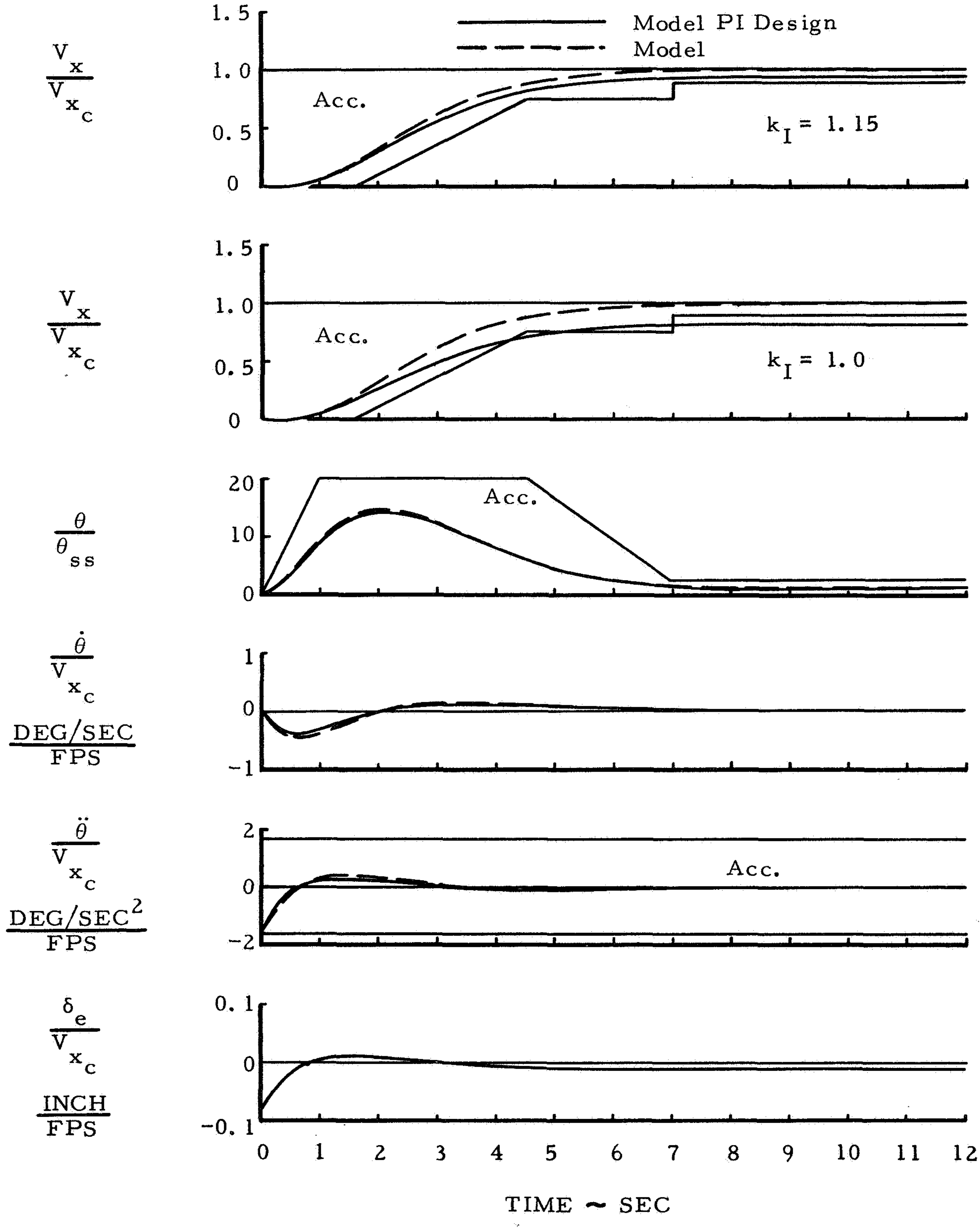


Figure 6-27 Step Response Comparison of the Model and the Model PI Design for a Third Order Approximation to the Velocity Command FCS

the step response is Acceptable if an input gain of $k_I = 1.15$ is used. The estimated handling qualities for this design can be checked by considering the approximate transfer function of V_{x_c} to θ neglecting the outer-loop feedback, i.e.

$$\frac{\theta(s)}{V_{x_c}(s)} = \frac{k_I(1.687)(s + 0.02056)}{(s + 0.0769)\left[s^2 + 2(0.92)(1.70)s + (1.70)^2\right]} \tag{6-56}$$

The damping/inertia parameter for the dominant second order mode of (6-56) is $2\zeta\omega = 3.12$ and the maximum $\ddot{\theta}$ per unit V_{x_c} for a step V_{x_c} input is given by

$$\left|\frac{\ddot{\theta}}{V_{x_c}}\right|_{Max} = \left|\lim_{s\to\infty}\left[\frac{\ddot{\theta}(s)}{V_{x_c}(s)}\right]\right| = 1.687k_I \ \frac{deg/sec^2}{fps} \tag{6-57}$$

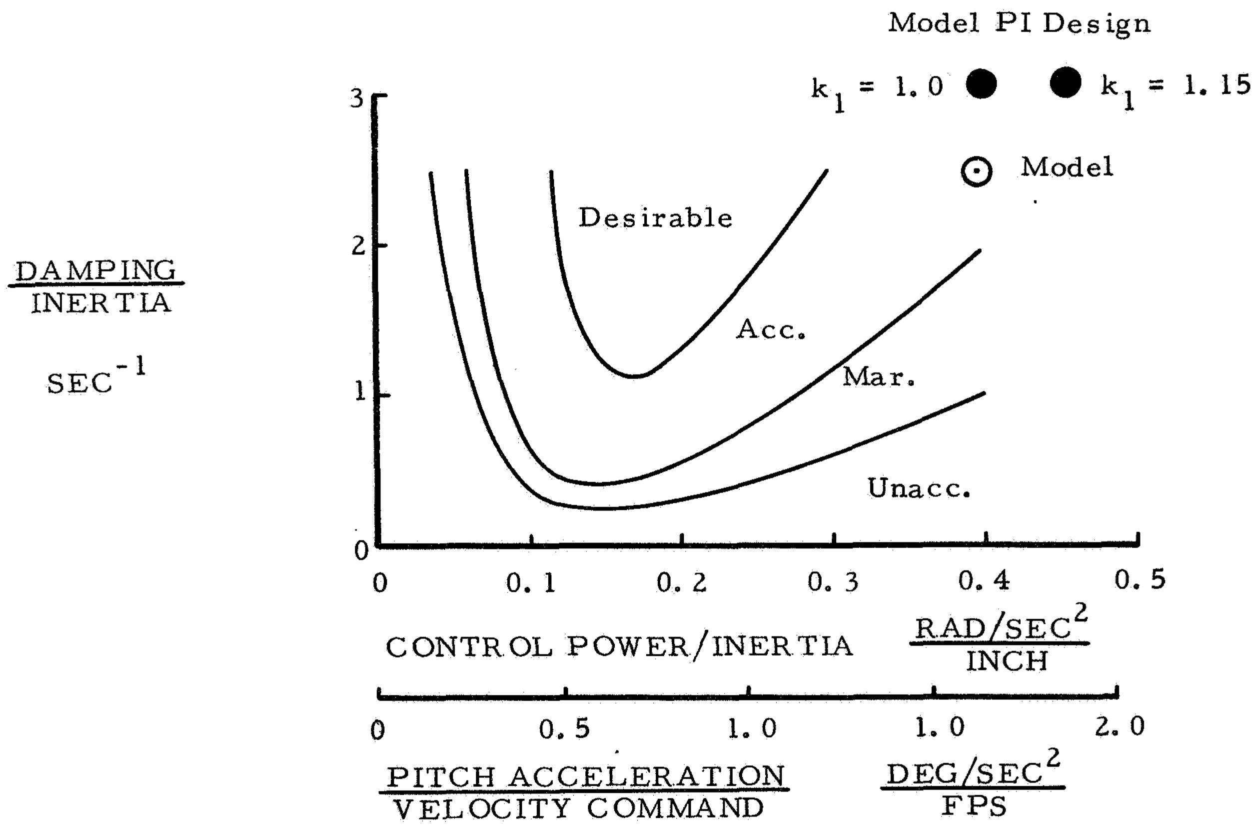


Figure 6-28 Estimated Handling Qualities of the Model PI Design

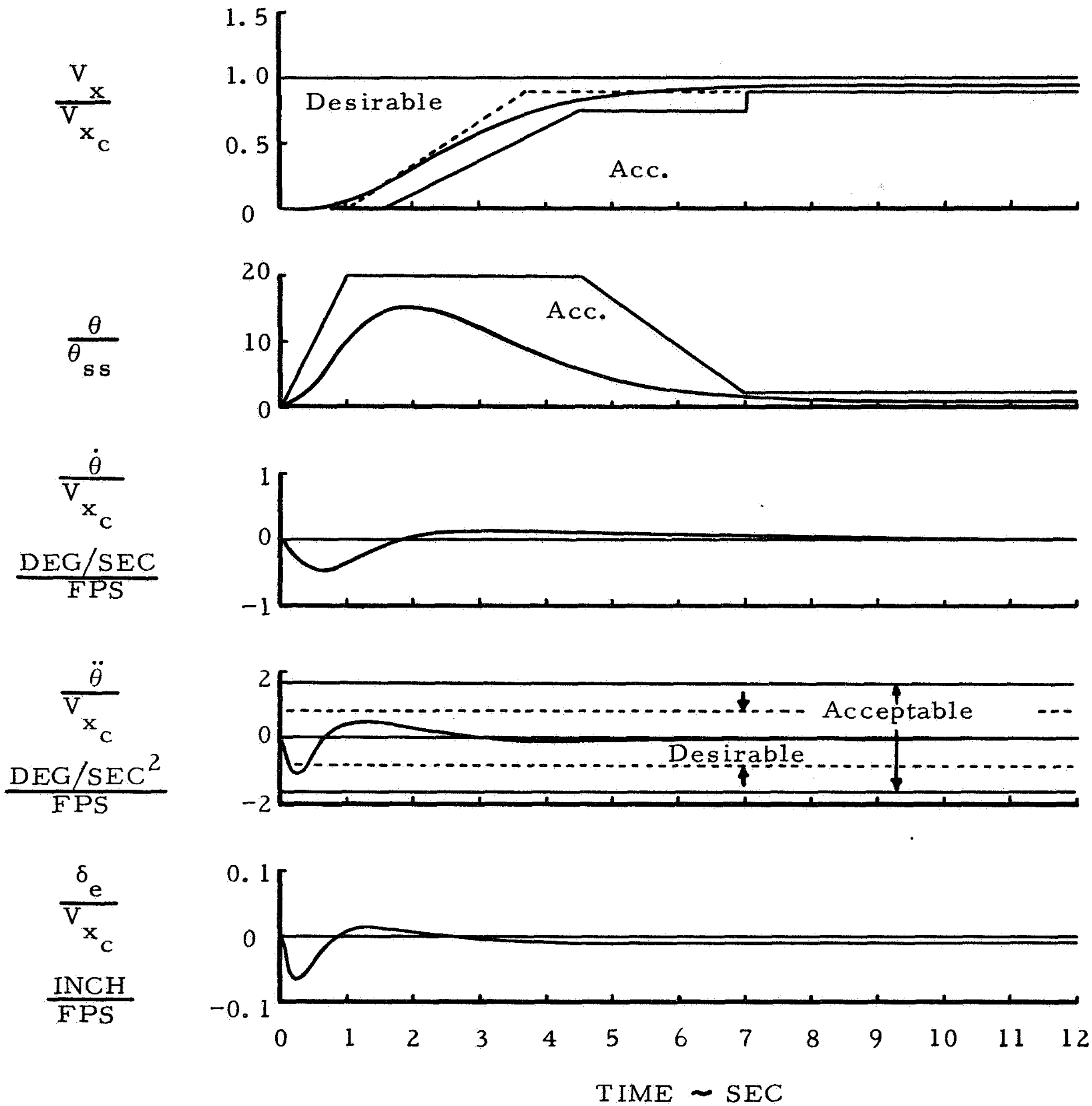


Figure 6-29 Step Response of a Velocity Command FCS Designed by the Model PI Method

Figure 6-28 shows the estimated handling qualities in terms of these parameters for input gains of $k_I = 1.0$ and $k_I = 1.15$, which indicates that this design has characteristics slightly beyond the range tested in reference 35 for establishing these handling qualities criteria. However, if the range were extended, it appears that the Model PI design would lie in the Acceptable handling qualities region.

As a final check of this design ($k_I = 1.15$) figure 6-29 presents the step response of the closed-loop system with the servo-rotor dynamics included. The Acceptable step response specifications are satisfied. In addition, the lag due to the servo-rotor dynamics has reduced the maximum magnitude of θ/V_{x_c} to about 1.2 deg/sec^2 per fps which is nearer the Desirable handling qualities region in figure 6-28 than indicated by the third order approximation. This design appears to be a good compromise between the requirements of a fast velocity response and Acceptable short term handling qualities.

6.3.3 Comparison with Design in Reference 52

The object of this section is to show that if a result similar to that obtained in reference 52 is desired, the Model PI method can provide a comparable design. The design specifications used in reference 52 for the response to a command input are essentially that any oscillatory mode should have a damping ratio of 0.5 or greater and that the velocity response time should be less than 4.0 seconds. These specifications can be represented by a model with a V_{x_c} to V_{x_m} transfer function of

$$\frac{V_{x_m}(s)}{V_{x_c}(s)} = \frac{\left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{(1+s)\left[1 + \frac{2(0.7)}{(2.0)}s + \left(\frac{s}{2.0}\right)^2\right]} \qquad (6\text{-}58)$$

The step response of this model, presented in figure 6-30, shows that the velocity reached 95% of its final value in less than 4.0 seconds. The extended coefficient vector and pseudo IC vector corresponding to (6-58) are

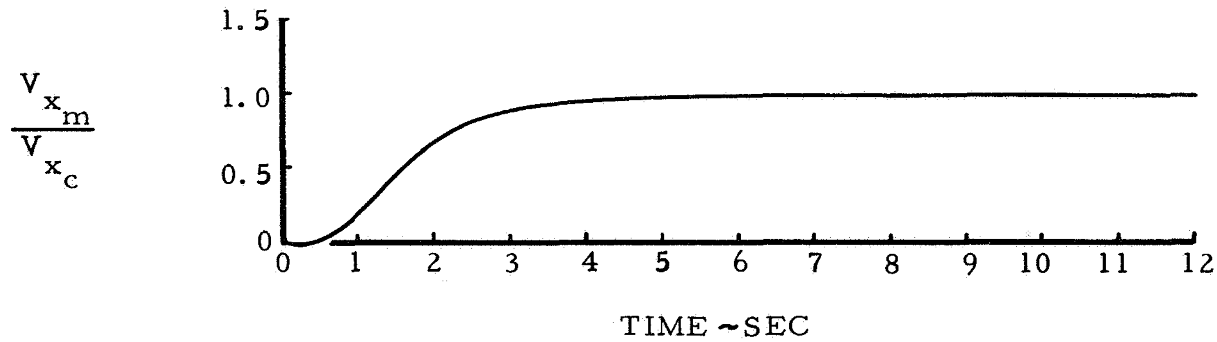


Figure 6-30 Velocity Step Response of a Model Representing the Specifications of Reference 52

$$\underline{\tilde{\alpha}}' = [4.0 \quad 6.8 \quad 3.8 \quad 1.0]$$

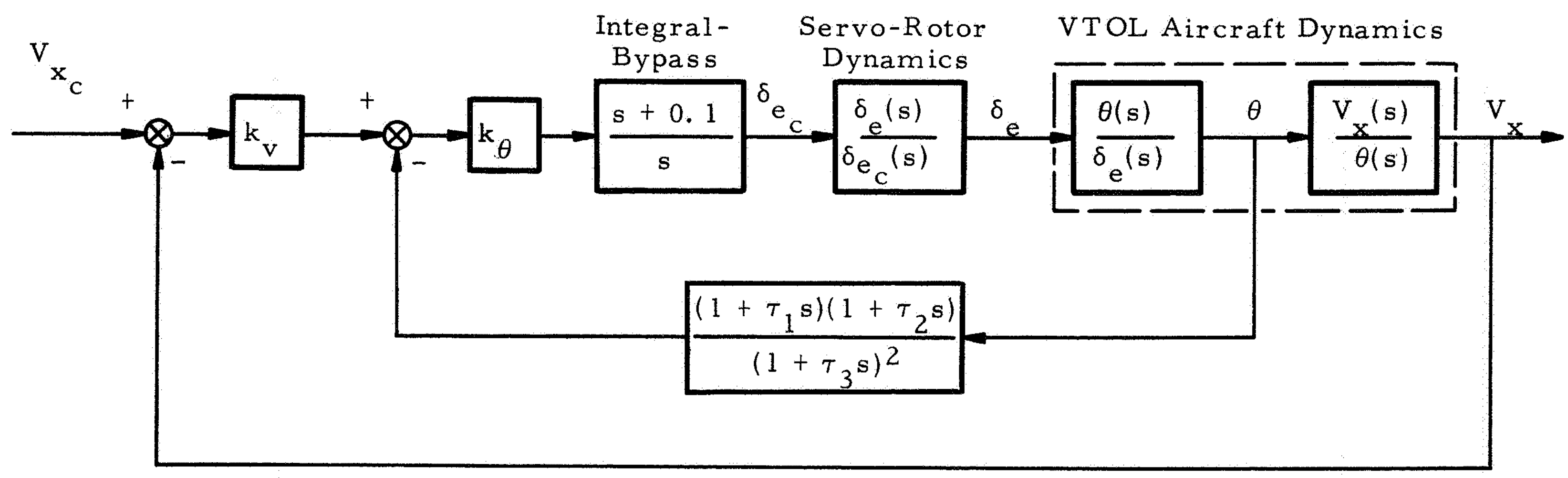
$$\underline{\tilde{x}}'_{m_0} = [\underline{x}'_{m_0} \;\vdots\; -\underline{x}'_{m_0}\underline{\alpha}]$$

$$\underline{x}'_{m_0} = [-1.0 \quad -.0727 \quad 0.0987]$$

The feedback configuration considered here, shown in figure 6-31, differs from that in reference 52 only in the velocity feedback path. A first order lag with a 0.3 second time constant is included in the velocity feedback path in reference 52. The integral-bypass shown in the forward path in figure 6-31 is the same as that used in reference 52 and is assumed to be fixed. The lead-lag configuration in the pitch attitude feedback is chosen to allow for lead in the frequency range of the dominant modes and yet attenuate high frequency noise. It can be written as

$$\frac{(1 + \tau_1 s)(1 + \tau_2 s)}{(1 + \tau_3 s)^2} = \frac{(1 + \tau_{\dot{\theta}} s + \tau_{\ddot{\theta}} s^2)}{(1 + \tau_3 s)^2} \qquad (6\text{-}61)$$

where



$$\frac{\theta(s)}{\delta_e(s)} = \frac{2.543\left(1 + \frac{s}{0.02056}\right)}{\left(1 + \frac{s}{0.8669}\right)\left[1 - \frac{2(0.226)}{(0.429)} s + \left(\frac{s}{0.429}\right)^2\right]} \quad , \quad \frac{\text{DEG}}{\text{INCH}}$$

$$\frac{V_x(s)}{\theta(s)} = \frac{-26.773\left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{\left(1 + \frac{s}{0.02056}\right)} \quad , \quad \frac{\text{FPS}}{\text{DEG}}$$

$$\frac{\delta_e(s)}{\delta_{e_c}(s)} = \frac{1}{\left(1 + \frac{s}{15.2}\right)\left(1 + \frac{s}{14.3}\right)}$$

Figure 6-31 Block Diagram of a Velocity Command FCS for Comparison to Reference 52

$$\tau_{\dot{\theta}} = \tau_1 + \tau_2$$

$$\tau_{\ddot{\theta}} = \tau_1 \tau_2$$

The double lag time constant, τ_3, can be set by noise considerations and should not significantly affect the dominant step response characteristics. Therefore τ_3 is assumed to be a fixed constant in the parameter optimization process and to have a sufficiently small value to be negligible. The pitch attitude gain, k_θ, the lead compensation constants, $\tau_{\dot{\theta}}$ and $\tau_{\ddot{\theta}}$, and the velocity gain, k_v, are the free design parameters. The servo-rotor dynamics will be neglected in the design process but will be included in the final check of the design. The closed-loop transfer function of V_{x_c} to V_x, neglecting τ_3 and the servo-rotor dynamics, is

$$\frac{V_x(s)}{V_{x_c}(s)} = \frac{b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \qquad (6\text{-}62)$$

where

$$a_3 = (0.6725 + 19.7 k_\theta \tau_{\dot{\theta}} + 0.197 k_v k_\theta \quad 2.375 k_\theta \tau_{\ddot{\theta}})/c$$

$$a_2 = (0.0152 + 19.7 k_\theta + 2.375 k_\theta \tau_{\dot{\theta}} + 0.0405 k_\theta \tau_{\ddot{\theta}} + 0.501 k_v k_\theta)/c$$

$$a_1 = (0.1593 + 2.375 k_\theta + 0.0405 k_\theta \tau_{\dot{\theta}} - 10.796 k_v k_\theta)/c$$

$$a_0 = (0.0405 k_\theta - 1.084 k_v k_\theta)/c \qquad (6\text{-}63)$$

and

$$b_3 = (0.197 k_v k_\theta)/c$$

$$b_2 = (0.501 k_v k_\theta)/c$$

$$b_1 = (-10.796 k_v k_\theta)/c$$

$$b_0 = (-1.084 k_v k_\theta)/c \qquad (6\text{-}64)$$

and

$$c = 1 + 19.7 k_\theta \tau_{\ddot{\theta}} \tag{6-65}$$

In applying the computer program for the Model PI design method, the subroutine COEF is written to generate the coefficients (6-63) and (6-64).

<u>Model PI Design</u>

Using a pseudo IC weighting factor of 1.3×10^{-4} in the Model PI and an initial choice for the free parameters of

$$k_\theta = 0.15 \text{ inch/deg}$$

$$\tau_{\dot{\theta}} = 0.11 \text{ sec}^{-1}$$

$$\tau_{\ddot{\theta}} = 0.667 \text{ sec}^{-2}$$

$$k_v = -0.667 \text{ deg/fps}$$

resulted in final values of

$$k_\theta = 0.251 \text{ inch/deg}$$

$$\tau_\theta = 0.469 \text{ sec}^{-1}$$

$$\tau_\theta = -0.071 \text{ sec}^{-2}$$

$$k_v = -1.04 \text{ deg/fps}$$

Using these values in equations (6-62) - (6-65) gives the approximate closed-loop system transfer function as

$$\frac{V_x(s)}{V_{x_c}(s)} = \frac{(-0.965)\left(1 + \frac{s}{0.1}\right)\left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{\left(1 + \frac{s}{0.0943}\right)\left(1 + \frac{s}{1.60}\right)\left[1 + \frac{2(0.8)}{(1.735)} s + \left(\frac{s}{1.735}\right)^2\right]} \tag{6-66}$$

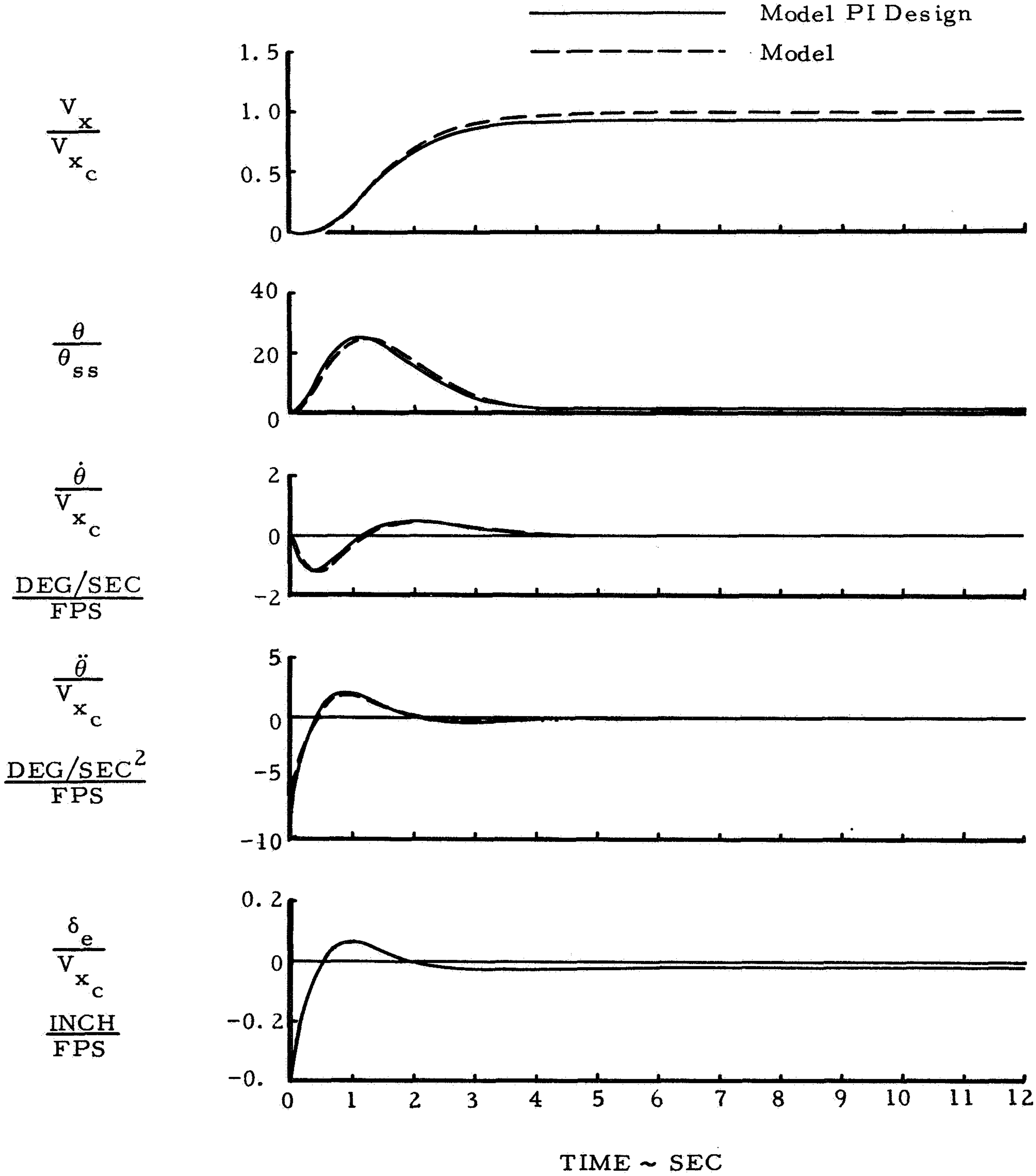


Figure 6-32 Step Response Comparison of the Model and Model PI Design for a Fourth Order Approximation to the FCS

The step response for this solution is shown in figure 6-32 to be very close to that of the model. The damping ratio of the oscillatory mode is 0.8, so that this design of the fourth order approximation to the system meets the design specifications of this section.

Since the Model PI design selected such a small value for $\tau_{\ddot{\theta}}$ it may be satisfactory to neglect $\tau_{\ddot{\theta}}$ altogether. If $\tau_{\ddot{\theta}}$ can be eliminated then it won't be necessary to include the double lag in the pitch attitude feedback path, and the compensation can be reduced to $(1 + \tau_{\dot{\theta}} s)$. The closed-loop system transfer function for the Model PI design, except $\tau_{\ddot{\theta}} = 0$, and including the servo-rotor dynamics is

$$\frac{V_x(s)}{V_{x_c}(s)} = \frac{(0.965)\left(1 + \frac{s}{0.1}\right)\left(1 - \frac{s}{6.2984}\right)\left(1 + \frac{s}{8.7396}\right)}{\left(1 + \frac{s}{0.094}\right)\left(1 + \frac{s}{1.09}\right)\left(1 + \frac{s}{7.4}\right)\left(1 + \frac{s}{19.5}\right)\left[1 + \frac{2(0.5)}{(2.0)} s + \left(\frac{s}{2.0}\right)^2\right]} \tag{6-67}$$

Comparing the oscillatory mode of transfer functions (6-66) and (6-67) one sees that neglecting $\tau_{\ddot{\theta}}$ gives a lower damping ratio but still within the specifications. The step response of this design is compared to that of the design from reference 52 in figure 6-33, which shows no significant differences. This Model PI design meets the command input design specifications of reference 52. The compensation used in the two designs is summarized in table 6-3 (figure 6-31) shows the basic feedback configuration).

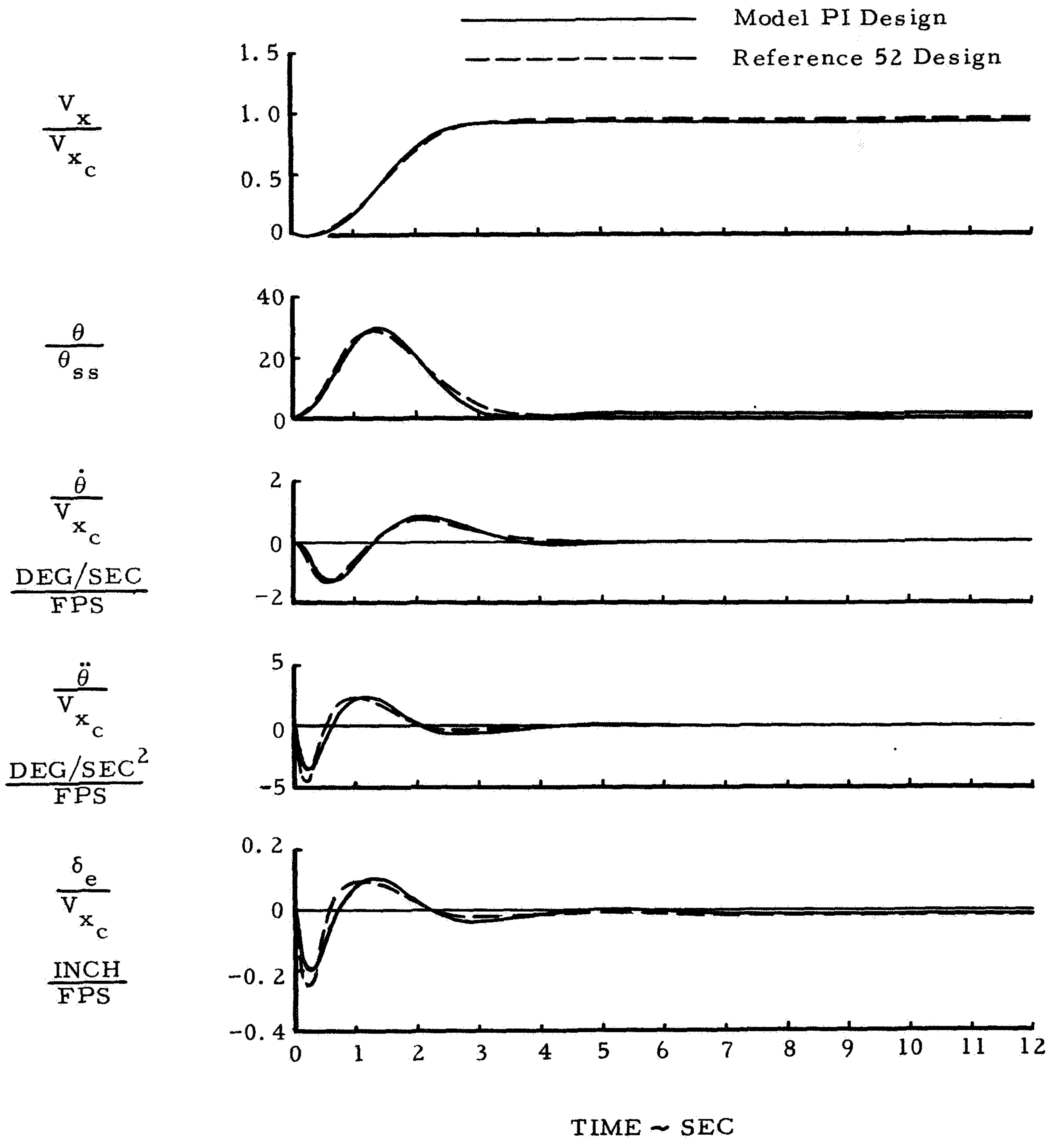


Figure 6-33 Comparison of the Step Responses of the Model PI Design and the Design of Reference 52

TABLE 6-3

COMPENSATION USED IN REFERENCE 52 AND IN THE MODEL PI DESIGNS

	Reference 52 Design	Model PI Design
Pitch Attutude Feedback Path Compensation	$\frac{(1 + 0.25s)^2}{(1 + 0.0357s)^2}$	(1 + 0.469s)
Pitch Attitude Gain, k_θ	0.40 inch/deg.	0.251 inch/deg.
Velocity Feedback Path Compensation	$\frac{1}{(1 + 0.3s)}$	Unity
Velocity Gain, k_v	−0.885 deg/fps	−1.04 deg/fps

CHAPTER 7

ON LINEAR OPTIMAL CONTROL

General quadratic functionals appear most frequently in modern control literature as cost functionals in optimal control theory. The main reason for this is that quadratic functionals are one of the few types of mathematical expressions that can give rise to closed-form analytical solutions for a variety of optimal control problems. Under suitable, well defined and well known, conditions (53) it is possible to force the optimal solution to be a linear feedback control law. The so called "optimal regulator" problem is one such solution that has been proposed and used as a synthesis tool for designing linear feedback systems. However there has been a gap between theory and application at the point of relating the quadratic cost functional to the desired closed-loop system characteristics. Rynaski and Whitbeck (26) and others have made important contributions towards closing this gap, but there is still a missing link, that of a logical basis for selecting the state vector weighting matrix in the quadratic functional. Without this link, there remains a large degree of arbitrariness in the "optimal solution" that incites substantial criticism of the technique.

This chapter presents some interesting new developments on the theory and application of linear optimal control resulting from the Model PI theory. By defining the cost functional as

$$J = \frac{1}{2}\int_0^{\infty} \left[\, \| \underline{\tilde{x}}(t) \|^2_{\tilde{Q}} + r\, u^2(t) \, \right] dt \qquad (7\text{-}1)$$

where the first term of the integral is the basic Model PI, one has a logical basis for selecting the weighting matrix, $\tilde{Q}$, directly in terms of a model. It will be proved in section 7. 2 that the optimal solution based on (7-1) approaches the model represented by $\tilde{Q}$ as the weighting on the control effort, r, approaches zero. The weighting matrix $\tilde{Q}$ can represent a model of equal or lower order than the system. If the model is of lower order, the excess system poles approach a Butterworth configuration as $r \rightarrow 0$. If the model and system are of the same order the optimal solution matches the model exactly <u>if and only if</u> $r = 0$. Section 7. 2. 2 shows how to interpret the output-regulator problem, in which the system output variables are used in the cost functional rather than the state vector, in terms of the Model PI concept. It will be shown in section 7. 2. 3 that Kalman's model-in-the-performance-index (54) in a state regulator form becomes independent of the state vector weighting matrix and is equivalent to (7-1) when the system and model are in the phase-variable canonical form.

Using the Model PI weighting matrix, $\tilde{Q}$, in the optimal regulator problem results in the poles of the model being the zeros of root square locus for the optimal solution. This fact leads to an interestingly simple solution to the linear optimal control synthesis procedure which is presented in section 7. 3. The simple pitch damper flight control system example considered in Chapter 6, section 6. 1, is redesigned to illustrate this procedure.

A procedure is presented in section 7. 4 for computing an equivalent Model PI for any quadratic function. This allows one to give a physical interpretation to some of the strictly mathematical examples that have appeared in optimal control theory literature. An example of this is also presented.

The standard approach to the optimal regulator problem will be reviewed in section 7. 1 before presenting the new material.

7. 1 The Single Control Optimal Regulator Problem

Consider the nth order linear, time invariant system given by

$$\dot{\underline{x}}(t) = F\,\underline{x}(t) + \underline{g}\,u(t) \tag{7-2}$$

$$\underline{y}(t) = H\,\underline{x}(t) \tag{7-3}$$

with initial condition $\underline{x}_0$, where $\underline{x}(t)$ is an $(n \times 1)$ state vector, $u(t)$ is a scalar control variable and $\underline{y}(t)$ is the output vector. The system is assumed to be controllable and observable (53). Also it is assumed that the system does not have direct transmission paths from $u(t)$ to any components of $\underline{y}(t)$, i. e. the corresponding transfer functions have at least one more pole than zero. With these two assumptions it is completely general to assume that the state equation (7-2) is in the canonical phase-variable form*, i. e.

$$F = \left[\begin{array}{c|c} \underline{0} & I \\ \hline \multicolumn{2}{c}{-\underline{a}'} \end{array}\right] \tag{7-4}$$

$$\underline{g} = \left[\begin{array}{c} \underline{0} \\ \hline 1 \end{array}\right] \tag{7-5}$$

The cost functional is defined as

$$J = \tfrac{1}{2} \int_0^{\infty} \left[\| \underline{y}(t) \|^2_Q + r\, u^2(t) \right] dt \tag{7-6}$$

where Q is a positive semi-definite symmetric matrix and r is a positive scalar.

The optimal regulator problem is to find the control $u(t)$, that takes the state from the initial condition, $\underline{x}_0$, to the origin, $\underline{0}$, along the trajectory that minimizes the cost functional (7-6). The well known solution (53) is

---

* The elements of the rows of H in this case are simply the numerator coefficients of the respective transfer functions.

$$u(t) = -\underline{k}'\underline{x}(t) \tag{7-7}$$

where

$$\underline{k}' = r^{-1}\underline{g}'P \tag{7-8}$$

and P is a positive definite symmetric matrix satisfying the algebraic equation

$$F'P + PF - P\underline{g}\,r^{-1}\underline{g}'P + H'QH = O \tag{7-9}$$

The " optimal regulator" is then given by

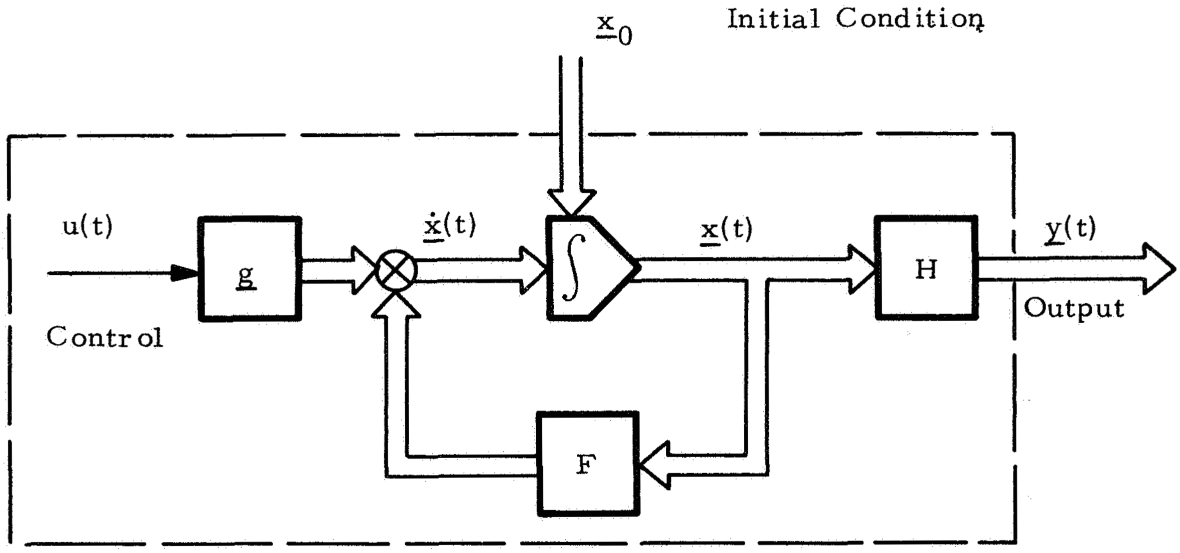
$$\begin{aligned} \underline{x}(t) &= [F - \underline{g}k'] \; \underline{x}(t) \\ &= F_{or} \, \underline{x}(t) \end{aligned} \tag{7-10}$$

where

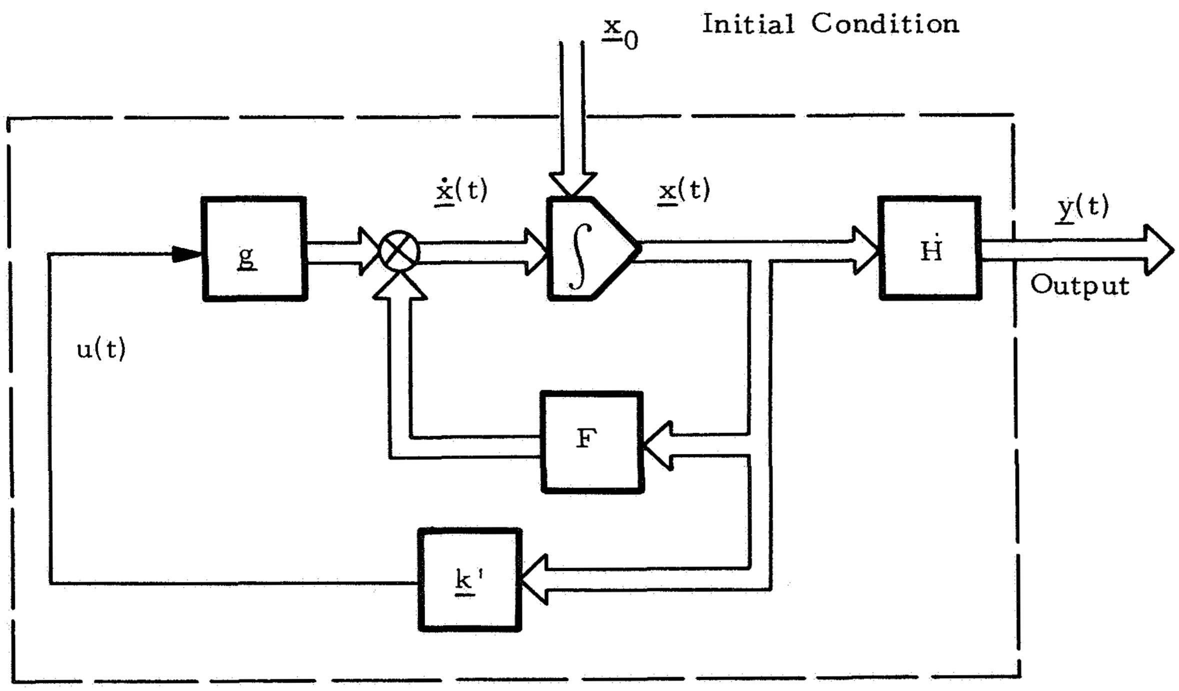
$$F_{or} = \left[\begin{array}{c|c} \underline{0} & I \\ \hline \multicolumn{2}{c}{-(\underline{a} + \underline{k})'} \end{array} \right] \tag{7-11}$$

and the output is given by (7-3). This is sometimes referred to as the " output-regulator problem" since the output vector is used in the cost functional. If the state vector is used in the cost functional it is called the " state-regulator problem". The solution to the state-regulator problem is obtained from the above by setting H = I.

The open- and closed-loop systems are depicted in figure 7-1 by vector-matrix block diagrams. These diagrams help emphasize the fact that this is a regulator type problem since there is no input to the system. One can not talk about closed-loop zeros for this system because there is no input-output transfer. The H matrix represents the zeros of the u(t) to $\underline{y}(t)$ transfer functions, but u(t) is strictly a control variable not an external input. The optimal control law (7-7) only changes the characteristic equation, i. e. poles, of the regulator.



(a) Open-Loop System



(b) Closed-Loop Optimal Regulator

Figure 7-1 Block Diagrams of Open- and Closed-Loop Systems for the Optimal Regulator Problem

The optimal regulator problem has been suggested as a procedure for designing linear feedback control systems. However one should realize that it is only a procedure for selecting the closed-loop characteristic equation. The solution for the feedback gains depends on the arbitrary weighting matrix, Q, and the scalar r as indicated in equations (7-8) and (7-9). The synthesis procedure using the optimal regulator solution is outline below.

1. Select a weighting matrix Q and the scalar r; typically Q is taken to be diagonal.
2. Compute the feedback gain vector, $\underline{k}$, for this choice of Q and r from equations (7-8) and (7-9) or P can be computed as the steady-state solution of the matrix Ricatti equation.
3. Compare the closed-loop system using these feedback gains to the design specifications.
4. Repeat steps 1 through 3 until the specifications are met.

There has not been any general guide in the past for choosing Q and r, which meant that several iterations of trial and error might be necessary. Chang (55) showed that the weighting factors of J, for the scalar output case, are related to the closed-loop poles of the optimum system and its adjoint by a root square locus. Rynaski and Whitbeck (26) extended Chang's work to the multivariable case and obtained some very interesting results with root square locus. They showed that in general the Q and R^* matrices are related to the closed-loop poles by a root square locus. However their treatment resulted in a form for the root square locus which is difficult to use as a guide to selecting Q except for the scalar output case. The difficulty is well illustrated by an example of reference 26 in which Q is a 2×2 diagonal matrix,

---

* R is the multi-control weighting matrix which is the counter part of r in equation (7-6).

$$Q = \begin{bmatrix} q_1 & 0 \\ 0 & q_2 \end{bmatrix} \tag{7-12}$$

and the control is a scalar, i.e. R is a scalar, r. Their solution was a root square locus with r^{-1} as the gain, and the numerator was another root square locus with the ratio (q_1/q_2) as the gain. In other words, one would have to use a root square locus to obtain the zeros of another root square locus. It becomes increasingly more difficult as Q has more elements. This is the type of complexity which analytical design techniques were developed to avoid.

Two ways in which a model can be included in the cost functional that have been considered in the literature are nodel-following method and Kalman's model-in-the-performance-index method (both are treated in reference 26). However in each of these one is still left with the task of selecting Q and r by a trial and error procedure similar to that outlined above. The Model PI concept to be considered in the next section provides a direct method for selecting the state vector weighting matrix apriori based on a model, which then leaves r as the only arbitrary quantity in the cost functional.

7.2 The Optimal Regulator Via the Model PI

The nth order system given by (7-2) and (7-3) can be written equivalently as

$$\underline{\tilde{x}}'(t)\,\underline{\tilde{a}} = u(t) \tag{7-13}$$

$$\underline{y}(t) = \tilde{H}\underline{\tilde{x}}(t) \tag{7-14}$$

with initial condition $\underline{\tilde{x}}_0$, where

$$\underline{\tilde{x}}'(t) = [\,\underline{x}'(t) \;\vdots\; x^{(n)}(t)\,] = [\,\underline{x}'(t) \;\vdots\; -\underline{x}'(t)\,\underline{a} + u(t)\,] \tag{7-15}$$

$$\underline{\tilde{a}}' = [\;\underline{a}' \;\vdots\; 1\;] \tag{7-16}$$

$$\tilde{H} = [\;H \;\vdots\; \underline{0}\;] \tag{7-17}$$

Three important differences should be pointed out between the systems considered here and in Chapter 3. In the above formulation the coefficient vector, $\underline{a}$, defines the open-loop characteristic equation and is a known constant; whereas, in Chapter 3 it defined the closed-loop characteristic equation and was a function of free design parameters. The initial condition vector is a specified constant here; whereas, in Chapter 3 it was a pseudo IC vector representing the effect of the closed-loop system zeros and was, in general, a function of the free design parameters. This means that the "optimum" system in this case will be independent of $\underline{x}_0$. The third difference is in the basic theoretical concepts of "optimal control" versus "parameter optimization". That is, the optimization process here is to select a control law for u(t); whereas, in Chapter 3 it was to select free design parameters after the engineer chose the feedback configuration.

It is assumed here that the objective of the optimal regulator problem is somewhat different than that stated in section 7.1 (page 241). Rather than seeking the control u(t) that merely takes the state from $\underline{x}_0$ to $\underline{0}$ along the trajectory that minimizes an arbitrary cost functional, it is required that the minimum value of the cost functional correspond to a trajectory that approximates the trajectory of a preselected model.

7.2.1 State-Regulator Problem

The geometrical representation of linear autonomous systems, intorduced in Chapter 3, section 3.1.1, can be used to define an ℓth order model by its characteristic plane ($\underline{\hat{\alpha}}$-plane)

$$\underline{\hat{x}}'_m(t)\,\underline{\hat{\alpha}} = 0 \qquad (7\text{-}18)$$

and an initial condition vector $\underline{\hat{x}}_{m_0}$, where

$$\underline{\hat{\alpha}}' = [\underline{\alpha}' \;\vdots\; 1] \qquad (7\text{-}19)$$

and $\underline{\alpha}$ is an $(\ell \times 1)$ vector whose elements are the coefficients of the model's characteristic equation. The model's time response trajectory lies within the $\underline{\hat{\alpha}}$-plane in the $(\ell + 1)$ dimensional space. Since the system's trajectory would in general be defined in the (n + 1) dimensional space only its projection into the model's extended state space, can be considered for approximating the model's trajectory. If $\underline{\hat{x}}(t)$ is the projection of the system's trajectory into $(\ell + 1)$-space then

$$\underline{\hat{x}}(t) = [\, I \;\vdots\; O \,]\, \underline{\bar{x}}(t) \tag{7-20}$$

the model's state vector initial condition, $\underline{x}_{m_0}$, can be chosen to coincide with the first ℓ elements of the system's initial condition vector. Then, if it is possible to find a control law for u(t), such that the projection of the system's trajectory, $\underline{\hat{x}}(t)$, lies within the model's characteristic plane for all time greater than zero, $\underline{\hat{x}}(t)$ must coincide with $\underline{\hat{x}}_m(t)$ except for possibly an arbitrarily small region around $\underline{\hat{x}}_{m_0}$. This statement can be proved by an argument similar to that used in Chapter 3, section 3. 2. 1. Since only the "optimal" control law is of interest here, this general proof will not be presented. Instead, it will be shown subsequently that the dominant poles of the optimal regulator approach the model poles in a limiting case, which is comparable to making $\underline{\hat{x}}(t)$ lie within the model's characteristic plane, $\underline{\hat{\alpha}}$-plane.

Following the logic of Chapter 3, section 3. 2. 1, it is reasonable to expect that if the projection of the system's trajectory into the $(\ell + 1)$-space could be made to lie close to the $\underline{\hat{\alpha}}$-plane by proper selection of u(t), then it would be close to the model's trajectory. Therefore, a criterion for approximating the model's trajectory by that of the system can be the minimization of the same generalized measure of the distance between the projection of the system's trajectory and the model's characteristic plane that was used in defining the Model PI. It was shown in Chapter 3 that the instantaneous distance from $\underline{\hat{x}}(t)$ to the $\underline{\hat{\alpha}}$-plane is the same as the distance from $\underline{\bar{x}}(t)$ to the model's extended characteristic plane, $\underline{\bar{\alpha}}$-plane, in (n + 1)-space where

$$\underline{\bar{\alpha}}' = [\, \underline{\hat{\alpha}}' \;\vdots\; \underline{0}' \,] \tag{7-21}$$

It follows that (see equations (3-33) to (3-36)) that minimizing the quadratic functional

$$\int_0^{\infty} \| \tilde{\underline{x}}(t) \|^2_{\tilde{Q}} \, dt \qquad (7\text{-}22)$$

where $\tilde{Q}$ is an $(n + 1) \times (n + 1)$ matrix defined as

$$\tilde{Q} = \frac{\tilde{\underline{\alpha}} \tilde{\underline{\alpha}}}{\| \tilde{\underline{\alpha}} \|^2} \qquad (7\text{-}23)$$

tends to force the system's trajectory to approximate the model's trajectory in $(\ell + 1)$-space.

The optimal regulator problem includes a requirement that the control effort not be excessive; so that the cost functional is taken to be (7-1) which is repeated here

$$J = \tfrac{1}{2} \int_0^{\infty} \left[\| \tilde{\underline{x}}(t) \|^2_{\tilde{Q}} + r\, u^2(t) \right] dt \qquad (7\text{-}1)$$

where $\tilde{Q}$ is given by (7-23). This cost functional represents a compromise between attempting to force the projection of the system's trajectory to lie in the model's characteristic plane and minimizing the control effort. Note that the only arbitrary constant is the weighting on the control effort, r. Selecting a model completely specifies $\tilde{Q}$. One can see from its definition (7-23) that $\tilde{Q}$ is always positive semi-definite symmetric matrix.

The optimal control law for this formulation is the same as in section 7.1 if the model is of lower order than the system, but somewhat different if the system and model are of the same order. Consider the case where the model is of lower order than the system $(\ell < n)$, then at least the last element of $\tilde{\underline{\alpha}}$ must be zero, and

$$\| \tilde{\underline{x}}(t) \|^2_{\tilde{Q}} = \| \underline{x}(t) \|^2_{Q} \qquad (7\text{-}24)$$

where Q is an $n \times n$ matrix given by

$$Q = \frac{1}{\|\underline{\hat{\alpha}}\|^2} \begin{bmatrix} \underline{\hat{\alpha}} \\ --- \\ \underline{0} \end{bmatrix} \left[\underline{\hat{\alpha}}' \vdots \underline{0}' \right] \tag{7-25}$$

and $\underline{0}$ is an $(n - \ell - 1) \times 1$ null vector. Using (7-24) in (7-1) the cost functional becomes exactly the standard form (7-6) for the case where $H = I$. The solution is given by equations (7-7) - (7-9) with $H = I$ and Q defined by (7-25).

An alternate form of the solution in the Laplace transform domain, called the "root square locus" (26, 55) provides an interesting result. The minimum principle* is used to establish the necessary condition for the extremal control law. The Hamiltonian for this problem is

$$\mathcal{H} = \tfrac{1}{2} \left[\|\underline{x}(t)\|_Q^2 + r\, u^2(t) \right] + \underline{\lambda}'(t) \left[F \underline{x}(t) + \underline{g}\, u(t) \right] \tag{7-26}$$

so that the necessary conditions are

$$\frac{\partial \mathcal{H}}{\partial u(t)} = r\, u(t) + \underline{\lambda}'(t) \underline{g} = 0 \tag{7-27}$$

$$\frac{\partial \mathcal{H}}{\partial \underline{x}(t)} = \underline{x}'(t) Q + \underline{\lambda}'(t) F = -\dot{\underline{\lambda}}'(t) \tag{7-28}$$

together with equation (7-2). Taking the Laplace transform of these and rearranging, gives

---

* The procedure for determining the optimal control by Pontryagin's Maximum (minimum) Principle is found in several references, e.g. Athans and Falb (53).

$$\begin{bmatrix} [Is - F] & O & -\underline{g} \\ -Q & [-Is - F'] & \underline{0} \\ \underline{0}' & \underline{g}' & r \end{bmatrix} \begin{bmatrix} \underline{x}(s) \\ \underline{\lambda}(s) \\ u(s) \end{bmatrix} = \begin{bmatrix} \underline{x}(0) \\ -\underline{\lambda}(0) \\ 0 \end{bmatrix} \qquad (7\text{-}29)$$

which describes the optimal regulator and its adjoint system. The closed-loop poles of the optimal system and its adjoint are given by the determinant of the coefficient matrix of (7-29). Rynaski and Whitbeck (26) show that this determinant can be reduced to

$$D(s)D(-s) \left| r + \underline{g}'[-Is - F']^{-1} Q [Is - F]^{-1} \underline{g} \right| = 0 \qquad (7\text{-}30)$$

where

$$D(s) = |Is - F| \qquad (7\text{-}31)$$

The poles of the open-loop system and adjoint system are given by $D(s) = 0$ and $D(-s) = 0$ respectively. The expression

$$r^{-1}\underline{g}'[-Is - F']^{-1} Q [Is - F]^{-1} \underline{g} = -1 \qquad (7\text{-}32)$$

defines the root square locus of the closed-loop poles of the optimal regulator and its adjoint as a function of r^{-1}.

Since F is in the phase-variable form (7-4) and Q is taken to be the special form (7-25), it is possible to reduce (7-32) to a very simple form. Define a vector $\underline{\sigma}(s)$ as

$$\underline{\sigma}(s) = \text{Adj}\,[Is - F]\,\underline{g} \qquad (7\text{-}33)$$

where Adj $[\bullet]$ means the adjoint of $[\bullet]$. The vector $\underline{\sigma}(s)$ can be evaluated by considering the first equation in (7-29), assuming $\underline{x}(0) = \underline{0}$,

$$[Is - F]\,\underline{x}(s) - \underline{g}\,u(s) = \underline{0}$$

or

$$\underline{x}(s) = [Is - F]^{-1}\underline{g}\,u(s) = \frac{\underline{\sigma}(s)}{D(s)}\,u(s) \tag{7-34}$$

The Laplace transform of the state equation (7-13) with $\underline{x}_0 = \underline{0}$ gives

$$D(s)\,x(s) = u(s)$$

and thus

$$\begin{aligned} D(s)\,\dot{x}(s) &= s\,u(s) \\ D(s)\,\ddot{x}(s) &= s^2 u(s) \\ &\vdots \\ D(s)\,x^{(n-1)}(s) &= s^{n-1}u(s) \end{aligned} \tag{7-35a}$$

or collecting these into vector notation

$$D(s)\,\underline{x}'(s) = [1 \quad s \quad s^2 \;\cdots\; s^{n-1}]\;u(s) \tag{7-35b}$$

Comparing (7-35b) to (7-34) one sees that

$$\underline{\sigma}'(s) = [1 \quad s \quad s^2 \cdots s^{n-1}] \tag{7-36}$$

Similarly one can show that

$$\underline{g}'\mathrm{Adj}\,[-Is - F'] = \underline{\sigma}'(-s) \tag{7-37}$$

Using the definitions (7-31) and (7-33) in the expression for the root square locus (7-32) gives

$$\frac{r^{-1}\underline{\sigma}'(-s)\;Q\;\underline{\sigma}(s)}{D(-s)\;D(s)} = -1 \tag{7-38}$$

Finally using the definition of Q, (7-25) in the above gives

$$\frac{\dfrac{r^{-1}}{\|\underline{\hat{\alpha}}\|^2}\left(\underline{\sigma}'(-s)\begin{bmatrix}\underline{\hat{\alpha}} \\ \hline \underline{0}\end{bmatrix}[\underline{\hat{\alpha}}' \;\vdots\; \underline{0}']\;\underline{\sigma}(s)\right)}{D(-s)\;D(s)} = -1$$

or

$$\boxed{\frac{\kappa\,(\alpha_0 - \alpha_1 s + \alpha_2 s^2 - \cdots \pm s^{\ell})(\alpha_0 + \alpha_1 s + \alpha_2 s^2 + \cdots + s^{\ell})}{(a_0 - a_1 s + a_2 s^2 - \cdots \pm s^{n})(a_0 + a_1 s + a_2 s^2 + \cdots + s^{n})} = -1}$$

(7-39)

where

$$\kappa = \frac{r^{-1}}{\|\hat{\underline{\alpha}}\|^2} \tag{7-40}$$

The denominator of the above expression is the product of the open-loop system and adjoint system characteristic polynomials, i. e. the product of the open-loop poles. If $\kappa \to 0$ ($r \to \infty$) the poles of the optimal regulator are the open-loop poles. This is the solution one would expect because as the weighting on the control effort in the cost functional gets very large it is better to use no control and operate open-loop.

The numerator of this expression is the product of the root square locus gain and the characteristic polynomials of the model and its adjoint. That is, the zeros of the root square locus for the optimal regulator and its adjoint are the poles of the model and its adjoint. If $\kappa \to \infty$ ($r \to 0$), ℓ of the poles of the optimal regulator and of its adjoint approach the respective poles of the model and its adjoint, i. e. the zeros of the root square locus (7-39). Altogether there are 2ℓ zeros and $2n$ poles in the root square locus expression. The remaining $2(n - \ell)$ poles go to infinity as $\kappa \to \infty$ along asymptotes that branch out at angles of

$$[(2i - 1)/2(n - \ell)]\ \pi \text{ for } (n - \ell) \text{ even}$$
$$[(2i - 2)/2(n - \ell)]\ \pi \text{ for } (n - \ell) \text{ odd}$$

with the real axis, $i = 1, 2, \cdots, 2(n - \ell)$. The asymptotes are symmetric with respect to the imaginary axis, and the $(n - \ell)$ excess poles of the optimal regulator approach the asymptotes in the left half plane (LHP). For large root locus gain the excess poles are approximately equidistant from the origin, thus approaching a Butterworth configuration (poles symmetrically arranged on a semicircle in the LHP whose center is at the origin) as they go to infinity. The dominant poles of the optimal regulator are then those that approach the model's poles ar $r \to 0$. This shows that the optimal solution based on (7-1) approaches the model represented by $\tilde{Q}$ (7-23) as the weighting on the control effort in (7-1) approaches zero, for models of lower order than the system.

Now consider the case where the model and system are of the same order ($\ell = n$). In this case there are no zero elements in $\underline{\tilde{\alpha}}$ so that

$$\| \underline{\tilde{x}}(t) \|^2_{\tilde{Q}} = \frac{1}{\| \underline{\tilde{\alpha}} \|^2} \left(\underline{\tilde{x}}'(t) \underline{\tilde{\alpha}} \right)^2$$

$$= \frac{1}{\| \underline{\tilde{\alpha}} \|^2} \left(\left[\underline{x}'(t) \; \vdots \; -\underline{x}'(t)\underline{a} + u(t) \right] \begin{bmatrix} \underline{\alpha} \\ \hdashline 1 \end{bmatrix} \right)^2$$

$$= \frac{1}{\| \underline{\tilde{\alpha}} \|^2} \left(\underline{x}'(t) (\underline{\alpha} - \underline{a}) + u(t) \right)^2 \tag{7-41}$$

Then the cost functional (7-1) is equivalent to

$$J = \frac{1}{2} \int_0^\infty \left[\frac{1}{\| \underline{\tilde{\alpha}} \|^2} \left(\underline{x}'(t) (\underline{\alpha} - \underline{a}) + u(t) \right)^2 + r\, u^2(t) \right] dt \tag{7-42}$$

The Hamiltonian corresponding to (7-42) with the constraint equation (7-2) is

$$\mathcal{H} = \frac{1}{2} \left[\frac{1}{\| \underline{\tilde{\alpha}} \|^2} \left(\underline{x}'(t) (\underline{\alpha} - \underline{a}) + u(t) \right)^2 + r\, u^2(t) \right]$$

$$+ \underline{\lambda}'(t) \left[F\underline{x}(t) + \underline{g}\, u(t) \right] \tag{7-43}$$

Using the minimum principle, the necessary conditions for an extremal control law are

$$\frac{\partial \mathcal{H}}{\partial u(t)} = \rho u(t) + \underline{x}'(t) \frac{(\underline{\alpha}-\underline{a})}{\|\underline{\tilde{\alpha}}\|^2} + \underline{\lambda}'(t)\underline{g} = 0 \tag{7-44}$$

$$\frac{\partial \mathcal{H}}{\partial \underline{x}(t)} = \frac{(\underline{\alpha}-\underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} u(t) + \underline{x}'(t) \frac{(\underline{\alpha}-\underline{a})(\underline{\alpha}-\underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} + \underline{\lambda}'(t)F = -\dot{\underline{\lambda}}'(t) \tag{7-45}$$

together with equation (7-2), where ρ is a positive scalar defined as

$$\rho = r + \frac{1}{\|\underline{\tilde{\alpha}}\|^2} \tag{7-46}$$

The control law is then

$$u(t) = -\rho^{-1}\left(\underline{g}'\lambda(t) + \frac{(\underline{\alpha}-\underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} \underline{x}(t) \right) \tag{7-47}$$

It is easily shown that this extremal control gives a local minimum value of $\mathcal{H}$. Since

$$\frac{\partial^2 \mathcal{H}}{\partial u^2(t)} = \rho > 0 \tag{7-48}$$

a second order variation in u(t) from (7-47) would cause an increase in $\mathcal{H}$.

Using the optimal control law (7-47) in equations (7-2) and (7-44) results in

$$\dot{\underline{x}}(t) = \bar{F}\underline{x}(t) - \underline{g}\rho^{-1}\underline{g}'\underline{\lambda}(t) \tag{7-49}$$

$$\dot{\underline{\lambda}}(t) = -\bar{F}'\underline{\lambda}(t) - \bar{Q}\underline{x}(t) \tag{7-50}$$

where

$$\bar{F} = \left[F - \rho^{-1}\underline{g} \frac{(\underline{\alpha}-\underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} \right] \tag{7-51}$$

$$\bar{Q} = \left(1 - \frac{\rho^{-1}}{\|\underline{\alpha}\|^2}\right) \frac{(\underline{\alpha}-\underline{a})(\underline{\alpha}-\underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} \tag{7-52}$$

It is easily shown by substitution into (7-49) and (7-50) that

$$\underline{\lambda}(t) = \bar{P}\underline{x}(t) \tag{7-53}$$

where $\bar{P}$ is a positive definite symmetrix matrix that satisfies the algebraic matrix equation

$$\bar{F}'\bar{P} + \bar{P}\bar{F} - \bar{P}\underline{g}\rho^{-1}\underline{g}'\bar{P} + \bar{Q} = O \tag{7-54}$$

Using (7-53) in (7-47) gives the optimal control law to be

$$u(t) = -\underline{k}'\underline{x}(t) \tag{7-55}$$

where

$$\underline{k}' = \rho^{-1}\left(\underline{g}'\bar{P} + \frac{(\underline{\alpha}-\underline{a})'}{\|\underline{\tilde{\alpha}}\|^2}\right) \tag{7-56}$$

The optimal solution for the case in which the model and system are of the same order is seen, from the above results, to be somewhat different from the usual solution (compare the above to equations (7-7) - (7-9)), although very similar in form. The possibility of exact model matching exists when the model and system are of the same order. It will now be shown that the optimal control law given by (7-54) - (7-56) will produce exact model matching if and only if r = 0.*

Assume that r = 0, then (7-56) becomes

$$\underline{k}' = (\underline{\alpha} - \underline{a})' + \|\underline{\tilde{\alpha}}\|^2 \underline{g}'\bar{P} \tag{7-57}$$

From the definitions of $\bar{Q}$ and $\bar{F}$, (7-52) and (7-51) respectively, with

---

* An exception is the special, trivial case in which the open-loop system matches the model exactly, i.e. $\underline{a} = \underline{\alpha}$, and the optimal control is $u(t) = 0$.

r = 0, it follows that $\bar{Q} = O$ and

$$F = \left[\begin{array}{c:c} \underline{0} & I \\ \hdashline \multicolumn{2}{c}{-\underline{a}'} \end{array}\right] - \left[\begin{array}{c} O \\ \hdashline (\underline{\alpha} - \underline{a})' \end{array}\right] = \left[\begin{array}{c:c} \underline{0} & I \\ \hdashline \multicolumn{2}{c}{-\underline{\alpha}'} \end{array}\right] \tag{7-58}$$

which is the coefficient matrix of the model. Equation (7-54) reduces to

$$\bar{F}'\bar{P} + \bar{P}\bar{F} - \bar{P}\underline{g}\,\|\tilde{\underline{\alpha}}\|^2\,\underline{g}'\bar{P} = O \tag{7-59}$$

so that

$$\bar{P} = O \tag{7-60}$$

is a solution. It follows from Potter's work (56) that equation (7-59) can only have one positive semi-definite solution*, so that (7-60) is the solution to use in the optimal feedback gain equation (7-57), i. e.

$$\underline{k}' = (\underline{\alpha} - \underline{a})' \tag{7-61}$$

and the optimal control is

$$u(t) = (\underline{\alpha} - \underline{a})'x(t) \tag{7-62}$$

Using (7-62) in (7-2) results in the optimal regulator state equation

$$\begin{aligned} \dot{\underline{x}}(t) &= [F - \underline{g}(\underline{\alpha} - \underline{a})'] \, \underline{x}(t) \\ &= F_{or}\,\underline{x}(t) \end{aligned} \tag{7-63}$$

where

---

* Since only asymptotically stable models are considered here, $\bar{F}$ is a stable matrix so that Potter's regularity condition is met in equation (7-59).

$$F_{or} = \left[\begin{array}{c|c} \underline{0} & I \\ \hline \multicolumn{2}{c}{-\underline{a}'} \end{array}\right] - \left[\begin{array}{c} O \\ \hline (\underline{\alpha} - \underline{a})' \end{array}\right] = \left[\begin{array}{c|c} \underline{0} & I \\ \hline \multicolumn{2}{c}{-\underline{\alpha}'} \end{array}\right] \qquad (7\text{-}64)$$

The closed-loop coefficient matrix (7-64) is identically that of the model, so that the optimal regulator based on $\tilde{Q}$ matches the model represented by $\tilde{Q}$ exactly if $r = 0$.

It will be shown by the root square locus technique, that the optimal solution matches the model exactly only if $r = 0$. This will also show that if exact model matching is not wanted, i.e. $r \neq 0$, the optimal regulator at least approaches the model as r approaches zero.

Taking the Laplace transform of the necessary condition (7-2), (7-44) and (7-45) and rearranging gives

$$\left[\begin{array}{c|c|c} [Is - F] & O & -\underline{g} \\ \hline -\dfrac{(\underline{\alpha} - \underline{a})(\underline{\alpha} - \underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} & [-Is - F'] & -\dfrac{(\underline{\alpha} - \underline{a})}{\|\underline{\tilde{\alpha}}\|^2} \\ \hline \dfrac{(\underline{\alpha} - \underline{a})'}{\|\underline{\tilde{\alpha}}\|^2} & \underline{g}' & \rho \end{array}\right] \left[\begin{array}{c} \underline{x}(s) \\ \hline \underline{\lambda}(s) \\ \hline u(s) \end{array}\right] = \left[\begin{array}{c} \underline{x}(0) \\ \hline -\underline{\lambda}(0) \\ \hline 0 \end{array}\right] \qquad (7\text{-}65)$$

which describes the optimal regulator and its adjoint system. Following the procedure suggested by Rynaski and Whitbeck (26) one can obtain the expression

$$\frac{r^{-1}}{\|\underline{\tilde{\alpha}}\|^2}\Big(1 + \underline{g}'[-Is - F']^{-1}(\underline{\alpha} - \underline{a})\Big)\Big(1 + (\underline{\alpha} - \underline{a})'[Is - F]^{-1}\underline{g}\Big) = -1$$

as the root square locus equation in this case. Using the previously defined vector $\underline{\sigma}(s)$, equation (7-33), the above can be reduced to

$$\frac{r^{-1}}{\|\underline{\tilde{\alpha}}\|^2}\left(1 + \frac{\underline{\sigma}(-s)(\underline{\alpha} - \underline{a})}{D(-s)}\right)\left(1 + \frac{(\underline{\alpha} - \underline{a})'\underline{\sigma}(s)}{D(s)}\right) = -1 \qquad (7\text{-}67)$$

which can be further reduced using (7-36) to

$$r \|\underline{\tilde{\alpha}}\|^2 D(-s)\, D(s) + (\alpha_0 - \alpha_1 s + \alpha_2 s^2 - \cdots \pm s^n)(\alpha_0 + \alpha_1 s + \alpha_2 s^2 + \cdots + s^n) = 0 \qquad (7\text{-}68)$$

The poles of the optimal regulator (and its adjoint) given by (7-68) can match those of the model (and its adjoint) only if the first term of (7-68) is zero. The open-loop system characteristic polynomial, D(s), is zero only at the open-loop poles, i. e. u(t) = 0, which is a trivial solution. Then the first term is zero only if r = 0 (excluding the trivial solution), so that the optimal regulator based on $\tilde{Q}$ matches the model represented by $\tilde{Q}$ exactly only if r = 0.

Equation (7-68) can also be written in the root square locus form

$$\frac{\kappa\,(\alpha_0 - \alpha_1 s + \alpha_2 s^2 - \cdots \pm s^n)(\alpha_0 + \alpha_1 s + \alpha_2 s^2 + \cdots + s^n)}{(a_0 - a_1 s + a_2 s^2 - \cdots \pm s^n)(a_0 + a_1 s + a^2 s^2 + \cdots + s^n)} = -1 \qquad (7\text{-}69)$$

where κ is defined by (7-40). Therefore the poles of the model (and its adjoint) represented by $\tilde{Q}$ in the cost functional (7-1) are the zeros of the root square locus for the optimal regulator (and its adjoint). As $\kappa \to \infty$ ($r \to 0$) the poles of the optimal regulator and its adjoint approach the zeros of (7-69), and thus, the poles of the model and its adjoint respectively. The optimal regulator not only matches the model exactly if r = 0, but it approaches the model in a well defined manner as $r \to 0$.

The use of cost functional (7-1) in which $\tilde{Q}$ defined by (7-23), represents a specific model, has thus been justified for models of equal or lower order than the system. The root square locus expression (7-39) is valid for $\ell \leq n$ and can be written down directly from the model and open-loop system characteristics. It is interesting that although the optimal control laws are somewhat different for $\ell < n$ and $\ell = n$, the root square loci are of identical form.

7.2.2 Output-Regulator Problem

From the approach taken here, the output-regulator problem is no longer a meaningful problem. In fact, it complicates and restricts the selection of Q to represent a specific model. It was pointed out in section 7.1 that the optimal regulator problem is only a procedure for selecting the closed-loop characteristic equation of a system. It is true that the output matrix, H, does affect the optimal solution when treating the problem as an output-regulator. However one can always get the same solution using the state-regulator approach by selecting the appropriate Q. Since Q has been an arbitrary matrix in previous treatments anyway it seems somewhat superfluous to define the output-regulator problem separately from the state-regulator problem. As long as the system is observable from the output vector, $\underline{y}(t)$ and the closed-loop system is stable, then $\underline{y}(t)$ will go to zero as $\underline{x}(t) \to \underline{0}$. The only question is the manner in which $\underline{x}(t) \to \underline{0}$, which can be established conveniently using the Model PI approach suggested in the previous section.

However, since the output regulator is often considered in the literature, it is of interest to show how to intepret the cost functional in terms of the Model PI concept for such cases. The cost functional is (7-6) which is repeated here

$$J = \frac{1}{2} \int_0^\infty [\, \| \underline{y}(t) \|^2_Q + r\, u^2(t) \,] \qquad (7\text{-}6)$$

The weighting matrix, Q, is assumed here to be of the special form

$$Q = \underline{q}\underline{q}' \qquad (7\text{-}70)$$

For the case in which Q is a general, symmetric, positive semi-definite matrix, one would have to use the procedure described in section 7.4. The cost function can be written as

$$J = \frac{1}{2} \int_0^\infty [\, \| \tilde{\underline{x}}(t) \|^2_{\tilde{H}'Q\tilde{H}} + r\, u^2(t) \,]\, dt \qquad (7\text{-}71)$$

where $\tilde{\underline{x}}(t)$ and $\tilde{H}$ are defined by (7-15) and (7-17) respectively. Then

$$\tilde{H}'Q\tilde{H} = \tilde{H}'\underline{q}\,\underline{q}'\tilde{H} \tag{7-72}$$

Define a vector $\underline{\tilde{\alpha}}$ in this case to be

$$\underline{\tilde{\alpha}} = \tilde{H}'\underline{q} \tag{7-73}$$

and define

$$\bar{r} = \frac{1}{\| \underline{\tilde{\alpha}} \|^2} \tag{7-74}$$

$$\bar{J} = \frac{J}{\| \underline{\tilde{\alpha}} \|^2} \tag{7-75}$$

Minimizing $\bar{J}$ is equivalent to minimizing J because $\| \underline{\tilde{\alpha}} \|^2$ is just a positive scalar. Then

$$\bar{J} = \tfrac{1}{2} \int_0^\infty [\, \| \tilde{\underline{x}}(t) \|^2_{\tilde{Q}} + \bar{r}\, u^2(t) \,]\, dt \tag{7-76}$$

where

$$\tilde{Q} = \frac{\underline{\tilde{\alpha}}\,\underline{\tilde{\alpha}}'}{\| \underline{\tilde{\alpha}} \|^2} = \frac{\tilde{H}'Q\tilde{H}}{\| \underline{\tilde{\alpha}} \|^2} = \frac{\tilde{H}'\underline{q}\,\underline{q}'\tilde{H}}{\| \tilde{H}'\underline{q} \|^2} \tag{7-77}$$

The problem has thus been transformed into the one considered in section 7.2.1, so that the solution and root square locus techniques discussed previously can be applied.

If one wishes to formulate an output-regulator problem, then the definition (7-73) can be used to interpret the weighting matrix in terms of a model. But now there is a constraint on selecting the model. In section 7.2.1, $\underline{\tilde{\alpha}}$ was free to be selected, but here, only $\underline{q}$ is free. $\tilde{H}$ is a specified matrix whose elements are the numerator polynomial coefficients of the open-loop u(s) to $\underline{y}(s)$ transfer functions, i.e. the ith row of $\tilde{H}$, $\tilde{\underline{h}}_i$, is

$$\tilde{\underline{h}}_i = [b_{i0}\ b_{i1}\ b_{i2} \cdots b_{im}\quad 0\quad 0 \cdots 0] \tag{7-78}$$

for the ith output variable transfer function, $y_i(s)/u(s)$, with numerator polynomial

$$N_{[u,\ y_i]}(s) = b_{im}s^m + b_{im-1}s^{m-1} + \cdots + b_{i1}s + b_{i0} \tag{7-79}$$

The relationship (7-73) puts an interesting constraint on the model selected. The model's characteristic equation, defined by $\underline{\tilde{\alpha}}$, is a weighted sume of the numerator polynomials of the output variable transfer functions, with the elements of $\underline{q}$ being the weighting factors, i. e.

$$\underline{\tilde{\alpha}} = \sum_{i=1}^{j} q_i \tilde{\underline{h}}_i \tag{7-80}$$

where j is the number of output variables. This point is easier to see in a simple example.

Consider an open-loop system with two output variables given by

$$\frac{y_1(s)}{u(s)} = \frac{b_{12}s^2 + b_{11}s + b_{10}}{s^3 + a_2s^2 + a_1s + a_0}$$

$$\frac{y_2(s)}{u(s)} = \frac{b_{21}s + b_{20}}{s^3 + a_2s^2 + a_1s + a_0} \tag{7-81}$$

which can be represented in canonical, phase-variable, form as

$$\dot{\underline{x}}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix} \underline{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t) \tag{7-82}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} b_{10} & b_{11} & b_{12} \\ b_{20} & b_{21} & 0 \end{bmatrix} \underline{x}(t) = H\underline{x}(t) \tag{7-83}$$

$$\tilde{H} = [H \;\vdots\; \underline{0}] = \begin{bmatrix} b_{10} & b_{11} & b_{12} & 0 \\ b_{20} & b_{21} & 0 & 0 \end{bmatrix} \tag{7-84}$$

If the cost functional is

$$J = \frac{1}{2}\int_0^{\infty} [(q_1 y_1(t) + q_2 y_2(t))^2 + r\,u^2(t)]\,dt \tag{7-85}$$

$$\underline{\alpha} = \tilde{H}'\underline{q} = \begin{bmatrix} H' \\ \hline \underline{0}' \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

or

$$\begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = q_1 \begin{bmatrix} b_{10} \\ b_{11} \\ b_{12} \\ 0 \end{bmatrix} + q_2 \begin{bmatrix} b_{20} \\ b_{21} \\ 0 \\ 0 \end{bmatrix} \tag{7-86}$$

The coefficients for the model's characteristic equation is given by (7-86) in this example so that for any q_1 and q_2 one can write down the corresponding model and root square locus directly. But one is severely restricted in the models that could be represented by the weighting matrix in this example, with only q_1 and q_2 free to be chosen.

Two final observations can be made. First, the order of the model defined by (7-73) is the order of the highest order numerator polynomial among the $y_i(s)/u(s)$ transfer functions. Secondly, if only one output variable is included in the cost functional, i.e. Q is a scalar, then the zeros of the open-loop control to output transfer function are the poles of the model defined by (7-73), and therefore are zeros of the optimal output-regulator root square locus. As $r \rightarrow 0$ in (7-76) the dominant poles of optimal solution will approach the zeros of the open-loop system.

7.2.3 Relationship to Kalman's "Model-in-the-Performance-Index"

Kalman (54) suggested a method for including the coefficient matrix of a model in a quadratic cost functional in such a way as to try to force the derivative of the system's output vector to match the derivative of the model's state vector. This technique, which is referred to as "model-in-the-performance-index", proceeds as follows.

A model of the same order as the system output vector, where the system is given by equations (7-2) and (7-3), is defined as

$$\dot{\underline{x}}_m(t) = F_m \underline{x}_m(t) \tag{7-87}$$

Then the cost functional is taken to be

$$J = \frac{1}{2} \int_0^{\infty} [\, \| \dot{\underline{y}}(t) - F_m \underline{y}(t) \|_Q^2 + r\, u^2(t) \,]\, dt \tag{7-88}$$

If the control u(t) forces $\dot{\underline{y}}(t) - F_m \underline{y}(t)$ to be zero then the system output rate will match the derivative of the model's state vector. This is an "output-regulator" type of formulation. Only the "state-regulator" type of formulation will be considered here, leaving the treatment of the "output-regulator" type for a future effort. The "state-regulator" form is obtained from (7-88) by defining

$$H = [I \quad O] \tag{7-89}$$

where I is an $\ell \times \ell$ identity matrix and O is an $\ell \times (n - \ell)$ null matrix so that (7-3) becomes

$$\underline{y}(t) = [I \quad O]\, \underline{x}(t) \tag{7-90}$$

It will be shown in this case that if the system and model are in the canonical phase-variable form, then (7-88) is independent of Q, and indeed, becomes equivalent to the cost functional (7-1) based on the Model PI concept.

Assume that the model is ℓth order and its coefficient matrix is of the form

$$F_m = \begin{bmatrix} \underline{0} & \vdots & I \\ \hline & -\underline{\alpha}' & \end{bmatrix} \tag{7-91}$$

The variables $\dot{\underline{y}}(t)$ and $\underline{y}(t)$ can be eliminated from (7-88) by using (7-2) and (7-3) to give

$$J = \tfrac{1}{2} \int_0^{\infty} \left[\, \| (HF - F_m H)\underline{x}(t) + H\underline{g}\, u(t) \|_Q^2 + r\, u^2(t) \,\right] dt \tag{7-92}$$

First consider the case in which the model is of lower order than the system ($\ell < n$). In that case, at least the last column of H, (7-89), is a zero vector so that $H\underline{g} = \underline{0}$ in (7-92). The terms HF and $F_m H$ can be reduced as follows*:

$$HF = [I_{\ell\times\ell} \,\vdots\, O] \begin{bmatrix} \underline{0} & \vdots & I_{(n-1)\times(n-1)} \\ \hline & -\underline{a}' & \end{bmatrix}$$

$$= \begin{bmatrix} \underline{0} & \vdots & I_{\ell\times\ell} & \vdots & O \end{bmatrix} \tag{7-93}$$

---

* The dimensions of the various identity matrices are indicated by subscripts here to avoid confusion, and the O's are appropriately dimensioned null matrices.

$$F_m H = F_m [I_{\ell \times \ell} \; \vdots \; O] = [F_m \; \vdots \; O]$$

$$= \left[\begin{array}{c:c:c} \underline{0} & I_{(\ell-1)\times(\ell-1)} & \\ \hdashline \multicolumn{2}{c:}{-\underline{\alpha}'} & O \end{array} \right] \tag{7-94}$$

Then

$$(HF - F_m H) = \left[\begin{array}{c} O \\ \hdashline -[\underline{\alpha}' \quad 1 \quad \underline{0}'] \end{array} \right] = \left[\begin{array}{c} O \\ \hdashline -[\underline{\hat{\alpha}}' \quad \underline{0}'] \end{array} \right] \tag{7-95}$$

where $\underline{\hat{\alpha}}$ is defined by (7-19). The first term in the cost functional (7-92) reduces to

$$\left\| \left[\begin{array}{c} O \\ \hdashline -[\underline{\hat{\alpha}}' \quad \underline{0}'] \end{array} \right] \underline{x}(t) \right\|_Q^2 = \underline{x}'(t) \left[\begin{array}{c:c} -\underline{\hat{\alpha}} & \\ \hdashline \underline{0} & O \end{array} \right] Q \left[\begin{array}{c:c} \multicolumn{2}{c}{O} \\ \hdashline -\underline{\hat{\alpha}}' & \underline{0}' \end{array} \right] \underline{x}(t)$$

$$= \underline{x}'(t) \left[\begin{array}{c} \underline{\hat{\alpha}} \\ \hdashline \underline{0} \end{array} \right] q_{\ell\ell} [\underline{\hat{\alpha}}' \; \vdots \; \underline{0}'] \, \underline{x}(t)$$

$$= q_{\ell\ell} \left(\underline{\tilde{x}}'(t) \, \underline{\tilde{\alpha}} \right)^2 \tag{7-96}$$

where $q_{\ell\ell}$ is the element in the last row and column of Q, and $\underline{\tilde{x}}(t)$ and $\underline{\tilde{\alpha}}$ are defined by equations (7-15) and (7-21) respectively. Then the cost functional (7-92) can be written as

$$J = \frac{1}{2} \int_0^\infty [q_{\ell\ell} \left(\underline{\tilde{x}}'(t) \, \underline{\tilde{\alpha}} \right)^2 + r \, u^2(t)] \, dt \tag{7-97}$$

Notice that the cost functional is completely independent of Q because only one of the two scalars $q_{\ell\ell}$ and r in (7-97) is necessary to establish the relative weighting of the two quadratic terms. The scalar $q_{\ell\ell}$ can be taken to be

$$q_{\ell\ell} = \frac{1}{\|\underline{\alpha}\|^2} \tag{7-98}$$

without loss of generality since r has not been specified. Using (7-98) in (7-97) results in the Model PI form, (7-1), i. e.

$$J = \frac{1}{2}\int_0^{\infty} [\ \|\underline{\tilde{x}}(t)\|^2_{\tilde{Q}} + r\,u^2(t)\ ]\,dt \tag{7-1}$$

where $\tilde{Q}$ is given by (7-23).

Now consider the case in which the model and system are of the same order ($\ell = n$). In that case H = I, so that the first term in the integrand of (7-92) becomes

$$\|(F - F_m)\underline{x}(t) + \underline{g}\,u(t)\|^2_Q = \left\| \begin{bmatrix} O \\ \hline (\underline{\alpha} - \underline{a})' \end{bmatrix} \underline{x}(t) + \begin{bmatrix} \underline{0} \\ \hline u(t) \end{bmatrix} \right\|^2_Q$$

$$= q_{\ell\ell}\Big((\underline{\alpha} - \underline{a})'\,x(t) + u(t) \Big)^2 \tag{7-99}$$

Again take $q_{\ell\ell}$ to be (7-98). Comparing (7-99) to (7-41) one sees that

$$\|(F - F_m)\,\underline{x}(t) + \underline{g}\,u(t)\|^2_Q = \|\underline{\tilde{x}}(t)\|^2_{\tilde{Q}} \tag{7-100}$$

for any Q as long as $q_{\ell\ell}$ is given by (7-98). This means that (7-92) also reduces to (7-1) for models of the same order as the system.

Therefore, if the system and model are in the canonical phase-variable form, the "model-in-the-performance-index" cost functional in state-regulator form is completely independent of the Q matrix and is equivalent to the Model PI cost functional.

7.3 Feedback Control System Synthesis by Root Square Locus

The cost functional (7-1) based on the Model PI concept resulted in the simple form of the root square locus for the optimal regulator and its adjoint (7-39), that provided a theoretical justification for the cost functional. In this section, a synthesis procedure based on this simple root square locus form is presented. The procedure provides a direct solution to the optimal regulator problem as a function of the control weighting factor, r, which can be obtained graphically from one root square locus diagram using standard root locus techniques. A simple flight control system design example is used to illustrate the procedure.

Using a cost functional of the form (7-1) results in the optimal closed-loop system (and its adjoint) having poles given by the root square locus (7-39), repeated below,

$$\frac{\kappa(\alpha_0 - \alpha_1 s + \alpha_2 s^2 - \cdots \pm s^\ell)(\alpha_0 + \alpha_1 s + \alpha_2 s^2 + \cdots + s^\ell)}{(a_0 - a_1 s + a_2 s^2 - \cdots \pm s^n)(a_0 + a_1 s + a_2 s^2 + \cdots + s^n)} = -1 \tag{7-39}$$

for $\ell \leq n$, and $\kappa = (r \|\underline{\tilde{\alpha}}\|^2)^{-1}$. The corresponding closed-loop system (optimal regulator) characteristic equation is

$$x^{(n)}(t) + \underline{x}'(t)\,\underline{a}_{or} = 0 \tag{7-101}$$

where $\underline{a}_{or}$ is optimal regulator coefficient vector given by

$$\underline{a}_{or} = \underline{a} + \underline{k} \tag{7-102}$$

$\underline{a}$ is the open-loop system coefficient vector and $\underline{k}$ is the optimal feedback gain vector. The optimal regulator problem is solved once $\underline{k}$ is computed, which can be easily done from the root square locus (7-39) and equation (7-102). The simple form of (7-39) allows one to write it down immediately from the system and model coefficient vectors, $\underline{a}$ and $\underline{\alpha}$ respectively. Standard root locus techniques are used to plot the root square locus. For a specific value of κ one can obtain the optimal closed-loop poles from the locus and form $\underline{a}_{or}$ from the product of the poles. Once $\underline{a}_{or}$ is known, $\underline{k}$ is easily computed from (7-102).

Note that this provides a simple, almost trivial, graphical technique for solving the simgle-control, linear, optimal regulator problem for any order system and for models of equal or lower order. This can be done quickly by hand, and requires only a knowledge of standard root locus techniques.

This technique can be used to synthesize feedback control systems to meet engineering specifications given as desired closed-loop pole locations in the s-plane. The aircraft handling qualities requirements in the design examples of Chapter 6 are examples of such engineering specifications. The synthesis procedure using this approach would be as follows:

1. Plot the poles of the open-loop system (the plant) and of its adjoint, i. e. the denominator of (7-39).

2. Select zeros for the root square locus (7-39) that would tend to give a locus going through the desired location for the closed-loop system poles. These zeros then establish the model and correspondingly the $\tilde{Q}$ matrix.

3. Draw the root square locus with $(r \| \underline{\alpha} \|^2)^{-1}$ as the root locus gain. If some location on the root square locus gives satisfactory closed-loop poles, then calculate the corresponding gain vector, $\underline{k}$.

4. If this root square locus doesn't give any satisfactory closed-loop pole locations then repeat with a different set of zeros, i. e. a new $\tilde{Q}$ matrix. The first try should provide a good guide for selecting a new set of zeros.

5. Or if the gain vector $\underline{k}$ has some elements too large from a practical implementation standpoint, repeat with a new set of zeros.

An example is presented subsequently to illustrate the procedure.

At this point one might logically ask, why go through the immediate step of drawing a root square locus? Why not just pick the desired closed-loop pole locations in the s-plane, form the closed-loop coefficient vector, then compute $\underline{k}$ from an expression like (7-102)? Actually, with low order systems, one might be able to do just that and obtain a satisfactory design with reasonable levels of control effort. The problem with that approach in general is that the designer must select all of the pole locations. For high order systems it may be very difficult to select all the poles in this arbitrary fashion and still maintain reasonable levels of control effort. The root square locus provides a very useful guide in this respect. The designer only needs to select the desired dominant modes, by using a low order model in (7-39), and the poles of the remaining modes are known to approach a Butterworth configuration in a well defined manner. Since the root square locus gain is inversely proportional to the control effort weighting factor in the cost functional, one can use the root square locus gain as a relative measure of the control effort required along the locus. If one arbitrarily selects pole locations for the higher order modes in a Butterworth configuration, or some other configuration known to give well behaved characteristics, the feedback gains may be unnecessarily high causing excessive control effort. By using the root square locus one can quickly judge the trade between obtaining the desired dominant characteristics and excessive control effort. The importance of the simple root locus form resulting from the Model PI concept becomes apparent here. If the designer had to use several root square loci just to obtain the zeros of the final root square locus, as in reference 26, the technique may be of questionable value.

7.3.1 Design of a Pitch Damper for the X-15 Aircraft by Root Square Locus

The simplified pitch damper for the X-15 aircraft considered in Chapter 6 section 6.1, will be redesigned to the same specification using root square locus. A block diagram of the open-loop system is shown in figure 7-2.

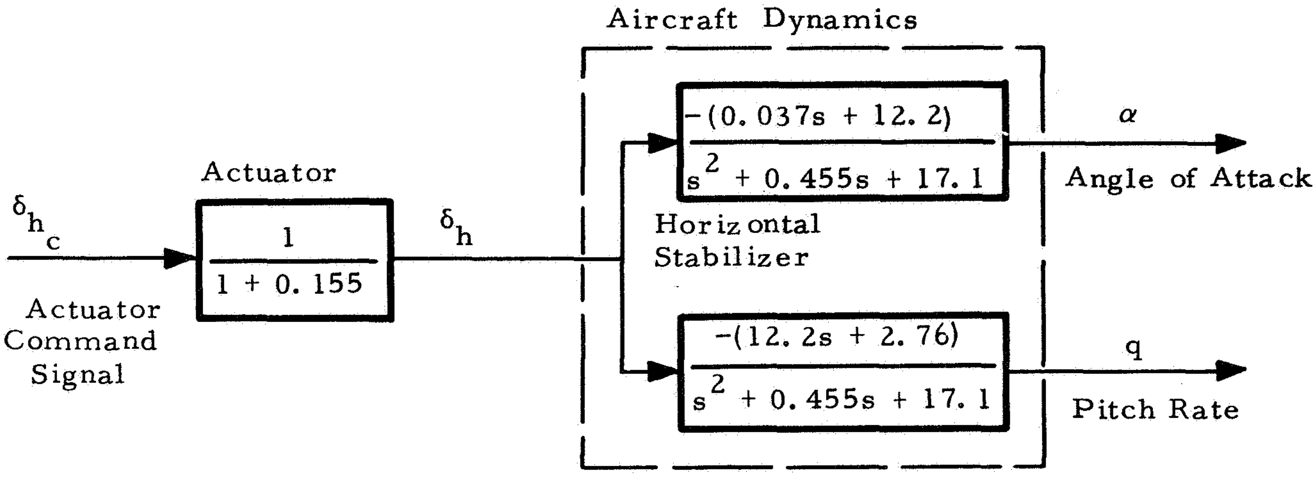


Figure 7-2 Block Diagram of Open-Loop System for Root Square Locus Design Example

The open-loop transfer functions of servo-command, δ_{h_c} to angle of attack, α, and pitch rate, q, are (data from section 6.1)

$$\frac{\alpha(s)}{\delta_{h_c}(s)} = \frac{-6.67(0.037s + 12.2)}{(s + 6.67)(s^2 + 0.455s + 17.1)}$$

$$\frac{q(s)}{\delta_{h_c}(s)} = \frac{-6.67(12.2s + 2.76)}{(s + 6.67)(s^2 + 0.455s + 17.1)} \qquad (7\text{-}103)$$

The design specifications for this example are to obtain dominant second order poles of the closed-loop system lying within the Satisfactory handling qualities region as indicated in figure 6-1.

The root square locus equation for this example can be written directly from (7-103),

$$\frac{\kappa(\alpha_0 - \alpha_1 s + \cdots \pm s^{\ell})(\alpha_0 + \alpha_1 s + \cdots + s^{\ell})}{(-s + 6.67)(s^2 - 0.455 s + 17.1)(s + 6.67)(s^2 + 0.455 s + 17.1)} = -1 \qquad (7\text{-}104)$$

where the model poles, i.e. the zeros of (7-104), are left arbitrary for the moment. Following the procedure outlined above, the poles of (7-104) are plotted as x' on figure 7-3. which includes the handling qualities criteria from figure 6-1. The actual system is indicated in the LHP and its adjoint in the RHP. Only the upper half of the s-plane is shown for convenience. From experience with root locus techniques, one can see that if the zeros of (7-104) are chosen to be in the "Good" region of figure 7-3, the locus must pass through the Satisfactory region (the zero-phase locus is required in this case).

A second order model with poles

$$(s + 4 + j3)(x + 4 - j3) = (s^2 + 8s + 25) \qquad (7\text{-}105)$$

should be a good choice. This makes the root square locus equation

$$\frac{\kappa(s^2 - 8s + 25)(s^2 + 8s + 25)}{(s - 6.67)(s^2 - 0.455s + 17.1)(s + 6.67)(s^2 + 0.455s + 17.1)} = +1 \qquad (7\text{-}106)$$

The zeros of (7-106), which are the poles of the model (7-105) and its adjoint, are indicated on figure 7-3 by $\odot$.

The root locus for (7-106) is shown on figure 7-4, which represents the locus of the closed-loop poles of the " optimal" regulator and its adjoint as a function of $(r\,\|\underline{\tilde{\alpha}}\|^2)^{-1}$. Note that it isn't necessary to write down the cost functional or even $\tilde{Q}$ since these are implied by selecting the model poles (7-105). The root locus is seen to pass through the Satisfactory handling qualities region as anticipated. The designer can choose a position along the root locus for the closed-loop system poles that meets the handling qualities requirement with an acceptable level of control effort. A possible solution is indicated on

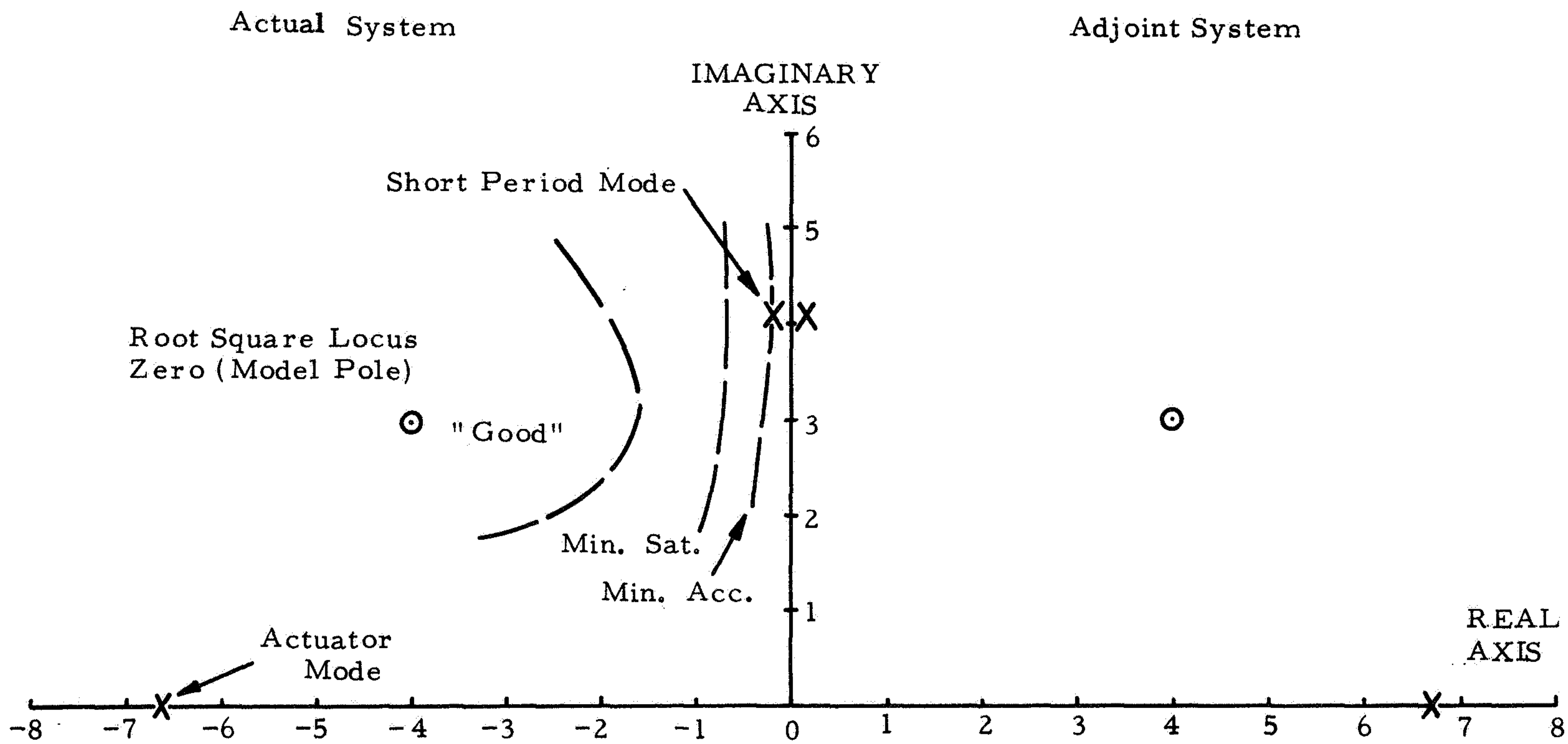


Figure 7-3 Root Square Locus Zeros (Model Poles) Selected Based on Longitudinal Handling Qualities Criteria

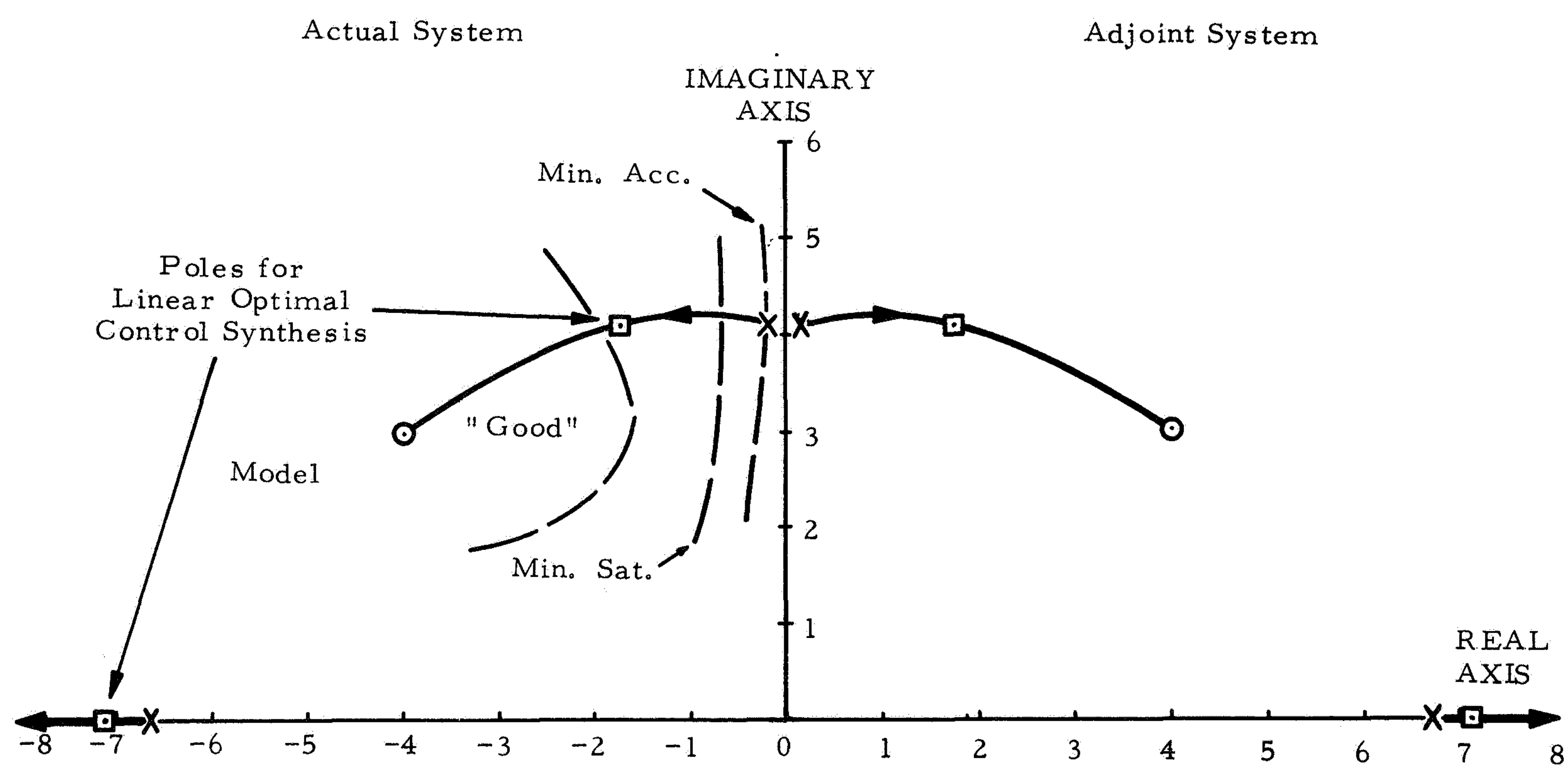


Figure 7-4 Root Square Locus for Optimal Regulator Compared to Longitudinal Handling Qualities Criteria

figure 7-4 by ⊡, which corresponds to a value of $\kappa = 13.2$. The closed-loop poles for this solution are

$$\begin{aligned} s_1 &= -7.1 \\ s_{2,3} &= -1.75 \pm j\,4.17 \end{aligned} \tag{7-107}$$

so that the closed-loop characteristic equation is

$$s^3 + 10.6\,s^2 + 45.2\,s + 145 = 0 \tag{7-108}$$

The open-loop characteristic equation was

$$s^3 + 7.1\,s^2 + 20.1\,s + 114 = 0 \tag{7-109}$$

Relating the coefficients of these two equations to (7-102) it follows that

$$\begin{aligned} k_0 &= 145 - 114 = 31 \\ k_1 &= 45.2 - 20.1 = 25.1 \\ k_2 &= 10.6 - 7.1 = 3.5 \end{aligned} \tag{7-110}$$

These are the feedback gains for the system state (phase) variables, $\underline{x}$, which are related to the variables α, q, and δ_h by

$$\begin{bmatrix} \alpha \\ q \\ \delta_h \end{bmatrix} = H\underline{x} \tag{7-111}$$

where*

$$H = 6.67 \begin{bmatrix} -12.2 & -0.037 & 0 \\ -2.76 & -12.2 & 0 \\ 17.1 & 0.455 & 1 \end{bmatrix} \tag{7-112}$$

---

* The elements of H are obtained directly from the transfer functions (7-103)

Therefore the desired gain vector is obtained from

$$\begin{bmatrix} k_\alpha \\ k_q \\ k_{\delta_h} \end{bmatrix} = (H')^{-1}\underline{k} \tag{7-113}$$

Taking the inverse of H, and performing the multiplication indicated results in

$$\begin{aligned} k_\alpha &= +0.424 \text{ deg } \delta_{h_c}/\text{deg } \alpha \\ k_q &= -0.291 \text{ deg } \delta_{h_c}/\text{deg per sec } q \\ k_{\delta_h} &= +0.525 \text{ deg } \delta_{h_c}/\text{deg } \delta_h \end{aligned} \tag{7-114}$$

If these gains are reasonable from the implementation standpoint the synthesis is complete. The associated block diagram for this solution is shown in figure 7-5.

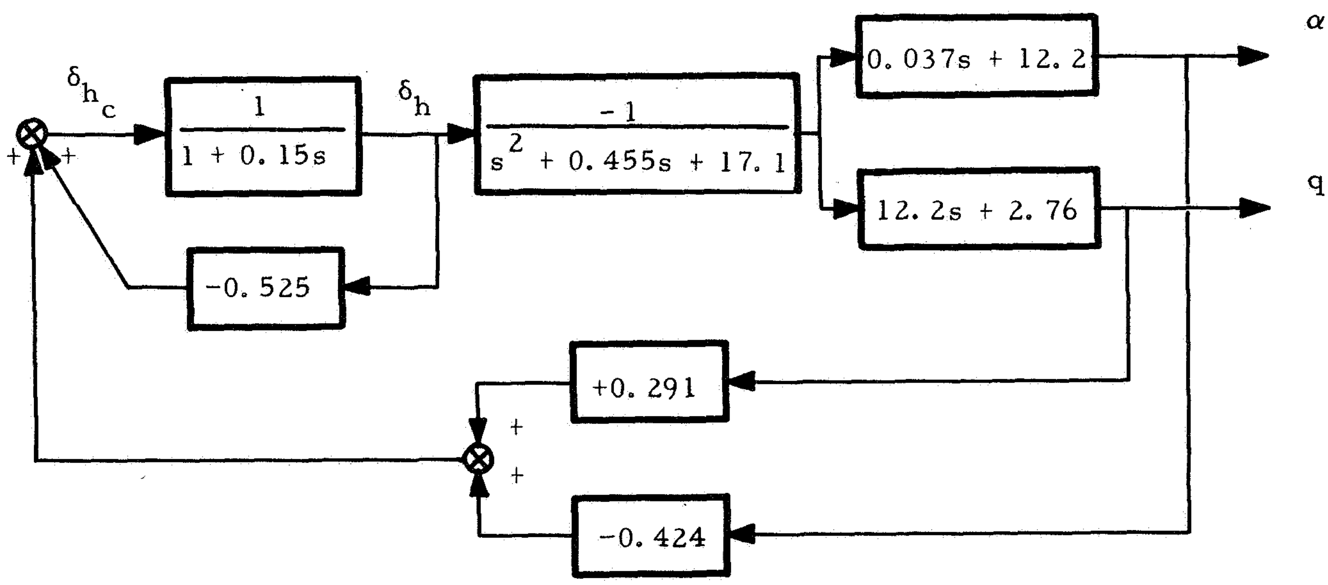


Figure 7-5 Block Diagram of Pitch Damper Designed by Root Square Locus

7.4 Equivalent Model PI for General Quadratic Functionals

Consider an nth order, linear system in canonical (phase-variable) form given by equations (7-2) - (7-5), and a general quadratic cost functional

$$\bar{J} = \frac{1}{2}\int_0^{\infty} [\, \|\underline{x}(t)\|^2_{\bar{Q}} + \bar{r}\, u^2(t)\,]\, dt \qquad (7\text{-}115)$$

where $\bar{Q}$ is any positive, semi-definite symmetric matrix and $\bar{r}$ is a non-negative scalar. Two cost functionals will be called "equivalent" if optimization of the two cost functionals give the same results. It will be shown in this section that the general quadratic cost functional (7-115) is equivalent to one with a diagonal weighting matrix, which in turn, is equivalent to a cost functional of the form (7-1) based on the Model PI concept. A general expression is developed for relating the elements of a general weighting matrix, $\bar{Q}$, to those of a diagonal weighting matrix, Q_d, and to those of the Model PI weighting matrix, $\tilde{Q}$, for equivalent cost functionals. Using this expression it is possible to interpret any general quadratic cost functional in terms of the Model PI concept.

The basic idea used to establish this relationship is a direct extension of that used by Aizerman (18) and Rekasius (19), which was discussed in Chapter 3, section 3.3.1. Their procedure is completely valid here because the optimization process is independent of the system initial conditions. At this point, only the first term in (7-115) needs to be considered. Expanding the integrand in terms of the state (phase) variables*

$$\|\underline{x}\|^2_{\bar{Q}} = \sum_{i=1}^{n} \bar{q}_{ii}\,[x^{(i-1)}]^2 + 2\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \bar{q}_{ij}\, x^{(i-1)} x^{(j-1)} \qquad (7\text{-}116)$$

---

* The time argument is suppressed for convenience when the meaning is clear.

where $\bar{q}_{ij}$ are the elements of $\bar{Q}$, and noting that

$$x^{(j-1)} = \frac{d^{(j-i)}}{dt^{(j-i)}} x^{(i-1)} \tag{7-117}$$

for $j = i,\ i+1,\ i+2, \cdots, n$, when the system is in canonical phase-variable form, one can see that the first term of (7-115) is a linear combination of integrals of the form

$$I = \int_0^\infty \left[x^{(i-1)} \frac{d^{(j-i)}}{dt^{(j-i)}} x^{(i-1)} \right] dt \tag{7-118}$$

This type of integral can be integrated by parts successively to establish the general formula,

$$I = (-1)^k \sum_{k=0}^{\frac{1}{2}(j-i-1)} x^{(k-i-1)} x^{(j-k-2)} \Big]_0^\infty \qquad \text{for } (j-i) \text{ odd} \tag{7-119}$$

and

$$I = (-1)^k \sum_{k=0}^{\frac{1}{2}(j-i-2)} x^{(k-i-1)} x^{(j-k-2)} \Big]_0^\infty + (-1)^{\frac{1}{2}(j-i)} \int_0^\infty \left[x^{(\frac{1}{2}j+\frac{1}{2}i-1)} \right]^2 dt \qquad \text{for } (j-i) \text{ even} \tag{7-120}$$

If the system is asymptotically stable, $\underline{x}(t) \to \underline{0}$ as $t \to \infty$, then (7-119) and the first term of (7-120) are constants that only depend on the system initial conditions Using the above general formula the first term in (7-115) can be reduced to

$$\tfrac{1}{2}\int_0^{\infty} \|\underline{x}\|^2_{\bar{Q}}\, dt = c(\underline{x}_0) + \tfrac{1}{2}\int_0^{\infty} \Bigg\{ \sum_{i=1}^{n} \bar{q}_{ii}\left[x^{(i-1)}\right]^2$$

$$+ 2\nu \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \bar{q}_{ij}(-1)^{\frac{1}{2}(j-i)} \left[x^{(\frac{1}{2}j+\frac{1}{2}i-1)}\right]^2 \Bigg\} dt \qquad (7\text{-}121)$$

where $c(\underline{x}_0)$ is a constant depending on the system initial conditions, and ν is defined as

$$\nu = \begin{cases} 0 & \text{if } (j-i) \text{ is odd} \\ 1 & \text{if } (j-i) \text{ is even} \end{cases} \qquad (7\text{-}122)$$

Consider the results thus far, indicated by (7-121). Only the integral term of (7-121) can affect the optimization process since $c(\underline{x}_0)$ is a constant. The first summation in the integrand contains only the diagonal elements of $\bar{Q}$. The double summation term only contributes terms from the alternate diagonals of $\bar{Q}$ where (j-i) is even. The other alternate diagonal elements of $\bar{Q}$, where (j-i) is odd, can in no way affect the optimization. For example a 4 × 4 matrix of the form

$$\bar{Q} = \begin{bmatrix} q_{11} & 0 & q_{13} & 0 \\ 0 & q_{22} & 0 & q_{24} \\ q_{31} & 0 & q_{33} & 0 \\ 0 & q_{42} & 0 & q_{44} \end{bmatrix} \qquad (7\text{-}123)$$

is a completely general 4 × 4 weighting matrix for a cost functional. Note also that only the squares of state variables appear in (7-121), so that it can be written with a diagonal weighting matrix. If the diagonal elements of a diagonal matrix, Q_d are denoted by $q_{d_{ii}}$ then

$$q_{d_{ii}} = \bar{q}_{ii} + 2 \sum_{\mu=1}^{\eta} (-1)^{\mu} \bar{q}_{(i-\mu)(i+\mu)} \tag{7-124}$$

for $i = 1, 2, 3, \cdots, n$, where

$$\eta = \min \{(i-1), (n-i)\} \tag{7-125}$$

To obtain (7-124) from the integrand of (7-121) requires a transformation of the summation indices of the double sum so that the state variable indices are the same in both summations. It is easy to verify by example that they are the same.

A cost functional

$$J_d = \frac{1}{2} \int_0^{\infty} [\| \underline{x}(t) \|^2_{Q_d} + r\, u^2(t)]\, dt \tag{7-126}$$

where Q_d is a diagonal matrix with elements (7-124), is equivalent to J (7-115) since

$$\bar{J} = c(\underline{x}_0) + J_d \tag{7-127}$$

and $c(\underline{x}_0)$ is a constant. A diagonal weighting matrix can always be formed, using the expression (7-124), that will produce an equivalent quadratic cost functional to the general form (7-115).

It is an easy matter to relate this result to the Model PI form of the cost functional because the Model PI weighting matrix is just a special case of the above development. For this treatment it is convenient to define the (n + 1) vector $\underline{\bar{\alpha}}$ as

$$\underline{\bar{\alpha}}' = [\alpha_0 \, \alpha_1 \cdots \alpha_{n-1} \quad 0] = [\underline{\alpha} \;\vdots\; 0] \tag{7-128}$$

to use in

$$Q = \frac{\underline{\bar{\alpha}}\, \underline{\bar{\alpha}}'}{\| \underline{\bar{\alpha}} \|^2} \tag{7-129}$$

The Model PI cost functional (7-1) can then be written as

$$J = \frac{1}{2} \frac{1}{\|\underline{\alpha}\|^2} \int_0^\infty [\ \|\underline{x}(t)\|^2_{\underline{\alpha}\underline{\alpha}'} + r\|\underline{\alpha}\|^2 u^2(t)\ ]\ dt \qquad (7\text{-}130)$$

which is of the same form as (7-115). The previous results can be used directly to relate the elements of $\underline{\alpha}\underline{\alpha}'$ to the elements of the diagonal weighting matrix for equivalent cost functionals. From (7-124) one gets*

$$q_{d_{ii}} = \alpha^2_{(i-1)} + 2\sum_{\mu=1}^{\eta} (-1)^\mu \alpha_{(i-1-\mu)}\alpha_{(i-1+\mu)} \qquad (7\text{-}131)$$

for $i = 1, 2, \cdots, n$, and η given by (7-125). A general weighting matrix and a Model PI weighting matrix for equivalent cost functionals can be related by equating (7-124) and (7-131), i.e.

$$\boxed{\alpha^2_{(i-1)} + 2\sum_{\mu=1}^{\eta} (-1)^\mu \alpha_{(i-1-\mu)}\alpha_{(i-1+\mu)} = q_{ii} + 2\sum_{\mu=1}^{\eta} (-1)^\mu q_{(i-\mu)(i+\mu)}} \qquad (7\text{-}132)$$

for $i = 1, 2, 3, \cdots, n$, and η given by (7-125).

If the n equations (7-132) can be solved for the n elements of $\underline{\alpha}$ in terms of the elements of $\bar{Q}$, then it is possible to establish a cost functional of the Model PI form (7-1) or (7-130) that is equivalent to the general cost functional (7-115). The general cost functional $\bar{J}$ (7-115) is related to the Model PI cost function J (7-130) by

---

* The indices on α are shifted by -1 because the elements of $\underline{\alpha}$ are the characteristic equation coefficients of the model which are typically index in this manner.

$$\bar{J} = c_1(\underline{x}_0) + \|\underline{\tilde{\alpha}}\|^2 J \tag{7-133}$$

where $c_1(\underline{x}_0)$ is a constant depending on the initial conditions. Since $c_1(x_0)$ and $\|\underline{\tilde{\alpha}}\|^2$ can not affect the optimization process, the two cost functionals are equivalent.

To summarize the results, it is possible to establish a quadratic cost functional with a diagonal weighting matrix, using the expression (7-124), or a quadratic cost functional of the form (7-1) based on the Model PI concept, using the expression (7-132), which is equivalent to the general quadratic cost functional (7-115). If a specific quadratic cost functional is given, then one can determine the $\underline{\alpha}$ from (7-132) that corresponds to a model for that cost functional, in the Model PI concept.

This result can clarify a misconception about "optimal" systems. It is often mentioned (e.g. references 26 - 28 and 55) that the optimal regulator design generally has good response characteristics, meaning fast and well damped. This intuitive conclusion probably developed from the fact that for many of the weighting matrices used in the early optimal regulator treatments resulted in an approximate Butterworth configuration. But it is clear from the results of this chapter that the response of the "optimal" regulator is directly dependent on the state weighting matrix, and one can certainly choose a weighting matrix that would result in an "optimal" regulator with terrible response characteristics. For example, if a weighting matrix were chosen, through ignorance, that corresponded to a model with very lightly damped complex poles, the dominant characteristic of the "optimal" regulator would tend to be very lightly damped for small values of control effort weight factor. It is very important to understand the relationship between the state weighting matrix and the optimal solution. The fact that stability is guaranteed is not enough for a design engineer.

The results of this section can also be applied to parameter optimization problems of the type considered in Chapter 3 <u>if and only if</u> the pseudo initial conditions are independent of the free parameters.

7.4.1 An Example

A simple optimal regulator problem considered in references 27 and 28 with a general cost functional will be interpreted here in terms of the Model PI concept by forming an equivalent Model PI form cost functional. The system considered is the simplified pitch damper for the X-15 aircraft treated in section 7.3.1 (also Chapter 6, section 6.1). The cost functional used in references 27 and 28 is

$$J = \int_0^{\infty} [q_{11}\alpha^2 + q_{22}q^2 + r\,\delta_{h_c}{}^2]\, dt \qquad (7\text{-}134)$$

with $q_{11} = q_{22}$, and $r = 1.0$, where α and q are output variables and δ_{h_c} is the control variable. The output variables are related to the state (phase) variables by

$$\begin{bmatrix} \alpha \\ q \end{bmatrix} = H\underline{x} \qquad (7\text{-}135)$$

where H is a 2×3 matrix given by the first two rows of (7-117), i.e.

$$H = 6.67 \begin{bmatrix} -12.2 & -0.037 & 0 \\ -2.76 & -12.2 & 0 \end{bmatrix} \qquad (7\text{-}136)$$

Then the state vector weighting matrix, Q, for this example is

$$\bar{Q} = H'QH \qquad (7\text{-}137)$$

where

$$Q = \begin{bmatrix} q_{11} & 0 \\ 0 & q_{22} \end{bmatrix} = q_{11} I \qquad (7\text{-}138)$$

or

$$Q = 6630\, q_{11} \begin{bmatrix} 1.05 & 0.229 & 0 \\ 0.229 & 1.0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad (7\text{-}139)$$

As pointed out above, the off diagonal elements in (7-139) do not affect the optimization process. Using the expression (7-132) one can obtain the $\underline{\alpha}$ for the equivalent Model PI cost function as

$$\underline{\alpha}' = \sqrt{6630 q_{11}}\, [1.025 \quad 1.0 \quad 0] \qquad (7\text{-}140)$$

The cost functional (7-134) can then be interpreted as representing a first order model with a pole, $s = -1.025$.

It is interesting to look at the root square locus for this example, which can be written down directly using the simple form (7-39). The system poles were given in section 7.3.1.

$$\frac{6630\, q_{11}\, (s - 1.025)(s + 1.025)}{(s - 6.67)(s^2 - 0.455\, s + 17.1)(s + 6.67)(s^2 + 0.455\, s + 17.1)} = -1 \qquad (7\text{-}141)$$

The root square locus (7-191) plotted on figure 7-6 is identical to that in figure 5 of reference 28 which was obtained by solving the steady-state Riccati equation. The poles of the optimal regulator for $q_{11} = 0.2$, 0.5 and 1.0 are indicated by ⊡. One can clearly see from figure 7-6 that as the root square locus gain increases the dominant characteristic of the optimal regulator approaches the model represented by the weighting matrix. The excess poles are seen to approach a Butterworth configuration. The dominant characteristic is also shown to approach the model by the time response shown in figure 7-7, which is from reference 28, but with the model response added.

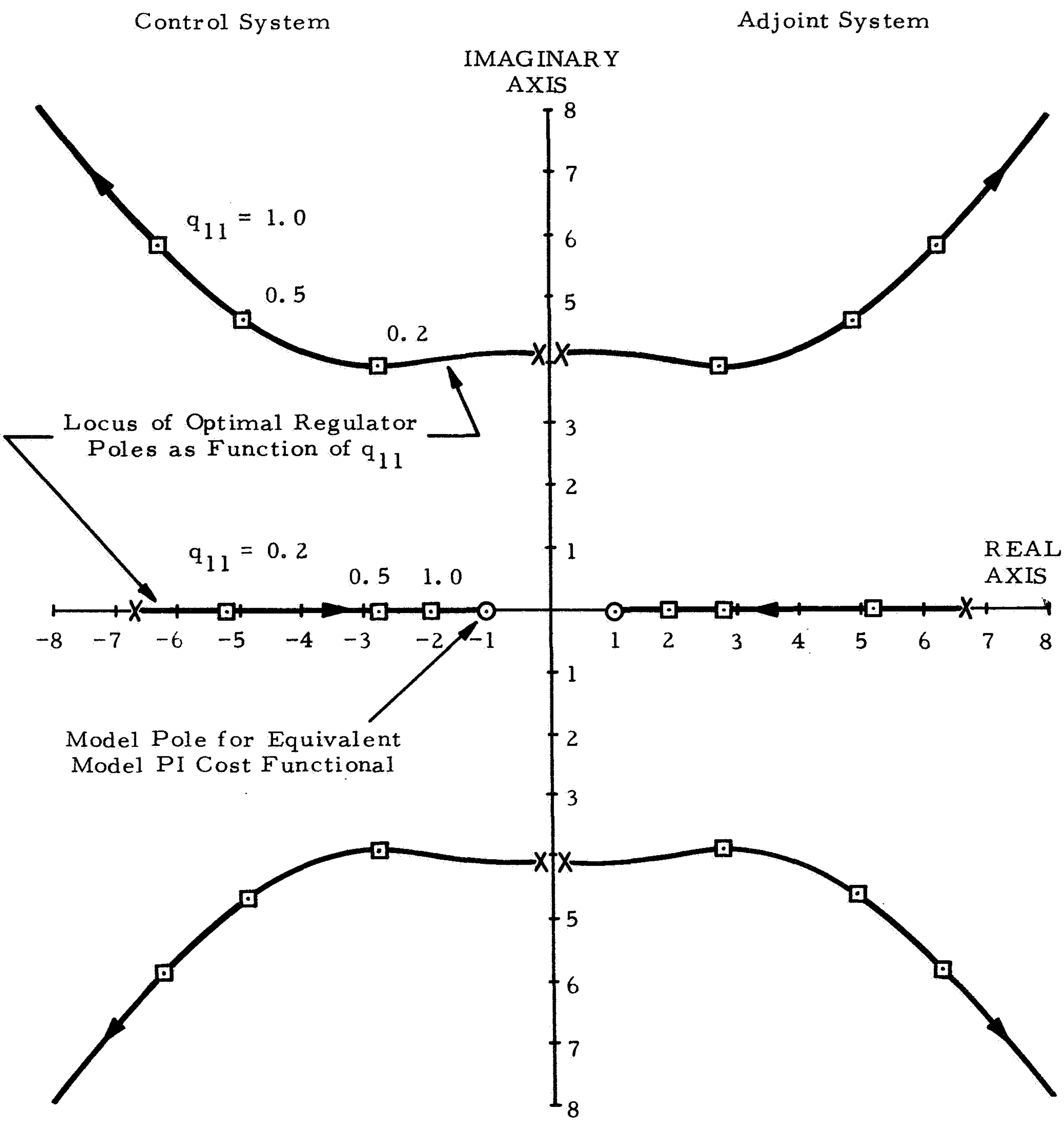


Figure 7-6 Root Square Locus for a Cost Functional Used in References 27 and 28

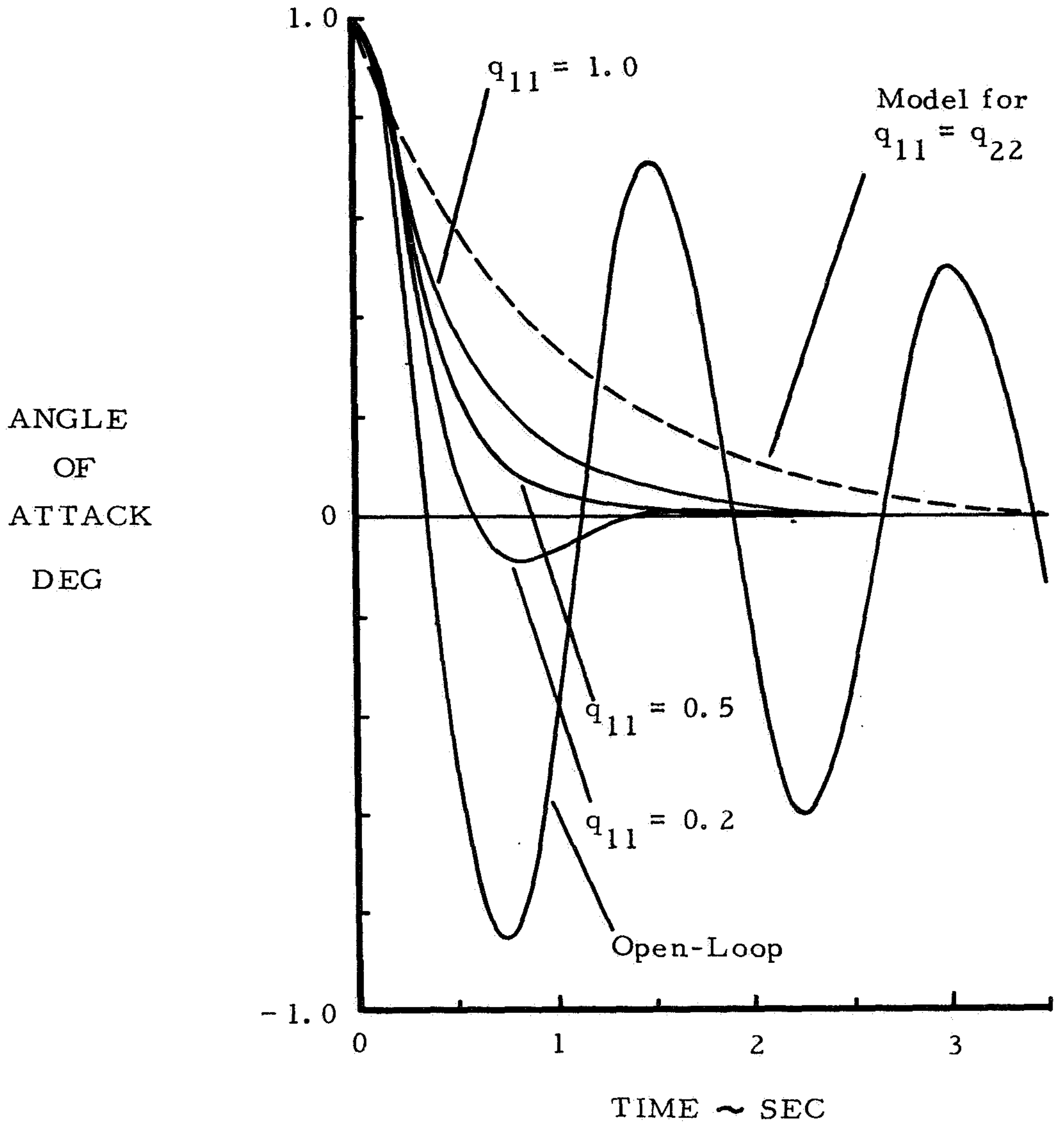


Figure 7-7 Time Response of Pitch Damper Designed in References 27 and 28 Compared to the Model Response Represented by the Cost Functional

CHAPTER 8

CONCLUSIONS
AND
SYNTHESIS OF RESULTS

8.1 Conclusions

This thesis develops a new performance index, the Model PI, that brings engineering design specifications into the analytical design process. Application to flight control systems is emphasized although the techniques apply to linear, time invariant, deterministic systems in general. The basic form of the Model PI is the same as that of quadratic functionals frequently appearing in modern control theory. The important difference is the ability to interpret the weighting matrix of the Model PI directly in terms of a model that relates to engineering specifications. A parameter optimization design procedure is established that starts with practical engineering specifications and uses the Model PI as a synthesis tool to obtain a satisfactory design.

The Model PI is different from the familiar model-referenced integral squared error (ISE) performance index, except in certain special cases. It can be used effectively in designing practical control systems. And, it is substantially more efficient to use than a comparable model-referenced ISE performance index.

Some interesting new developments on the theory and application of linear optimal control have resulted from the Model PI theory. Only the single-control regulator problem is considered. Using a form of the Model PI, it is possible to interpret the state vector weighting matrix in terms of a model which the optimal regulator will approach in a limiting case. The Model PI concept provides an interestingly simple solution to the linear optimal control synthesis procedure using one root square locus. It is possible to compute a Model PI cost functional that is equivalent to a general quadratic functional.

The Model PI represents a new criterion for approximating one dynamical system (the model) by another, based on a novel geometrical representation of linear autonomous systems. It is possible to represent a linear model by a characteristic hyperplane and pseudo initial condition vector in the model's extended state space. A system's transient time response trajectory can be made to approximate that of the model, if the projection of the system's trajectory into the model's extended state space can be made to lie close to the model's characteristic plane, and if the system's and model's pseudo initial condition vectors are close in the model's state space. The criterion represented by the basic form of the Model PI can be thought of as a generalized measure of the distance between the system's trajectory and the model's characteristic plane. The model's characteristic plane enters the Model PI in the definition of the state vector weighting matrix used in a quadratic functional. In some cases, for systems with zeros, it is necessary to include a quadratic penalty on the error between the system's and model's pseudo initial condition vectors. Minimizing the Model PI tends to make the system's trajectory match the model's trajectory in the model's extended state space; in other words, the system's output and its first ℓ derivatives tend to match the corresponding model output variables, where ℓ is the order of the model. If sufficient freedom is provided in the design parameters, minimizing the Model PI will produce exact model matching.

The state space formulation of analytical design problems makes it possible to establish general digital computer programs for designing linear control systems by parameter optimization. Such a program has been established for the Model PI design procedure, which only requires providing the appropriate input data cards and writing one simple subroutine to change from one design problem to another. The numerical optimization procedure uses a technique, derived here, for evaluating the gradient of the performance index directly rather than using a numerical difference procedure.

The Model PI design procedure has been shown to be a practical synthesis technique in three flight control system design examples, in which realistic type design specifications were used.

8.2 Synthesis of Results

Significant results have been obtained in this thesis effort that will be of interest to the practical control systems engineer as well as the modern control theorist. To the practical engineer, the main contribution is the development of a systematic design technique that allows him to introduce engineering design specifications into the process, in the convenient form of a model, and yet implement the technique in a general and efficient manner on a digital computer. Control technology has been developing in the direction of these general goals for several years. Models have been included in previous analytical design techniques, notably in the form of model-referenced integral squared error (ISE) performance indices. The introduction of state space notation into control system analysis by Kalman has provided a convenient means of writing general computer programs for control system design and analysis. The technology for writing general computer programs for designing linear control systems by parameter optimization using a model-referenced ISE performance index has existed for some time now, but there is little indication in the literature that such programs exist. One problem that has hindered the application of parameter optimization techniques to practical system design, is that for even moderately high order systems and models the computational task can become enormous. However the Model PI design procedure has a potential of reducing the computation time by 20% to possibly 85% of that required for the model-referenced ISE procedure, which should stimulate a renewed interest in parameter optimization design techniques.

Although the Model PI method is of general applicability, it is of particular interest for flight control system design. An analytical design technique must include a model in some manner to be effective in most flight control system design problems because of the specific response characteristics needed for a satisfactory design. Also the design of a flight control system involves applying the design procedure numerous times because of the wide range of flight conditions, weight, center of gravity locations, and flight configurations (flaps, landing gear, speed brakes, external stores, etc.). In such a large design effort the computational efficiency of an analytical design technique becomes an important factor and a potential savings of 20% to 85% offered by the Model PI method is quite impressive.

The general computer program developed here for the Model PI method has already proved to be an effective synthesis tool even though it is only a prototype and can be improved considerably. Some possible improvements and extensions will be discussed subsequently. As with any new technique, one must apply it to many types of design problems, more than was possible in this thesis effort, to assess its true potential and its limitation. But it seems quite likely that it can be developed into an efficient, general design program for linear systems that would perform virtually all of the tedious work of the design process.

One improvement needed to meet this desirable objective is in the process of forming the closed-loop system transfer function (or functions) in terms of the free design parameters or doing away with this step completely. For multivariable systems this can become a tedious task. This step in the design process is common to all of the parameter optimization procedures, regardless of the performance index used, when implemented on a digital computer. A possible approach is to establish certain standard feedback configurations that are easily incorporated into the necessary closed-loop transfer function. Another possible approach is to transform the open-loop system into the canonical phase-variable form and redefine the free parameters, as functions of the original parameters, such that they are only gains of state (phase) variables. However both of these may place undesirable constraints on the design procedure.

Another area for improvement in the general computer program is a procedure for establishing a good initial choice for the free design parameters. Since most efficient optimization algorithms find only local minima, the initial choice of free parameters is important. A possible approach that appears rather interesting, is to use the linear optimal regulator solution as a first step in the computer program, then switch to the parameter optimization procedure. To do this one would have to formulate the problem such that certain state variable feedback gains in the optimal regulator problem correspond to the free parameters of the parameter optimization problem. Assuming this can be done without too many restrictions, one would compute the optimal feedback gains using the Model PI (as discussed in Chapter 7), then use the values of those gains corresponding to the free parameters as the initial choice in the parameter optimization procedure. This assumes that one does not wish to feedback all state variables in the final design. If this is a feasible approach it would combine some of the best features of the two design methods. In the optimal-regulator problem the gains are very easily calculated, but the solution requires feedback of all state variables. Whereas, in the parameter optimization method finding the "optimum" gains is harder, particularly for a poor initial choice, but the designer can specify the feedback configuration.

There are several results of this thesis effort that are of basic interest in control theory. The Model PI represents a new and different criterion for control system design, although it has certain characteristics in common with the performance indices considered by Aizerman (18) and Rekasius (19). The rigorous treatment of the Model PI theory has not only provided it with a solid foundation, but has also pointed out some significant limitations to Aizerman's and Rekasius' performance indices and some erroneous results in the case of the latter. These are discussed in Chapter 3, section 3.3. Although the Model PI criterion is established in a somewhat abstract mathematical sense, it is just as valid a criterion as the generalized integral squared error. When exact model matching is possible both criteria would produce it. When exact model matching isn't possible with the particular free parameters available, both criteria are measures of the deviation from the condition of exact model matching. The only meaningful basis for judging these criteria is whether they can be used effectively in a design procedure to meet the actual design specifications. On this basis, the Model PI has been shown to be a valid criterion.

The geometrical representation of linear autonomous systems, used to derive the Model PI, is a novel concept that should interest modern control theorists and stimulate considerable future activity. Even though it does not present any new information about linear systems, it is an interesting way of interpreting their transient response trajectories. Using the concept of a system and model being defined by their respective characteristic planes and pseudo initial condition vectors one might develop other criterion for approximating one system by another. For example, a simple, but probably very crude, criterion would be to make the two characteristic planes as close together as possible, i.e. minimize the angle between the two extended coefficient vectors. Or an even simpler and cruder criterion would be to minimize the vector error between the two coefficient vectors.

This geometrical property may provide some interesting interpretations in nonlinear control also. For example, using the Model PI in the optimal "bang-bang" control problem would still define a model's characteristic plane going through the origin of the extended state space; the system given by

$$\underline{\tilde{x}}'(t)\,\underline{\tilde{a}} = u(t) \tag{8-1}$$

where u(t) can take on values of ±1only, thus

$$\underline{\tilde{x}}'(t)\,\underline{\tilde{a}} = +1 \tag{8-2}$$

or

$$\underline{\tilde{x}}'(t)\,\underline{\tilde{a}} = -1 \tag{8-3}$$

which define planes in the extended state space normal to $\underline{\tilde{a}}$ and translated one unit from the origin along $\underline{\tilde{a}}$; and the optimal "bang-bang" solution could be interpreted as choosing the switching from the plane (8-2) to the plane (8-3) that kept the system's trajectory the closest to the model's plane. One might also give a geometrical interpretation to the singular control problem and other special problems in optimal control theory.

Only the simplest of linear optimal control problems, i. e. the single-control optimal regulator problem was considered here. A logical basis for selecting the state vector weighting matrix results from using the Model PI concept. It is shown rigorously that the optimal solution approaches the model represented by the Model PI weighting matrix as the weighting on the control effort in the cost functional goes to zero. The arbitrariness of selecting the weighting matrix has been a weak point in the theory up to now. Extending this result to the multiple control case should be the next step.

APPENDIX A

GENERAL MODEL PI DESIGN FOR SECOND ORDER SYSTEMS

It is possible to obtain a general analytical solution to the Model PI design of second order systems for various models and various free system design parameters. The purpose of presenting this solution is essentially academic and theoretical. It demonstrates the use of the Model PI for the simplest possible examples and the procedure for obtaining an analytical solution when such is practical. It also provides a preliminary insight into the nature of system designs obtained using the Model PI method. From a theoretical standpoint, one is ostensibly required to exhibit general solutions, no matter how trivial, whenever they exist.

A. 1 Problem Statement

Consider the general second order autonomous system

$$\underline{\dot{x}} = F\,\underline{x} \tag{A-1}$$

where

$$F = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix}$$

and with initial condition

$$\underline{x}_0' = [\,-1 \quad 0\,]$$

The extended state vector, $\tilde{\underline{x}}$, is related to the state vector, $\underline{x}$, by

$$\tilde{\underline{x}} = M \underline{x} \tag{A-2}$$

where

$$M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -a_0 & -a_1 \end{bmatrix}.$$

The coefficients a_0 and a_1 are functions of free design parameters.

The Model PI

$$PI = \int_0^{\infty} \| \tilde{\underline{x}} \|^2_{\tilde{Q}} \, dt \tag{A-3}$$

with $\tilde{Q} = \dfrac{\tilde{\underline{\alpha}} \, \tilde{\underline{\alpha}}'}{\| \tilde{\underline{\alpha}} \|^2}$ and

$$\tilde{\underline{\alpha}}' = [\alpha_0 \quad \alpha_1 \quad \alpha_2] \tag{A-4}$$

can be used to represent arbitrary models of zero, first, and second order. A zero order model is one which has a unity transfer function, i. e. the output is identically equal to the input. The Model PI in that case reduces to the familiar integral squared error (ISE) criterion.

The general Model PI design problem for second order systems is to select a_0 and a_1, via the free parameters, that minimize (A-3) as a function of α_0, α_1, and α_2.

A. 2 General Solution

It is relatively simple to solve analytically for the minimum point of the Model PI for second order systems. The PI can be evaluated from (see Chapter 4)

$$PI = \underline{x}_0' \, P \, \underline{x}_0 \tag{A-5}$$

where P is a 2×2 symmetric matrix that is the solution of

$$F'P + PF = - M'\tilde{Q}M \tag{A-6}$$

Equation (A-6) represents only three linearly independent equations for the three independent elements of P. Solving for P and substituting into (A-5) gives

$$PI = \frac{1}{\|\underline{\tilde{\alpha}}\|^2} \left(- \alpha_0 \alpha_1 + \frac{a_1 \alpha_0^2}{2a_0} + \frac{\alpha_0^2 + a_0 \alpha_1^2 + a_0^2 \alpha_2^2 - 2a_0 \alpha_0 \alpha_2}{2a_1} \right) \tag{A-7}$$

In order to proceed from this point one must know the functional dependence of a_0 and a_1 on the free design parameters. Three cases are treated which cover a wide variety of second order system design problems. Any others can be easily solved by a similar procedure starting with (A-7) and the specific functional relationships for the free parameters. The cases considered are

Case 1. a_1 is a fixed, known constant and a_0 is a free design parameter

Case 2. a_0 is a fixed, known constant and a_1 is a free design parameter

Case 3. a_0 and a_1 are linear functions of a free design parameter, k.

A. 2. 1 Case 1. a_1-Fixed and a_0-Free

The necessary condition for a minimum point of PI is

$$\frac{\partial PI}{\partial a_0} = \frac{1}{\|\underline{\tilde{\alpha}}\|^2} \left(- \frac{a_1 \alpha_0^2}{2a_0^2} + \frac{\alpha_1^2 + 2a_0 \alpha_2^2 - 2\alpha_0 \alpha_2}{2a_1} \right) = 0 \tag{A-8}$$

The solution of (A-8) is considered for the three types of models: zero, first, and second order.

Case la. Zero Order Model (same as the ISE criterion)

A zero order model corresponds to $\alpha_1 = \alpha_2 = 0$ and α_0 arbitrary in the Model PI. For this case (A-8) reduces to

$$\frac{a_1 \alpha_0^2}{a_0^2} = 0 \tag{A-9}$$

which requires $a_0 \rightarrow \infty$. The second order system design corresponding to this solution would have a infinitely large frequency with nearly zero damping.

Case lb. First Order Model

A first order model is given by $\alpha_2 = 0$ and α_1 and α_0 nonzero constants in the Model PI. The α_1 can be taken to be unit without loss of generality. The solution of (A-8) for this case is simply

$$a_0 = \alpha_0 a_1 \tag{A-10}$$

The characteristic equation of the second order system design for this solution is

$$s^2 + a_1 s + \alpha_0 a_1 = 0 \tag{A-11}$$

as a function of the known constant a_1.

Equation (A-11) is the general Model PI design for any first order model with a_0 as a free design parameter. The corresponding poles are plotted on figure A-1 in the s-plane scaled by α_0 as a function of a_1. One can see from figure A-1 that as $a_1 \rightarrow \infty$ the dominant pole approaches the model pole.

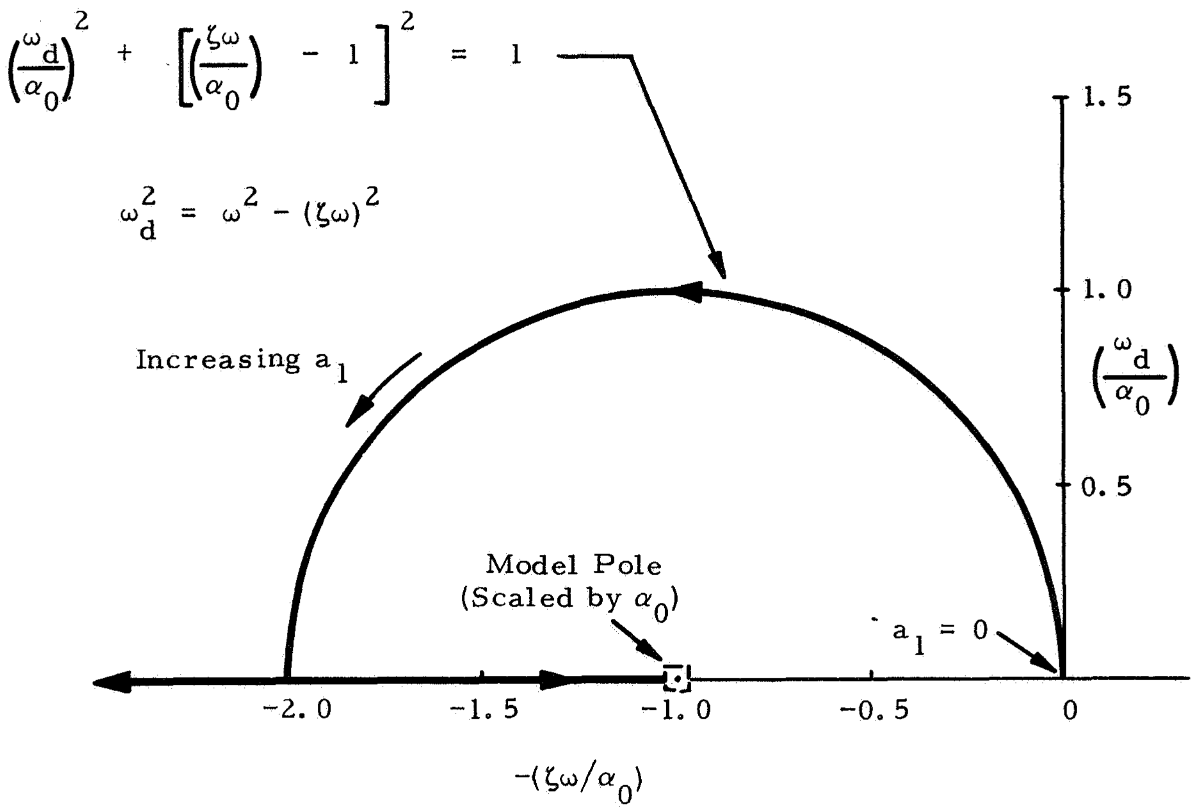


Figure A-1 Locus of Poles for Model PI Design for Case 1b as a Function of a_1 (First Order Model)

<u>Case 1c. Second Order Model</u>

A second order model corresponds to α_0 and α_2 being nonzero and α_1 being arbitrary in the Model PI. The α_2 can be taken to be unity without loss of generality. For this case equation (A-8) can be written as

$$2a_0^3 + (\alpha_1^2 - 2\alpha_0)\, a_0^2 + \alpha_0^2 a_1^2 = 0 \qquad \text{(A-12)}$$

Not much can be done to solve this cubic in general. For any specific example it is easily solved numerically to give the Model PI design. There is an interesting special case when the model has a frequency of one radian per second and a damping ratio of $1/\sqrt{2}$ in which a general solution is possible. In that case $\alpha_0 = 1$ and $\alpha_1^2 = 2\alpha_0 = 2$, so that (A-12) gives the solution

$$a_0 = (\tfrac{1}{2}a_1^2)^{\frac{1}{3}} \qquad \text{(A-13)}$$

The system characteristic equation for this solution is

$$s^2 + a_1 s + (\tfrac{1}{2}a_1^2)^{\frac{1}{3}} = 0 \qquad \text{(A-14)}$$

The corresponding poles have been computed as a function of a_1 and are plotted on figure A-2. For one specific value of a_1 ($a_1 = \sqrt{2}$) the

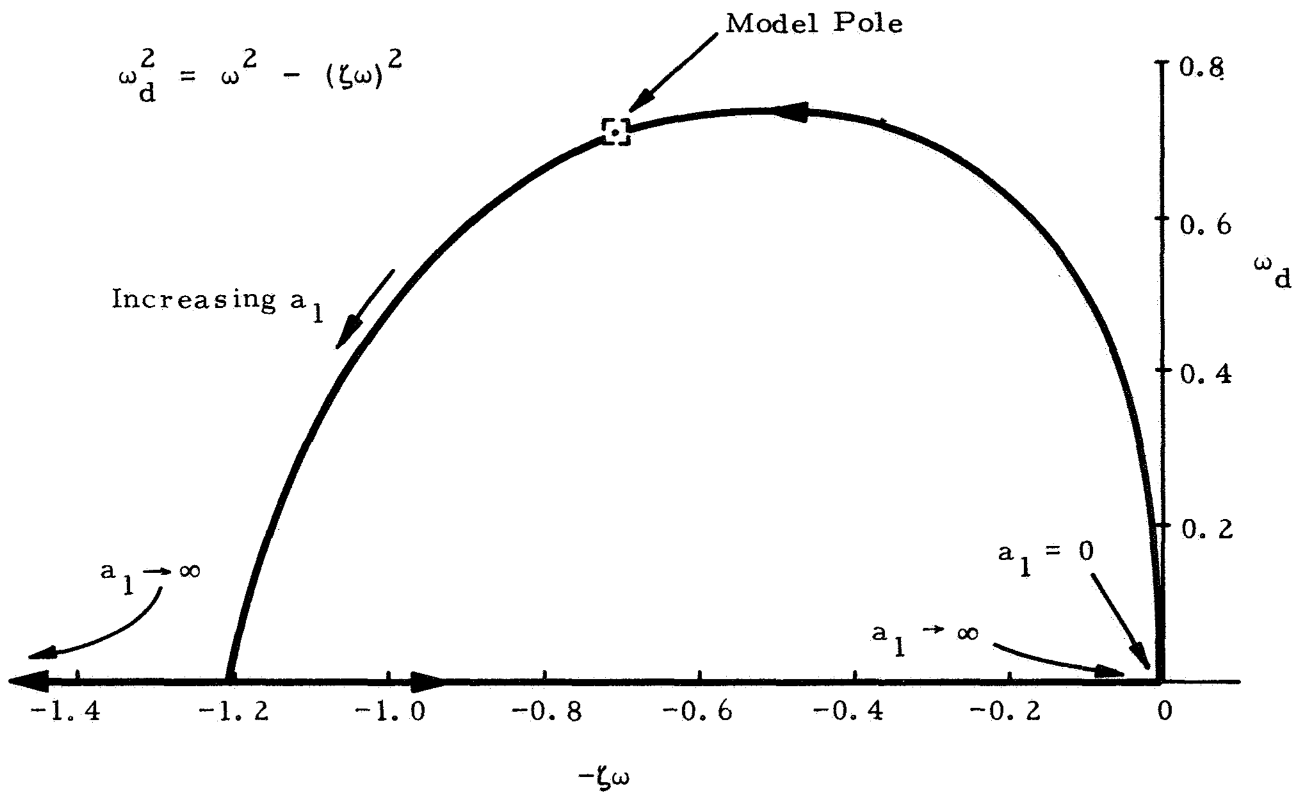


Figure A-2 Locus of Poles for Model PI Design for Case 1c as a Function of a_1 (Second Order Model) system and model poles are identical. Figure A-2 shows that there are ranges of a_1 values for which the Model PI solution produces poles substantially different from those of the model, however with only a_0 as a free design parameter this is bound to occur no matter what design procedure used. In such a case the design problem would be poorly posed.

A. 2. 2 Case 2. a_0-Fixed and a_1-Free

The necessary condition for a minimum point of the PI given by (A-7) for this case is

$$\frac{\partial PI}{\partial a_1} = \frac{1}{\|\underline{\alpha}\|^2}\left(\frac{\alpha_0^2}{2a_0} - \frac{\alpha_0^2 + a_0\alpha_1^2 + a_0^2\alpha_2^2 - 2a_0\alpha_0\alpha_2}{2a_1^2}\right) = 0 \qquad \text{(A-15)}$$

Again the solution is considered for the three types of models: zero, first, and second order.

Case 2a. Zero Order Model (same as the ISE criterion)

For this case ($\alpha_1 = \alpha_2 = 0$ and α_0 arbitrary) equation (A-15) gives the solution

$$a_1 = \sqrt{a_0} \qquad \text{(A-16)}$$

The corresponding system characteristic equation is

$$s^2 + \sqrt{a_0} \qquad \text{(A-17)}$$

which is the well known solution to the ISE criterion (16, 17, 42) for a second order system. The damping ratio is 0.5 for all values of a_0.

Case 2b. First Order Model

For a first order model ($\alpha_2 = 0$, $\alpha_1 = 1$) equation (A-15) has a general solution

$$a_1 = \sqrt{a_1 + (a_0^2 / \alpha_0^2)} \qquad \text{(A-18)}$$

The system characteristic equation for this solution is

$$s^2 + \sqrt{a_0 + (a_0^2 / \alpha_0^2)}\; s + a_0 = 0 \qquad \text{(A-19)}$$

Equation (A-19) is the general Model PI design for second order systems with a_1 a free design parameter and for any first order model. The corresponding pole locations are plotted on figure A-3 as a function of the known constant a_0. The s-plane is again scaled by α_0 to make the graph independent of the model time constant. As $a_0 \to \infty$ the Model PI solution has a dominant pole approaching the model pole.

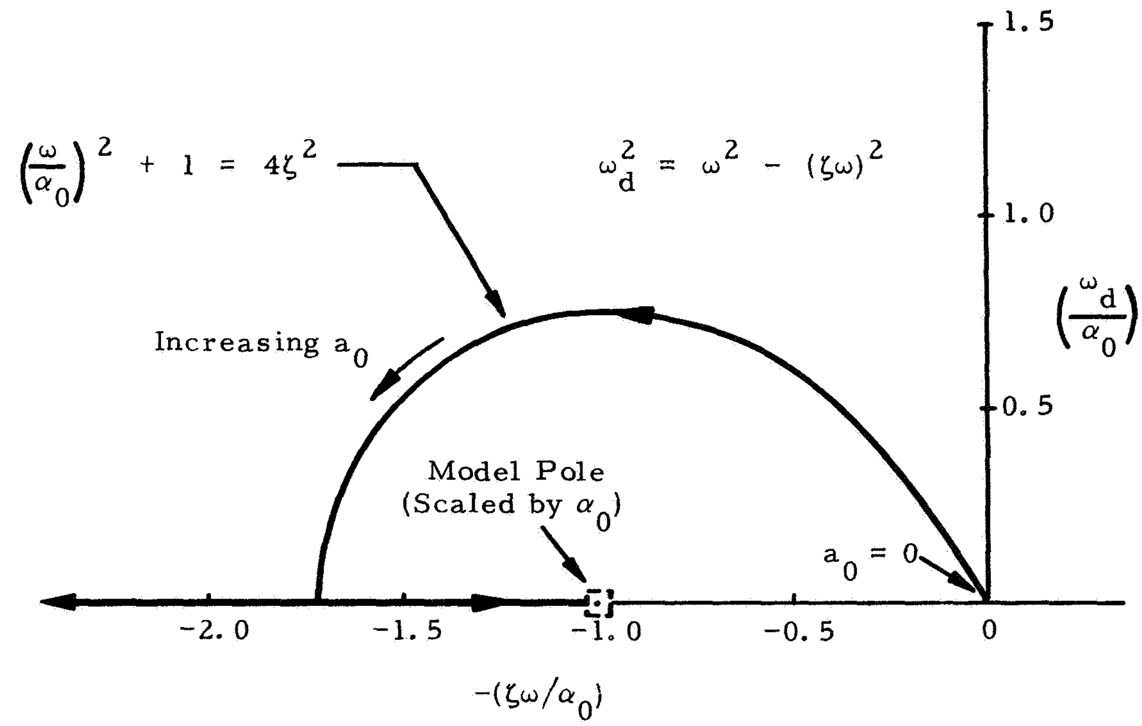


Figure A-3 Locus of Poles for Model PI Design for Case 2b as a Function of a_0 (First Order Model)

Case 2c. Second Order Model

For a second order model equation (A-15) has a general solution (with $\alpha_2 = 1$)

$$a_1 = \sqrt{a_0 (1 + a_0^2 / \alpha_0^2) + (\alpha_1^2 - 2\alpha_0) a_0^2 / \alpha_0^2} \qquad \text{(A-20)}$$

This gives a system characteristic equation

$$s^2 + \sqrt{a_0 (1 + a_0^2 / \alpha_0^2) + (\alpha_1^2 - 2\alpha_0) a_0^2 / \alpha_0^2}\; s + a_0 = 0 \qquad \text{(A-21)}$$

which is the general Model PI design for any second order model with a_1 a free parameter. Recall that it was not possible to obtain the completely general solution in case 1c. because of the cubic equation for a_0. In this case the solution is completely general.

It is still interesting to consider the special model used in case 1c. in order to present a graph of the pole locations for a one parameter family of solutions. For that model ($\alpha_0 = 1$, $\alpha_1^2 = 2\alpha_0 = 2$) the characteristic equation is

$$s^2 + \sqrt{a_0 + a_0^3}\; s + a_0 = 0 \qquad \text{(A-22)}$$

The corresponding poles are plotted on figure A-4 as a function of a_0. For $a_0 = 1$ the system and model poles are identical. The results are quite similar to those of case 1c, and similar comments hold here but with the roles of a_0 and a_1 reversed.

A. 2. 3 Case 3. a_0 and a_1 Linear Functions of a Free Parameter

The coefficients a_0 and a_1 may not be free design parameters themselves but rather functions of a free parameter. The most likely functional relationships to occur in a second order system design are linear functions. It is completely general to take

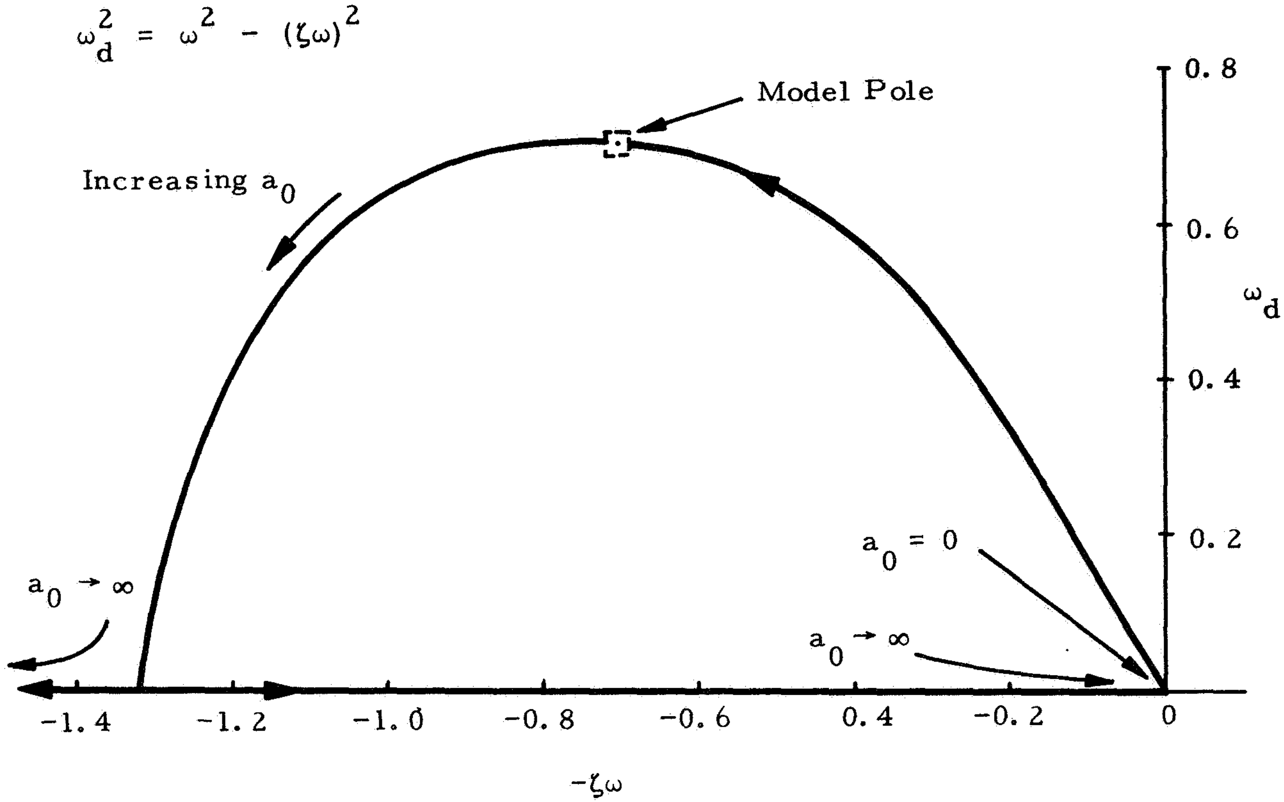


Figure A-4 Locus of Poles for Model PI Design for Case 2c as a Function of a_0 (Second Order Model)

$$a_0 = k$$
$$a_1 = c_1 + c_2 k \tag{A-23}$$

as the linear functions of the free parameter k.

The necessary condition for a minimum of the Model PI is

$$\frac{\partial PI}{\partial a_0} \cdot \frac{\partial a_0}{\partial k} + \frac{\partial PI}{\partial a_1} \cdot \frac{\partial a_1}{\partial k} = 0 \tag{A-24}$$

where the partial derivatives of PI are given by the expressions in (A-8) and (A-15) respectively. Performing the indicated operations in (A-24) and substituting in equation (A-23) gives

$$\frac{1}{\|\underline{\tilde{\alpha}}\|^2}\left[-\frac{(c_1 + c_2k)\,\alpha_0^2}{2k^2} + \frac{\alpha_1^2 + 2k\alpha_2^2 - 2\alpha_0\alpha_2}{2(c_1 + c_2k)}\right]$$

$$+ \frac{1}{\|\underline{\tilde{\alpha}}\|^2}\left[\frac{\alpha_0^2}{2k} - \frac{\alpha_0^2 + k\alpha_1^2 + k^2\alpha_2^2 - 2k\alpha_0\alpha_2}{2(c_1 + c_2k)^2}\right]c_2 = 0 \qquad \text{(A-25)}$$

which can be reduced to the quartic

$$(c_2\alpha_2^2)\,k^4 + (2c_1\alpha_2^2)\,k^3 + \left[c_1(\alpha_1^2 - 2\alpha_0\alpha_2) - (c_1c_2^2 + c_2)\alpha_0^2\right]k^2$$

$$-\,(2c_1^2c_2\alpha_0^2)\,k - c_1^3\alpha_0^2 = 0 \qquad \text{(A-26)}$$

For zero and first order models in the Model PI (both require $\alpha_2 = 0$) equation (A-26) reduces to a quadratic so that general analytical solutions for k as a function of c_1 and c_2 are possible. There appears to be little value in writing out these expressions and presenting graphs of the resulting two parameter family of solutions (for c_1 and c_2). Numerical solutions for specific cases can easily be computed from (A-26). A second order model requires solving the quartic which means only a numerical solution is possible.

APPENDIX B

A GENERAL DIGITAL COMPUTER PROGRAM FOR CONTROL SYSTEM DESIGN VIA PARAMETER OPTIMIZATION

A general computer program is described and listed in this appendix that was developed for designing linear, time invariant control systems via parameter optimization. Although it was developed for the Model PI, it can be used with any quadratic functional as the performance index. Specifically, its use with a model-referenced integral squared (ISE) performance index is described. The contents of Appendix B are listed below.

The program is written in FORTRAN - IV for an IBM 360 computer and uses several subroutines from IBM's Scientific Subroutine Package (SSP) for matrix operations. The compressed storage mode of SSP (see reference 47 for details) is used throughout this program. The description presented here is sufficient to use the program.

B. 1 Description of Main Program and Its Use

The main program, called An Averaged Gradient Direction Optimization Algorithm for Control System Design, computes a sequence of parameter values that approaches a local minimum point of the performance index over the parameter space. How rapid it converges and how close it comes to the minimum depend on certain characteristics of the specific problem that the user has little control over plus certain constants of the computer program that the user specifies. The algorithm used is discussed in Chapter 4, section 4. 4. It evaluates the performance index from equation (4-45a) and its gradient from (4-45b) for the current parameter values and provides a logic for stepping to a new set of parameter values. It includes a step size control and stopping conditions. A simplified functional flow diagram of the algorithm is presented in Chapter 4.

Six special subroutines are required for the main program. Three of these, MRDES, SNORM, and DIRCOS, are general subroutines that are never changed from problem to problem. One subroutine, CALSYS, must be changed for different classes of problems. Three versions are presented that cover the classes of interest here. The subroutine COEF is used in CALSYS MOD 1. It presents the basic data for the specific design problem and therefore a new COEF is written for each problem. The versions of CALSYS denoted MOD 2. and MOD 3. require two subroutines similar to COEF and is discussed in sections B. 2 and B. 3. The last subroutine, DELPC, is used to include parameter constraints if desired.

The subroutines MRDES is used to compute the steady-state solutions of equations (4-45c) and (4-45d). It was written to compute the continuous solution of the stationary matrix Riccati differential equation that occurs frequently in modern control theory and is a modification of a procedure used by Athans and Levine (48). Its general use is described in the listing starting on page 316. A code is included that makes MRDES return only the steady-state solution to the main program. Four constants must be specified by the user (part of the input data to main program) for using MRDES: the time increment for the numerical integration, denoted H; an upper limit on the number of time increments to integrate to in case the tests of steady-state are not satisfied, denoted IMAX; and two numbers, denoted SNE and YNE, used in the tests for steady-state. SNE is the maximum allowable magnitude of the norm* of the derivative of the matrix for the solution to be considered the steady-state value. YNE is the per cent allowable error in the norm of the matrix at steady-state.

---

* The norm used throughout this program is the sup-norm, i. e. the magnitude of the numerically largest element of the matrix or vector.

The subroutines SNORM and DIRCOS are described in the respective listings on pages 330 and 331.

The user must be sure to include the right subroutine CALSYS for his problem. MOD 1. is for the general Model PI for single input/output systems or multivariable systems treated as several single input/output systems. It computes all the quantities in equation (4-45) that are explicit functions of the free parameters. The functional dependence of these quantities on the free parameters is only through the system transfer function numerator and denominator coefficients which are computed in the subroutine COEF. Therefore CALSYS MOD 1. does not depend on the specific system being designed. The subroutine COEF and the data input cards, which are discussed subsequently, are the only items that depend on the specific design problem, unless parameter constraints are used. COEF is a simple listing of the numerator and denominator coefficients as a function of the free parameters. An example is presented in the subroutine listings (page 341).

The subroutine DELPC evaluates the contributions to the performance index and its gradient due to the parameter constraints, if such are included in the design problem. A constant, RPC, specified in the input data, is used to set the relative weighting of the parameter constraint contributions to the other contributions. No general form for DELPC is given here because of the many possible forms one might want to use. An example of DELPC for a quadratic penalty on the parameters is presented in the subroutine listings on page 341. In order to avoid needless computations when no constraints are required, set RPC = -0. 0 in the input data. However one must still include some DELPC subroutine in the total package.

The total computer program package consists of

Main Program
MRDES
SNORM
DIRCOS
CALSYS MOD 1.
COEF
DELPC

B.1.1 Input Data Format

The program is dimensioned for up to a 20th order system with up to 10 free design parameters. The model can also be of up to 20th order. Within these restrictions the program can be used for the design of any system by the Model PI method. The necessary input data cards for a specific problem are indicated below.

First Card - Format Statement No. 3

N, M, K, STEP, ITMAX, H, IMAX, SNE, YNE, RXO, RPC, LI

where

N	is the order of the system's denominator ($N \leq 20$).
M	is the order of the system's numerator ($M \leq N-1$).
K	is the number of free design parameters ($K \leq 10$).
STEP	is the initial step size for the optimization algorithm.
ITMAX	is an upper limit on the number of steps to be computed.
H	is the time increment for numerical integration in MRDES (H should be about 1/10 the smallest characteristic time of the system).
IMAX	is the maximum number of time increments to integrate to in MRDES.
SNE	is the maximum allowable value of the norm of the derivative of the matrix solution at steady-state in MRDES.
YNE	is the maximum allowable error in the norm of the matrix solution at steady-state in MRDES.
RXO	is the relative weighting of the quadratic penalty on the pseudo IC's in the general Model PI.
RPC	is the relative weighting of the parameter constraints in the performance index.
LI	is the order of the model plus one ($LI \leq N+1$).

Next Card(s) - Format Statement No. 24

PAR(I), I = 1, K

where PAR(I) is a vector whose elements are the free design parameters. An initial choice for PAR(I) is required here as input data.

Next Card(s) - Format Statement No. 24

XMO(I), I = 1, N

where XMO(I) is the model's pseudo IC vector.

Next Card(s) - Format Statement No. 24

W(I), I = 1, N

where W(I) is a vector whose elements are the diagonal elements of the pseudo IC weighting matrix (see equation (4-12)).

Next Card(s) - Format Statement No. 24

ALPHA(I), I = I, LI

where ALPHA(I) is the model's coefficient vector in the system's extended state space $\underline{\alpha}$ (see equation (3-26), divided by $\|\underline{\alpha}\|$.

B. 1. 2 Program Output

The program prints out STEP, PAR(I), F(I), XO(I), PI and DPI each optimization step (iteration), where F(I) is the last row of the system coefficient matrix, XO(I) is the system's pseudo IC vector, PI is the value of the performance index, and DPI is the change in the performance index from that at the previous value of PAR(I). If the program terminates it prints out the number of optimization steps taken, the current step size, PAR(I), PI, XO(I), and the final system numerator and denominator coefficient values. The program can terminate if any one of the following occur:

1. stopping conditions satisfied;
2. the upper limit on the number of steps to be computed, ITMAX, is reached, or
3. the stopping condition is bypassed because PI < .1 exp-10.

The reason for terminating is also printed.

B. 2 Procedure for the Weighted Averaged Pseudo IC Method for Multivariable Systems

This method for treating multivariable systems is discussed in Chapter 3, section 3. 2. 3. 2, titled "Second Method". The procedure for using the program with this method is the same as that discussed in B. 1 with three exceptions. First CALSYS MOD 2. must be used instead of MOD 1. Secondly, CALSYS MOD 2. requires two subroutines, COEF1 and COEF2, for the two system transfer characteristics considered (see section 3. 2. 3). Finally, the input data cards for XMO(I) must be the weighted averaged pseudo IC's of the two model transfer characteristics.

The subroutines COEF1 and COEF2 perform the same function as COEF except for two systems with the same denominator. They are written similar to COEF (for example see page 341) with the following format for their names:

COEF1 (ACOF, BCOF, PAR, CC)

COEF2 (BCOF, PAR, CC)

where ACOF is the denominator coefficient vector, BCOF in COEF1 is the numerator coefficient vector for the first system, BCOF in COEF2 is the numerator coefficient vector for the second system, and CC is the relative weighting factor for the two systems. The value of CC must be given in COEF1and (1-CC) must be given in COEF2 as CC.

The total computer program package consists of

Main Program
MRDES
SNORM
DIRCOS
CALSYS MOD 2.
COEF1
COEF2
DELPC

B. 3 Procedure for Using a Model-Referenced ISE Performance Index

The procedure for using a model-referenced ISE performance index in this computer program is quite similar to that described in B. 1, but there are some changes in the input data cards. In Chapter 5, section 5. the model-referenced ISE performance index is written as a quadratic functional of an augmented state vector consisting of the n-dimensional system state vector and the ℓ-dimensional model state vector. By defining a new $(n+\ell)$-dimensional autonomous system, whose coefficient matrix is partitioned into the original system and model coefficient matrices, and by appropriately redefining the state vector weighting matrix, Q, the optimization problem becomes of the same form as that for the basic Model PI method. The main computer program can be used without modification. All necessary changes occur in the subroutine CALSYS MOD 3.

Two subroutines, COEF and COEFM, must be written for use in CALSYS MOD 3. for each specific design problem using this type of performance index. COEF is of the same format as the example listed on page 341. It simply lists the functional relationships of the actual system's numerator and denominator coefficients to the free design parameters. COEFM presents the numerical values of the model's numerator and denominator coefficients and has the following name format:

COEFM (ALPHA, BETA, NM, MM)

where ALPHA and BETA are the model' denominator and numerator coefficient vectors respectively, and NM and MM are the orders of the model's denominator and numerator respectively. The numerical values of NM and MM must be given in COEFM.

Parameter constraints can be included in DELPC as discussed in B. 1.

The total computer program package consists of

Main Program
MRDES
SNORM
DIRCOS
CALSYS MOD 3.
COEF
COEFM
DELPC

B. 3. 1 Input Data Format

Since the main program is dimensioned for up to a 20th order system, the dimension of the augmented state vector (actual system plus model) can not be larger than 20. There can still be up to 10 free design parameters. The input data cards are basically of the same form as those described in B. 1. 1 but with the following change:

First Card

N	is the dimension of the augmented state vector, i. e. $N = n+\ell$ where n is the order of the actual system and ℓ is the order of the model ($N \leq 20$).
M	is the order of the actual system's numerator.
RXO	is set equal to −0. 0.
LI	is set numerically equal to N

Next Card(s)

PAR(I), I = 1, K

Same as before

Next Cards

XMO(I), I = 1, N

and

W(I), I = 1, N

These are irrelevant since RXO = −0. 0 but the appropriate number of data cards must be included.

Next Card(s)

ALPHA(I), I = 1, LI

Here ALPHA(I) is redefined as the row vector*

$$[\ 1 \ \vdots\ \underline{0}'_{\ell-1} \ \vdots\ -1 \ \vdots\ \underline{0}'_{n-1}\ ]$$

where $\underline{0}_{\ell-1}$ and $\underline{0}_{n-1}$ are (ℓ-1) and (n-1) dimensional null vectors respectively.

B. 4 Listing of Main Program and Subroutines

The main computer program and subroutines are listed on the following pages.

* This is equivalent to the vector $\underline{h}$ defined by (5-17) in Chapter 5.

```
      LL=N*(N+1)/2
      NN=N*N
      NK=N*K
      KN=K*(N-1)+1
      LIL=LI*(LI+1)/2
      KLI=NK+1
      LIK=LI*K
      LIN=LI*N
      REK=K
      DCK=1./SQRT(REK)
      ISAVE=IMAX
C
      IC=1
      DO 241 I=1,LI
      IL=I*(I+1)/2
      DO 240 J=IC,IL
      JI=J+1-IC
  240 Q(J)=ALPHA(I)*ALPHA(JI)
      IC=IL+1
  241 CONTINUE
C
C     INITIALIZE VARIABLES.
C
      PIOLD=.1E 20
      PIPC=0.
      PIX0=0.
      IT=0
      DO 25 I=1,LL
      XXIC(I)=0.0
      PIC(I)=0.0
   25 ZERO(I)=0.
      DO 27 I=1,K
      GX0(I)=0.
   27 GPC(I)=0.
C
      WRITE(6,1)
      WRITE(6,103)
      WRITE(6,3) N,M,K,STEP,ITMAX,H,IMAX,SNE,YNE,RX0,RPC,LI
      WRITE(6,109)
      WRITE(6,20) (ALPHA(I), I=1,LI)
      WRITE(6,104)
      WRITE(6,20) (XM0(I), I=1,N)
      WRITE(6,105)
      WRITE(6,20) (W(I), I=1,N)
      WRITE(6,106)
      WRITE(6,20) (Q(I), I=1,LIL)
      WRITE(6,107)
      WRITE(6,20) (PAR(I), I=1,K)
```

MAIN PROGRAM LISTING CONTINUED

```
C     START OF OPTIMIZATION ALGORITHM.
C
   30 CONTINUE
C
C     WITHIN EACH ITERATION, COMPUTE AN INTERMEDIATE POINT ONE STEP IN
C     THE DIRECTION OF THE LOCAL GRADIENT, THEN COMPUTE THE AVERAGE OF
C     THE DIRECTION COSINES OF THE GRADIENTS AT THESE TWO POINTS.
C     AT THE END OF THE ITERATION, THE NEW STEP WILL BE IN THIS
C     AVERAGE DIRECTION.
C
      DO 55 IA=1,2
C
  300 CONTINUE
      WRITE(6,12) STEP
      WRITE(6,20) (PAR(I), I=1,K)
      CALL CALSYS(F,AT,BT,XO,ALPHA,Q,HTQ,PAR)
      WRITE(6,20) (F(I), I=N,NN,N)
      WRITE(6,20) (XO(I), I=1,N)
C
C     COMPUTE CONTRIBUTION TO THE PERFORMANCE INDEX, PIX.
C
      IMAX=ISAVE
      IACT=0
      YN(1)=YNE
      DO 3000 I=1,LL
 3000 P(I)=PIC(I)
      CALL MRDES(F,ZERO,Q,P,N,H,SNE,YN,IMAX,IACT,0)
      IF(IMAX-1)302,301,302
  301 WRITE(6,19)
      GO TO 303
  302 WRITE(6,13) IACT
  303 WRITE(6,20) (YN(I), I=1,10)
      CALL MPRD(P,XO,AMAT,N,N,1,0,1)
      PIX=0.0
      DO 304 I=1,N
  304 PIX=PIX+XO(I)*AMAT(I)
C
C     IF RXO IS ZERO, BYPASS COMPUTING PIXO.
C
      IF(RXO)308,308,305
C
  305 DO 306 I=1,N
  306 AVEC(I)=XO(I)-XMO(I)
      CALL MPRD(W,AVEC,BVEC,N,N,2,0,1)
      PIXO=0.0
      DO 307 I=1,N
  307 PIXO=PIXO+AVEC(I)*BVEC(I)
  308 CONTINUE
```

```
C     IF RPC IS ZERO, BYPASS CALLING THE SUBROUTINE DELPC, THAT COMPUTES
C     THE VALUE AND GRADIENT OF THE PARAMETER CONSTRAINT FUNCTION.
C
      IF(RPC)310,310,309
C
  309 CALL DELPC(GPC,PIPC,PAR,RPC)
  310 CONTINUE
C
C     COMPUTE CURRENT VALUE AND CHANGE IN PERFORMANCE INDEX.
C
      PI=PIX+PIX0+PIPC
      DPI=PIOLD-PI
      WRITE(6,20) PI, DPI
C
C     STEP SIZE CONTROL AND STABILITY CONSTRAINT FOR ALGORITHM.
C
      IF(IT+IA-2)320,311,311
C
C     IF PERFORMANCE INDEX INCREASED, CUT STEP SIZE IN HALF.
C
  311 IF(DPI)312,312,320
  312 STEP=0.5*STEP
      IF(IA-1)313,313,315
C
C     IF PI INCREASED BETWEEN ITERATIONS, GO BACK HALF WAY TO THE
C     AVERAGE PARAMETER VALUE AND TRY AGAIN.  REPEAT THIS NO MORE THAN
C     FOUR TIMES.
C
  313 IF(KUT-3)3131,3131,320
 3131 DO 314 I=1,K
  314 PAR(I)=AVPAR(I)-STEP*GPI(I)
      KUT=KUT+1
      GO TO 300
C
C     IF PI INCREASED WITHIN THIS ITERATION, TAKE AN INTERMEDIATE POINT
C     HALF WAY BACK TO THE CURRENT PARAMETER VALUE.
C
  315 CONTINUE
      DO 316 I=1,K
  316 PAR(I)=PAROLD(I)-STEP*GPIOLD(I)
      GO TO 300
C
C     TERMINATE IF THE NUMBER OF ITERATIONS EXCEEDS ITMAX.
C
  320 IF(IA-1)3201,3201,3203
 3201 IF(ITMAX-IT)72,72,3202
```

```
 3202 WRITE(6,11) IT
      WRITE(6,12) STEP
      WRITE(6,20) (PAR(I), I=1,K)
      WRITE(6,14) PI
      WRITE(6,20) (F(I), I=N,NN,N)
      KUT=1
 3203 CONTINUE
C
C     COMPUTE GRADIENT OF THE PERFORMANCE INDEX.
C
C     COMPUTE THE GRADIENT OF THE INTEGRAL OF XQX, CALLED GIX.
C
      CALL MPRD(BT,AMAT,GIX,K,N,0,0,1)
C
      IMAX=ISAVE
      IACT=0.0
      YN(1)=YNE
      IC=1
      DO 322 I=1,N
      IL=I*(I+1)/2
      DO 321 J=IC,IL
      JI=J+1-IC
  321 XXO(J)=XO(I)*XO(JI)
      IC=IL+1
  322 CONTINUE
      CALL MTRA(F,FT,N,N,0)
      DO 3220 I=1,LL
 3220 XX(I)=XXIC(I)
      CALL MRDES(FT,ZERO,XXO,XX,N,H,SNE,YN,IMAX,IACT,0)
      IF(IMAX-1)324,323,324
  323 WRITE(6,18)
      GO TO 325
  324 WRITE(6,13) IACT
  325 WRITE(6,20) (YN(I), I=1,10)
C
      CALL MPRD(AT,XX,AMAT,K,N,0,1,N)
      CALL MPRD(AMAT,P,BMAT,K,N,0,1,N)
      DO 326 I=KN,NK
      J=1-KN+I
  326 GIX(J)=GIX(J)-BMAT(I)
C
      IF(N-LI)3260,3262,3262
C
 3260 CALL MPRD(AMAT,HTQ,BMAT,K,N,0,0,LI)
      DO 3261 I=KLI,LIK
      J=1-KLI+I
 3261 GIX(J)=GIX(J)-BMAT(I)
C
 3262 CONTINUE
```

```
C     IF RXO IS ZERO, BYPASS COMPUTING GRADIENT OF INITIAL CONDITION.
C
      IF(RX0)329,329,327
C
C     COMPUTE THE GRADIENT OF (XM0-X0)T*W*(XM0-X0), CALLED GX0.
C
  327 CALL MPRD(BT,BVEC,GX0,K,N,0,0,1)
      DO 328 I=1,K
  328 GX0(I)=RX0*GX0(I)
  329 CONTINUE
C
C     COMPUTE THE TOTAL GRADIENT OF THE PERFORMANCE INDEX, GPI.
C
      DO 540 I=1,K
  540 GPI(I)=GIX(I)+GX0(I)+GPC(I)
C
C     TERMINATE IF THE GRADIENT IS ESSENTIALLY ZERO, OTHERWISE COMPUTE
C     THE AVERAGE OF THE DIRECTION COSINES OF THE GRADIENTS FOR THE TWO
C     POINTS WITHIN THIS ITERATION.
C
      CALL SNORM(GPI,GPISN,K,2)
      IF(GPISN-(.1E-20))80,80,541
C
  541 IF(IA-1)542,542,544
C
C     DIRECTION COSINE OF GRADIENT AT FIRST POINT.
C
  542 CALL DIRCOS(GPI,GPIEN,GPIOLD,K)
C
C     STOPPING CONDITION.  TERMINATE IF BOTH THE DECREASE IN THE
C     PERFORMANCE INDEX IS LESS THAN.01 PER CENT, AND THE LENGTH OF THE
C     GRADIENT TIMES THE CURRENT STEP SIZE IS LESS THAN .1 PER CENT
C     OF THE PERFORMANCE INDEX.
C
      IF(PI-(.1E-10))70,70,5421
 5421 IF(ABS(DPI)-.0001*PI)5422,5422,5423
 5422 IF(GPIEN*STEP-.001*PI)71,71,5423
 5423 CONTINUE
C
C     COMPUTE AN INTERMEDIATE POINT FOR THIS ITERATION.
C
      STEP=0.5*STEP
      DO 543 I=1,K
      PAROLD(I)=PAR(I)
  543 PAR(I)=PAR(I)-STEP*GPIOLD(I)
      PIOLD=PI
      GO TO 55
```

```
C     DIRECTION COSINE OF GRADIENT AT INTERMEDIATE POINT.
C
  544 CALL DIRCOS(GPI,GPIEN,GPI,K)
C
C     ADD THE DIRECTION COSINES FOR BOTH POINTS OF THIS ITERATION.
C
      DO 545 I=1,K
  545 GPI(I)=GPI(I)+GPIOLD(I)
      CALL SNORM(GPI,GPISN,K,2)
C
C     IF THE SUM OF DIRECTION COSINES HAPPENS TO BE ALMOST ZERO, DEFINE
C     ARBITRARILY, A GRADIENT OF UNITY LENGTH AND WITH ALL COMPONENTS
C     EQUAL, OTHERWISE CONTINUE COMPUTATIONS.
C
      IF(GPISN-(.1E-20))546,546,548
  546 DO 547 I=1,K
  547 GPI(I)=DCK
      GO TO 55
C
C     COMPUTE THE DIRECTION COSINE OF THE SUM OF THE DIRECTION COSINES
C     FOR THE TWO POINTS, AND STORE IT IN GPI.
C
  548 CALL DIRCOS(GPI,GPIEN,GPI,K)
   55 CONTINUE
C
      IT=IT+1
      DO 550 I=1,LL
      XXIC(I)=XX(I)
  550 PIC(I)=P(I)
C
C     COMPUTE THE AVERAGE VALUE OF THE PARAMETER VECTOR WITHIN THIS
C     ITERATION AND STEP FROM THAT POINT IN THE DIRECTION OF THE
C     DIRECTION COSINE OF THE AVERAGE DIRECTION COSINES.
C
      STEP=2.0*STEP
      DO 56 I=1,K
      AVPAR(I)=.5*(PAR(I)+PAROLD(I))
   56 PAR(I)=AVPAR(I)-STEP*GPI(I)
C
      GO TO 30
C
   70 WRITE(6,6)
      GO TO 80
   71 WRITE(6,7)
      GO TO 80
   72 WRITE(6,5)
   80 CONTINUE
```

MAIN PROGRAM LISTING CONCLUDED

```
C     COMPUTE THE CHARACTERISTIC POLYNOMIAL COEFFICIENTS
C
      DO 81 I=1,N
      IN=I*N
   81 AMAT(I)=-F(IN)
C
C     COMPUTE THE NUMERATOR COEFFICIENTS CORRESPONDING TO THE INITIAL
C     CONDITIONS AND THE CHARACTERISTIC POLYNOMIAL COEFFICIENTS.
C
      NM=N-1
      BMAT(1)=-AMAT(1)*XO(1)
      DO 83 I=2,NM
      C=0.
      NI=N-I
      NII=NI+2
      DO 82 J=1,NI
      JI=J+I
      JJ=J+1
   82 C=C+AMAT(JI)*XO(JJ)
   83 BMAT(I)=XO(NII)+C
      BMAT(N)=XO(2)
C
      WRITE(6,11) IT
      WRITE(6,12) STEP
      WRITE(6,8)
      WRITE(6,20) (PAR(I), I=1,K)
      WRITE(6,14) PI
      WRITE(6,10)
      WRITE(6,20) (XO(I), I=1,N)
      WRITE(6,22)
      WRITE(6,20) (AMAT(I), I=1,N)
      WRITE(6,123)
      WRITE(6,20) (BMAT(I), I=1,N)
C
      CALL EXIT
      END
```

```
C    ***********************************************************************
C
C       HERMAN A. REDIESS,  MIT, DEPT. OF AERONAUTICS AND ASTRONAUTICS
C       MARCH 1968
C
C    ***********************************************************************
C
C       SUBROUTINE MRDES
C
C       PURPOSE
C          COMPUTES THE TIME SOLUTION OF THE STATIONARY MATRIX RICCATI
C          DIFFERENTIAL EQUATION
C
C            DY(T)/DT = AT*Y(T) + Y(T)*A - Y(T)*B*Y(T) + C ,   Y(0) = YO
C
C          WHERE A IS STABLE AND B, C, AND YO ARE SYMMETRIC AND POSITIVE
C          SEMI-DEFINITE.
C
C       USAGE
C          CALL MRDES(A,B,C,Y,N,H,SNE,YN,IMAX,IACT,KOUT)
C
C       DESCRIPTION OF PARAMETERS
C          A    - NAME OF COEFFICIENT MATRIX (NXN), GENERAL STORAGE MODE
C          B    - MATRIX IN QUADRATIC TERM (NXN), SYMMETRIC STORAGE MODE
C          C    - CONSTANT FORCING MATRIX (NXN), SYMMETRIC STORAGE MODE
C          Y    - UPON ENTRY, Y IS THE INITIAL CONDITION Y(0) AND UPON
C                 RETURN, IT IS THE SOLUTION (NXN), SYM. STORAGE MODE
C          N    - DIMENSION OF MATRICES. NOT GREATER THAN 20.
C          H    - TIME STEP SIZE
C          SNE  - MAX ALLOWABLE MAGNITUDE OF DY(T)/DT NORM FOR STEADY
C                 STATE TEST.
C          YN   - A VECTOR FOR STORING PAST 10 VALUES OF Y NORM USED
C                 IN A TEST FOR STEADY STATE.  THE FIRST ELEMENT, YN(1),
C                 IS USED AT ENTRY TO SPECIFY THE MAX ALLOWABLE ERROR
C                 IN THE NORM OF Y AT STEADY STATE.
C          IMAX - MAX NUMBER OF TIME STEPS ALLOWED. ALSO USED FOR AN
C                 OUTPUT FLAG TO DENOTE REACHING STEADY STATE OR IMAX.
C                    0 - STEADY STATE REACHED
C                    1 - IACT=IMAX
C          IACT - ACTUAL NUMBER OF TIME STEPS WHEN Y IS RETURNED TO THE
C                 MAIN PROGRAM.  CORRESPONDING TIME IS T = H*IACT.
C                 MAIN PROGRAM MUST SET IACT=0 INITIALLY.
C          KOUT - CODE TO SPECIFY THE KIND OF OUTPUT DESIRED.
C                    0 - RETURN ONLY THE STEADY STATE SOLUTION
C                    K - RETURN Y AFTER EVERY K TIME STEPS UP TO
C                        STEADY STATE, WHERE K IS ANY NON ZERO INTEGER.
C                        SEE NOTE UNDER REMARKS FOR THIS CASE.
C
C       SUBROUTINES AND FUNCTIONS REQUIRED
C          MPRD, MTRA, MADD, MSTR, MSUB, SNORM, ABS
```

MRDES LISTING CONTINUED

```
C     REMARKS
C        THE SUBROUTINE MRDES IS A MODIFICATION OF A PROCEDURE PRESENTED
C        BY ATHANS AND LEVINE, MIT ELECTRONIC SYSTEMS LABORATORY REPORT
C        ESL-R-276, JULY, 1966, USING A RUNGA-KUTTA SCHEME.  MRDES IS
C        WRITTEN USING THE COMPRESSED STORAGE MODES OF THE IBM SSP AND
C        USES SEVERAL SSP SUBROUTINES.
C
C        THE SOLUTION, Y(T), IS STORED IN SYMMETRIC STORAGE MODE.
C        SINCE THE SOLUTION IS STORED IN THE SAME LOCATION AS THE
C        INITIAL CONDITION, IT IS NECESSARY TO SAVE THE I.C. IN SOME
C        OTHER LOCATION PRIOR TO ENTRY INTO MRDES IF THE I.C. IS TO
C        BE RETAINED.
C
C        IF B=0, ALL COMPUTATIONS INVOLVING B ARE SKIPPED.
C
C        TO INCREASE THE DIMENSION OF THE MATRIX EQUATION ABOVE 20, IT
C        IS ONLY NECESSARY TO CHANGE THE TWO DIMENSION STATEMENTS FOR
C        THE DUMMY ARRAYS TO VALUES OF L AND NN RESPECTIVELY.
C
C        SNORM IS A SUBROUTINE TO OBTAIN THE SUP-NORM OF AN NXN MATRIX
C        STORED IN ANY STORAGE MODE USED IN IBM SSP, BUT IS NOT PART
C        OF THE SSP SO IT MUST BE PROVIDED BY THE USER.
C
C        NOTE.  IF Y IS TO BE RETURNED EVERY K TIME STEPS, THE MAIN
C        PROGRAM MUST SET UP A LOOP TO RE-ENTER MRDES AFTER EACH RETURN.
C        BE SURE NOT TO DISTROY THE CURRENT VALUES OF THE SUBROUTINES
C        ARGUMENTS BEFORE RE-ENTRY.  ALSO THE MAIN PROGRAM MUST CHECK
C        THE FLAG, IMAX, AFTER EACH RETURN.  UPON RETURN, IF IMAX IS
C        GREATER THAN 1 RE-ENTER MRDES. IF IMAX=0 OR 1, DO NOT REENTER.
C        AN EXAMPLE OF AN APPROPRIATE LOOP IN A MAIN PROGRAM IS
C
C           MAIN PROGRAM
C
C
C                 ...........
C                 ...........
C              10 CALL MRDES(A,B,C,Y,N,H,SNE,YN,IMAX,IACT,KOUT)
C                 CALL MXOUT(IY,Y,N,N,1,60,120,2)
C                 TIME=IACT*H
C                 WRITE(6,15) TIME
C                 IF(IMAX-1)11,11,10
C              11 CONTINUE
C                 ...........
C                 ...........
C
C        WHERE MXOUT IS A SUBROUTINE FOR PRINTING THE MATRIX Y.
```

```
C      METHOD
C         THE FOLLOWING RUNGA-KUTTA RECURSION FORMULA IS USED
C
C            Y(K+1) = Y(K) + (G1(K) + 2*G2(K) + 2*G3(K) + G4(K))/6
C
C         WHERE
C            G1(K) = H*(AT*Y(K) + Y(K)*A - Y(K)*B*Y(K) + C)
C
C            G2(K) = H*(AT*(Y(K) + .5*G1(K)) + (Y(K) + .5*G1(K))*A
C                   -(Y(K) + .5*G1(K))*B*(Y(K) + .5*G1(K)) + C)
C
C            G3(K) = H*(AT*(Y(K) + .5*G2(K)) + (Y(K) + .5*G2(K))*A
C                   -(Y(K) + .5*G2(K))*B*(Y(K) + .5*G2(K)) + C)
C
C            G4(K) = H*(AT*(Y(K) + G3(K)) + (Y(K) + G3(K))*A
C                   -(Y(K) + G3(K))*B*(Y(K) + G3(K)) + C)
C
C         AND AT REPRESENTS THE TRANSPOSE OF A.  THE TIME STEP SIZE, H,
C         IS CONSTANT.
C
C         MRDES INCLUDES TWO TESTS FOR STEADY STATE.  THE USER MAY
C         CHOOSE TO USE EITHER OR BOTH.  ONE TEST SPECIFIES THE MAX
C         ALLOWABLE VALUE OF THE NORM OF THE DERIVATIVE AT STEADY STATE,
C         SNE.  THE SECOND IS A TEST FOR THE ALLOWABLE ERROR IN THE NORM
C         OF THE SOLUTION AT STEADY STATE.  THE USER SPECIFIES THE PER
C         CENT ALLOWABLE ERROR IN THE NORM OF Y IN YN(1), I.E.  0.1
C         PER CENT OF YNORM WOULD BE YN(1)=0.001 .  THE SUBROUTINE
C         CHECKS THE PER CENT DIFFERENCE BETWEEN CURRENT YNORM AND
C         YNORM 10 TIME STEPS BACK.  IF THE CHANGE IS LESS THAN THE
C         ALLOWABLE ERROR, THEN YNORM IS COMPARED TO 3 AND 7 TIME STEPS
C         BACK.  IF THE CHANGE FOR 3 STEPS IS LESS THAN.3*(THE CHANGE FOR
C         10 TIME STEPS), AND FOR 7 STEPS IS LESS THAN .6*(THE CHANGE
C         FOR 10 TIME STEPS), THEN IT IS CONSIDERED STEADY STATE.
C
C         THE USER MUST SPECIFY A MAX NUMBER OF ITERATIONS, IMAX, IN CASE
C         NEITHER STEADY STATE TESTS ARE SATISFIED IN A REASONABLE TIME
C
C  *****************************************************************
C
       SUBROUTINE MRDES(A,B,C,Y,N,H,SNE,YN,IMAX,IACT,KOUT)
       DIMENSION A(1), B(1), C(1), Y(1), YN(10)
C
C      THE FOLLOWING DUMMY ARRAYS ARE ONLY USED WITHIN THE SUBROUTINE.
C
       DIMENSION X(210),XBX(210),GG(210),GSUM(210)
       DIMENSION XA(400), AX(400),XB(400)
```

```
      NN=N*N
      L=N*(N+1)/2
      KK=0
      IF(IACT)1,1,3
    1 YNE=YN(1)
      YNORM=1.
      DO 2 I=1,10
    2 YN(I)=10.**I
    3 CONTINUE
      CALL SNORM(B,BNORM,N,1)
    8 KK=KK+1
    9 IACT=IACT+1
C
C     THE ARRAYS X AND GG WILL BE USED TO REPRESENT SUCCESSIVELY, Y,
C     Y+.5G1, Y+.5G2, Y+G3, AND G1, G2, G3, AND G4 RESPECTIVELY.
C
      DO 10 I=1,L
   10 X(I)=Y(I)
      DO 40 J=1,4
C
C     COMPUTE A*X + X*A AND STORE IN AX TEMPERARILY.
C
      CALL MPRD(X,A,XA,N,N,1,0,N)
      CALL MTRA(XA,AX,N,N,0)
      CALL MADD(AX,XA,AX,N,N,0,0)
C
C     STORE AX + XA (NOW IN AX) IN SYMMETRIC STORAGE MODE AND CALL IT GG
C
      CALL MSTR(AX,GG,N,0,1)
C
C     COMPUTE GG= A*X +X*A + C
C
      CALL MADD(GG,C,GG,N,N,1,1)
C
C     IF NORM OF B IS ZERO, BYPASS COMPUTATIONS INVOLVING B.
C
      IF(BNORM)13,13,11
   11 CALL MPRD(X,B,XB,N,N,1,1,N)
      CALL MPRD(XB,X,XA,N,N,0,1,N)
C
C     THE PRODUCT X*B*X IS TEMPERARILY STORED IN XA. THE MATRIX X*B*X
C     SHOULD BE SYMMETRIC BUT ROUND OFF ERRORS MAY CAUSE IT TO BE
C     SLIGHTLY UNSYMMETRICAL.  IT IS MADE SYMMETRIC BELOW.
C
      CALL MTRA(XA,AX,N,N,0)
      CALL MADD(XA,AX,XA,N,N,0,0)
```

```
      DO 12 I=1,NN
   12 XA(I)=.5*XA(I)
C
C     STORE X*B*X (NOW IN XA) IN SYMMETRIC STORAGE MODE AS XBX.
C
      CALL MSTR(XA,XBX,N,0,1)
C
C     COMPUTE GG= A*X + X*A - X*B*X + C
C
      CALL MSUB(GG,XBX,GG,N,N,1,1)
C
C     TERMINATE IF NORM OF DY(T)/DT, WHICH IS GIVEN HERE AS GG (J=1),
C     IS LESS THAN THE ALLOWABLE LIMIT AT STEADY STATE, SNE.
C     HOWEVER, TERMINATE AFTER THIS ITERATION.
C
   13 IF(J-1)14,14,18
   14 CALL SNORM(GG,GNORM,N,1)
C
C     FOR J=1, X IS EQUAL TO Y AND FOR J=2,3,AND 4 THE FOLLOWING LOGIC
C     SETS X EQUAL TO Y+.5G1, Y+.5G2, AND Y+G3 RESPECTIVELY.  GG TAKES
C     ON THE VALUES OF G1, G2, G3, AND G4 SUCCESSIVELY AS J=1,2,3,4.
C
   18 IF(J-2)20,25,19
   19 IF(J-4)30,35,35
   20 DO 21 I=1,L
      GSUM(I)=GG(I)*H/6.
C
C     HERE H*GG=G1 AND X=Y+.5G1
C
   21 X(I)=Y(I)+.5*H*GG(I)
      GO TO 40
   25 DO 26 I=1,L
      GSUM(I)=GSUM(I)+GG(I)*H/3.
C
C     HERE H*GG=G2 AND X=Y+.5G2
C
   26 X(I)=Y(I)+.5*H*GG(I)
      GO TO 40
   30 DO 31 I=1,L
      GSUM(I)=GSUM(I)+GG(I)*H/3.
C
C     HERE H*GG=G3 AND X=Y+G3
C
   31 X(I)=Y(I)+H*GG(I)
      GO TO 40
   35 DO 36 I=1,L
   36 Y(I)=Y(I)+GSUM(I)+GG(I)*H/6.
```

MRDES LISTING CONCLUDED

```
      IF(SNE-GNORM)40,45,45
   40 CONTINUE
C
C     CHECK THE CHANGE IN THE NORM OF Y OVER 10 TIME STEPS.
C
      DO 401 I=1,9
      IO=10-I
      II=11-I
  401 YN(II)=YN(IO)
      YN(1)=YNORM
      CALL SNORM(Y,YNORM,N,1)
      TEST=ABS(1-YN(10)/YNORM)
C
C     TERMINATE IF THE NORM OF Y TEST IS SATISFIED.
C
      IF(TEST-YNE)402,402,41
  402 IF(ABS(1-YN(3)/YNORM)-.3*TEST)403,403,41
  403 IF(ABS(1-YN(7)/YNORM)-.6*TEST)45,45,41
C
C     CHECK THE KIND OF OUTPUT DESIRED.  IF KOUT=0, CONTINUE COMPUTING
C     AND RETURN ONLY THE STEADY STATE VALUE OF Y.  IF KOUT=K, COMPUTE
C     FOR K TIME STEPS THEN RETURN Y.
C
   41 IF(IACT-IMAX)42,46,46
   42 IF(KOUT)9,9,43
   43 IF(KOUT-KK)50,50,8
C
C     USE IMAX AS A FLAG TO INDICATE THAT EITHER IACT=IMAX OR STEADY
C     STATE HAS BEEN REACHED, BY SETTING IMAX=0 IF STEADY STATE , OR
C     IMAX=1 IF IACT=IMAX, PRIOR TO THE FINAL RETURN.
C
   45 IMAX=0
      GO TO 50
   46 IMAX=1
   50 RETURN
      END
```

```
C     ********************************************************************
C
C        HERMAN A. REDIESS,  MIT, DEPT. OF AERONAUTICS AND ASTRONAUTICS
C        MARCH 1968
C
C     ********************************************************************
C
C        SUBROUTINE SNORM
C
C        PURPOSE
C           COMPUTES THE SUP-NORM OF AN NXN MATRIX OR AN NX1 VECTOR.
C
C        USAGE
C           CALL SNORM(A,B,N,MS)
C
C        DESCRIPTION OF PARAMETERS
C           A - NAME OF MATRIX OR VECTOR
C           B - NAME OF THE SUP-NORM OF A
C           N - DIMENSION OF A
C           MS - ONE DIGIT NUMBER FOR STORAGE MODE OF A
C                 0 - GENERAL
C                 1 - SYMMETRIC
C                 2 - DIAGONAL OR VECTOR
C
C        FUNCTIONS REQUIRED
C           ABS
C
C     ********************************************************************
C
         SUBROUTINE SNORM(A,B,N,MS)
         DIMENSION A(1)
         B=0.
         IF(MS-1)10,11,12
      10 L=N*N
         GO TO 15
      11 L=N*(N+1)/2
         GO TO 15
      12 L=N
      15 DO 20 I=1,L
         IF(ABS(A(I))-B)20,20,18
      18 B=ABS(A(I))
      20 CONTINUE
         RETURN
         END
```

```
C   ****************************************************************
C
C      HERMAN A. REDIESS,  MIT, DEPT. OF AERONAUTICS AND ASTRONAUTICS
C      MARCH 1968
C
C   ****************************************************************
C
C      SUBROUTINE DIRCOS
C
C      PURPOSE
C         COMPUTES THE LENGTH AND DIRECTION COSINES OF A VECTOR
C
C      USAGE
C         CALL DIRCOS(A,B,DC,N)
C
C      DESCRIPTION OF PARAMETERS
C         A - NAME OF VECTOR
C         B - LENGTH OF VECTOR A
C         DC - VECTOR CONTAINING THE DIRECTIONAL COSINES OF VECTOR A
C         N - DIMENSION OF VECTORS A AND DC
C
C      FUNCTIONS REQUIRED
C         SQRT
C
C   ****************************************************************
C
      SUBROUTINE DIRCOS(A,B,DC,N)
      DIMENSION A(1), DC(1)
      C=0.
      DO 10 I=1,N
   10 C=C+A(I)*A(I)
      B=SQRT(C)
      DO 11 I=1,N
   11 DC(I)=A(I)/B
      RETURN
      END
```

```
C   ******************************************************************
C
C      HERMAN A. REDIESS,  MIT, DEPT. OF AERONAUTICS AND ASTRONAUTICS
C      JUNE 1968
C
C   *******************************************************************
C
C      SUBROUTINE CALSYS MOD 1
C
C      COMPUTES F, AT, BT, XO, Q, AND HTQ FOR CURRENT VALUE OF PAR.  THIS
C      SUBROUTINE IS ONLY VALID FOR SYSTEMS IN CANONICAL (PHASE VARIABLE)
C      FORM.
C
C   *******************************************************************
C
       SUBROUTINE CALSYS(F,AT,BT,XO,ALPHA,Q,HTQ,PAR)
C
       DIMENSION F(1),AT(1),BT(1),XO(1),PAR(1),Q(1),HTQ(1),ALPHA(1)
       DIMENSION ACOF(20), BCOF(20), P(10), AVEC(20)
C
       COMMON N, M, K, LI
C
       NN=N*N
       MI=M+1
       NM=N-M
       NMI=NM+1
       NMJ=NM+2
C
       DO 10 I=1,K
    10 P(I)=PAR(I)
C
       L=1
    11 CONTINUE
       PAR(L)=1.1*P(L)
       NUM=0
C
    12 CONTINUE
C
       CALL COEF(ACOF,BCOF,PAR)
C
       XO(1)=-BCOF(1)/ACOF(1)
       DO 22 I=2,NM
    22 XO(I)=0.0
       XO(NMI)=BCOF(MI)
       DO 25 I=NMJ,N
       C=0.0
       II=I-1
       NI=N-I+2
```

```
      DO 24 J=NMI,II
      JN=J+N-I+1
   24 C=C-ACOF(JN)*X0(J)
   25 X0(I)=BCOF(NI)+C
C
      IF(L-K)26,26,40
C
   26 IF(NUM-1)27,29,29
C
   27 NUM=1
      DO 28 I=1,N
      IKL=(I-1)*K+L
      AT(IKL)=ACOF(I)
      BT(IKL)=X0(I)
   28 CONTINUE
      PAR(L)=.9*P(L)
      GO TO 12
C
   29 DO 30 I=1,N
      IKL=(I-1)*K+L
      AT(IKL)=(AT(IKL)-ACOF(I))/(.2*P(L))
      BT(IKL)=(BT(IKL)-X0(I))/(.2*P(L))
   30 CONTINUE
      PAR(L)=P(L)
      L=L+1
      IF(L-K)11,11,12
C
   40 CONTINUE
      DO 41 I=1,NN
   41 F(I)=0.0
      DO 42 I=2,N
      J=(I-1)*(N+1)
   42 F(J)=1.0
      DO 43 I=1,N
      IN=I*N
   43 F(IN)=-ACOF(I)
C
      IF(N-LI) 44,50,50
C
   44 DO 45 I=1,N
   45 AVEC(I)=ALPHA(I)-ACOF(I)
      DO 47 I=1,LI
      NI=N*(I-1)
      DO 46 J=1,N
      JI=J+NI
   46 HTQ(JI)=AVEC(J)*ALPHA(I)
   47 CONTINUE
```

CALSYS MOD 1 LISTING CONCLUDED

```
C
      IC=1
      DO 49 I=1,N
      IL=I*(I+1)/2
      DO 48 J=IC,IL
      JI=J+1-IC
   48 Q(J)=AVEC(I)*AVEC(JI)
      IC=IL+1
   49 CONTINUE
C
   50 RETURN
      END
```

```
C   *****************************************************************
C
C      HERMAN A. REDIESS,  MIT, DEPT. OF AERONAUTICS AND ASTRONAUTICS
C      AUG. 1968
C
C   *****************************************************************
C
C      SUBROUTINE CALSYS MOD 2
C
C      FOR WEIGHTED AVERAGED PSEUDO IC VECTOR APPROACH TO MULTIVARIABLE
C      SYSTEM DESIGN
C
C      COMPUTES F, AT, BT, X0, Q, AND HTQ FOR CURRENT VALUE OF PAR.  THIS
C      SUBROUTINE IS ONLY VALID FOR SYSTEMS IN CANONICAL (PHASE VARIABLE)
C      FORM.
C
C   *****************************************************************
C
       SUBROUTINE CALSYS(F,AT,BT,X0,ALPHA,Q,HTQ,PAR)
C
       DIMENSION F(1),AT(1),BT(1),X0(1),PAR(1),Q(1),HTQ(1),ALPHA(1)
       DIMENSION ACOF(20), BCOF(20), P(10), AVEC(20), CX0(20)
C
       COMMON N, M, K, LI
C
       NN=N*N
       MI=M+1
       NM=N-M
       NMI=NM+1
       NMJ=NM+2
C
       DO 10 I=1,K
    10 P(I)=PAR(I)
C
       L=1
    11 CONTINUE
       PAR(L)=1.1*P(L)
       NUM=0
C
    12 CONTINUE
C
       IA=1
       CALL COEF1(ACOF,BCOF,PAR,CC)
       GO TO 14
    13 CALL COEF2(BCOF,PAR,CC)
       IA=2
    14 CONTINUE
C
       X0(1)=-BCOF(1)/ACOF(1)
       DO 22 I=2,NM
    22 X0(I)=0.0
       X0(NMI)=BCOF(MI)
```

```
      DO 25 I=NMJ,N
      C=0.0
      II=I-1
      NI=N-I+2
      DO 24 J=NMI,II
      JN=J+N-I+1
   24 C=C-ACOF(JN)*X0(J)
   25 X0(I)=BCOF(NI)+C
C
      CALL DIRCOS(X0,XLEN,X0,N)
      IF(IA-1)250,250,252
  250 DO 251 I=1,N
  251 CX0(I)=CC*X0(I)
      GO TO 13
  252 DO 253 I=1,N
  253 X0(I)=CX0(I)+CC*X0(I)
C
      IF(L-K)26,26,40
C
   26 IF(NUM-1)27,29,29
C
   27 NUM=1
      DO 28 I=1,N
      IKL=(I-1)*K+L
      AT(IKL)=ACOF(I)
      BT(IKL)=X0(I)
   28 CONTINUE
      PAR(L)=.9*P(L)
      GO TO 12
C
   29 DO 30 I=1,N
      IKL=(I-1)*K+L
      AT(IKL)=(AT(IKL)-ACOF(I))/(.2*P(L))
      BT(IKL)=(BT(IKL)-X0(I))/(.2*P(L))
   30 CONTINUE
      PAR(L)=P(L)
      L=L+1
      IF(L-K)11,11,12
C
   40 CONTINUE
      DO 41 I=1,NN
   41 F(I)=0.0
      DO 42 I=2,N
      J=(I-1)*(N+1)
   42 F(J)=1.0
```

CALSYS MOD 2 LISTING CONCLUDED

```
      DO 43 I=1,N
      IN=I*N
   43 F(IN)=-ACOF(I)
C
      IF(N-LI) 44,50,50
C
   44 DO 45 I=1,N
   45 AVEC(I)=ALPHA(I)-ACOF(I)
      DO 47 I=1,LI
      NI=N*(I-1)
      DO 46 J=1,N
      JI=J+NI
   46 HTQ(JI)=AVEC(J)*ALPHA(I)
   47 CONTINUE
C
      IC=1
      DO 49 I=1,N
      IL=I*(I+1)/2
      DO 48 J=IC,IL
      JI=J+1-IC
   48 Q(J)=AVEC(I)*AVEC(JI)
      IC=IL+1
   49 CONTINUE
C
   50 RETURN
      END
```

```
C    ******************************************************************
C
C      HERMAN A. REDIESS,  MIT, DEPT. OF AERONAUTICS AND ASTRONAUTICS
C      JUNE 1968
C
C    ******************************************************************
C
C      SUBROUTINE CALSYS MOD 3
C
C      COMPUTES F, AT, BT, AND X0 FOR CURRENT VALUE OF PAR FOR THE MODEL-
C      REFERENCED ISE DESIGN METHOD WITH F AND X0 PARTITIONED AS
C
C                 ( F(MODEL) '    0     )             ( X0(MODEL) )
C           F  =  (..........'........)  ,    X0  =  (...........)
C                 (     0     ' F(SYS) )             (  X0(SYS)   )
C
C      WHERE F(MODEL) AND F(SYS) ARE IN CANONICAL (PHASE VARIABLE) FORM.
C
C    ******************************************************************
C
      SUBROUTINE CALSYS(F,AT,BT,X0,ALPHA,Q,HTQ,PAR)
C
      DIMENSION F(1),AT(1),BT(1),X0(1),PAR(1),Q(1),HTQ(1)
      DIMENSION ACOF(20), BCOF(20), P(10), ALPHA(20), BETA(20)
C
      COMMON N, M, K, LI
C
C      COMPUTE X0(MODEL), I.E. THE FIRST NM ELEMENTS OF X0.
C
      CALL COEFM(ALPHA,BETA,NM,MM)
C
      NN=N*N
      MMI=MM+1
      NMM=NM-MM
      NMMI=NMM+1
      NMMJ=NMM+2
C
      X0(1)=-BETA(1)/ALPHA(1)
      IF(NM-1)220,220,19
   19 CONTINUE
      DO 20 I=2,NMM
   20 X0(I)=0.0
      X0(NMMI)=BETA(MMI)
      DO 22 I=NMMJ,NM
      C=0.0
      II=I-1
      NI=NM-I+2
      DO 21 J=NMMI,II
      JN=J+NM-I+1
   21 C=C-ALPHA(JN)*X0(J)
```

```
   22 XO(I)=BETA(NI)+C
  220 CONTINUE
C
C     COMPUTE AT AND BT, AND COMPLETE XO.
C
      NMK=NM*K
      NMI=NM+1
      NMJ=NM+2
      NMS=N-MS
      NMSI=NMS+1
      NMSJ=NMS+2
      MSI=MS+1
C
      DO 23 I=1,NMK
      AT(I)=0.0
   23 BT(I)=0.0
C
      DO 24 I=1,K
   24 P(I)=PAR(I)
C
      L=1
   25 CONTINUE
      PAR(L)=1.1*P(L)
      NUM=0
C
   26 CONTINUE
C
      CALL COEF(ACOF,BCOF,PAR)
C
      XO(NMI)=-BCOF(1)/ACOF(1)
      DO 27 I=NMJ,NMS
   27 XO(I)=0.0
      XO(NMSI)=BCOF(MSI)
      DO 30 I=NMSJ,N
      C=0.0
      II=I-1
      NI=N-I+2
      DO 28 J=NMSI,II
      JN=J+N-NM-I+1
   28 C=C-ACOF(JN)*XO(J)
   30 XO(I)=BCOF(NI)+C
C
      IF(L-K)31,31,40
C
   31 IF(NUM-1)32,35,35
```

```
   32 NUM=1
      DO 33 I=NMI,N
      IKL=(I-1)*K+L
      INM=I-NM
      AT(IKL)=ACOF(INM)
      BT(IKL)=X0(I)
   33 CONTINUE
      PAR(L)=0.9*P(L)
      GO TO 26
C
   35 DO 36 I=NMI,N
      IKL=(I-1)*K+L
      INM=I-NM
      AT(IKL)=(AT(IKL)-ACOF(INM))/(0.2*P(L))
      BT(IKL)=(BT(IKL)-X0(I))/(0.2*P(L))
   36 CONTINUE
      PAR(L)=P(L)
      L=L+1
      IF(L-K)25,25,26
C
   40 CONTINUE
C
C     COMPUTE THE PARTITIONED COEFFICIENT MATRIX, F.
C
      DO 41 I=1,NN
   41 F(I)=0.0
      IF(NM-1)420,420,410
  410 CONTINUE
      DO 42 I=2,NM
      J=(I-1)*(N+1)
   42 F(J)=1.0
  420 CONTINUE
      DO 43 I=1,NM
      INM=(I-1)*N+NM
   43 F(INM)=-ALPHA(I)
C
      DO 44 I=NMJ,N
      J=(I-1)*(N+1)
   44 F(J)=1.0
      DO 45 I=NMI,N
      IN=I*N
      INM=I-NM
   45 F(IN)=-ACOF(INM)
C
      RETURN
      END
```

AN EXAMPLE OF SUBROUTINE COEF

```
      SUBROUTINE COEF(ACOF,BCOF,PAR)
C
      DIMENSION ACOF(1), BCOF(1), PAR(1)
C
      PAR(4)=0.0
C
      BCOF(1)= 0.0
      BCOF(2)= -15.01069-6.74563*PAR(3)
      BCOF(3)= -15.13365+0.1178*PAR(2)-6.71773*PAR(3)
      BCOF(4)= -5.23896+6.74579*PAR(2)+0.0279*PAR(3)
      BCOF(5)= -5.116
C
      ACOF(1)=-.0006235-.0004162*PAR(3)
      ACOF(2)=.17968-15.0106*PAR(1)-.03287*PAR(2)+.068038*PAR(3)
     1  -6.1449*PAR(4)-6.7456*PAR(1)*PAR(3)
      ACOF(3)=3.1285-15.1336*PAR(1)-.0004941*PAR(2)+1.5127*PAR(3)
     1  -6.2516*PAR(4)+.11779*PAR(1)*PAR(2)-6.7329*PAR(1)*PAR(3)
      ACOF(4)=3.0453-5.239*PAR(1)-.10533*PAR(2)+1.4293*PAR(3)
     1  -2.4316*PAR(4)+6.7458*PAR(1)*PAR(2)+.01279*PAR(1)*PAR(3)
      ACOF(5)=1.0847-5.116*PAR(1)-1.432*PAR(2)-.0025*PAR(3)-2.325*PAR(4)
C
      RETURN
      END
```

AN EXAMPLE OF SUBROUTINE DELPC

```
      SUBROUTINE DELPC(GPC,PIPC,PAR,RPC)
C
      DIMENSION GPC(1), PAR(1)
      COMMON    N,M,K,LI
C
      SQ=0.0
      DO 10 I=1,K
      SQ=SQ + PAR(I)*PAR(I)
   10 GPC(I)=2.0*RPC*PAR(I)
C
      PIPC=RPC*SQ
C
      RETURN
      END
```

REFERENCES

1. Draper, C. S.; "Flight Control", Journal of the Royal Aeronautical Society, Vol. 59, pp. 449-477, July 1955.

2. Perkins, C. D. and Hage, R. E.; Airplane Performance and Control, John Wiley and Sons, Inc., New York, 1949.

3. Petersen, F. S., Rediess, H. A., and Weil, J.; Lateral-Directional Control Characteristics of the X-15 Airplane, NASA TM X-726, 1962.

4. Whitaker, P. H., Schlundt, R. W., and McKinney, T.; Automatic Lateral Control Systems for Manned Re-entry Vehicles, MIT Instrumentation Laboratory Report R-563, December 1966.

5. Seckel, E. and Traybar, J. J.; "Piloting and VTOL Instrumentation", Astronautics and Aeronautics, Vol. 3, No. 9, pp. 60-65, September 1965.

6. Mueller, L. J.; "Problems Unique to VTOL Automatic Flight Control", Journal of Aircraft, Vol. 2, No. 5, pp. 357-360, September-October 1965.

7. Josephs, L. C., III and Hesse, W. J.; "Survey of Significant Technical Problems Unique to VSTOL Encountered in the Development of the XC-142A", Journal of Aircraft, Vol. 3, No. 1, pp. 3-10, January-February 1966.

8. Popik, M. J.; "Automatic Flight Control for All-Weather Landing", Sperry Engineering Review, pp. 2-9, Fall 1964.

9. Chestnut, H. and Meyer, R. W.; Servomechanisms and Regulating System Design, Volume 1, John Wiley and Sons, Inc., New York 1951.

10. Blakelock, J. H.; Automatic Control of Aircraft and Missiles, John Wiley and Sons, Inc., New York, 1965.

11. Wiener, N.; The Extrapolation, Interpolation, and Smoothing of Stationary Time Series, NDRC Report, Cambridge, Mass. 1942. (also published by the M. I. T. Press, Cambridge, Mass. 1949)

12. Kolmogoroff, A.; "Interpolation und Extrapolation von stationaren zufalligen Folgen", Bulletin de l'academie des sciences de U. R. S. S., Ser. Math. 5, pp. 3-14, 1941.

13. Hall, A. C.; The Analysis and Synthesis of Linear Servomechanisms, The M. I. T. Press, Cambridge, Mass., pp. 19-25, 1943.

14. James, H. M., Nichols, N. B., and Phillips, R. S.; Theory of Servomechanisms, MIT Radiation Laboratory Series, McGraw-Hill Book Co., Inc., New York, 1947.

15. Newton, G. C., Jr., Gould, I. A., and Kaiser, J. F.; Analytical Design of Linear Feedback Controls, John Wiley and Sons, Inc., New York, 1957.

16. Graham, D. and Lathrop, R. C.; "The Synthesis of Optimum Transient Response: Criteria and Standard Forms", Transactions of the AIEE, Vol. 72, pp. 273-288, November 1953.

17. Gibson, J. E., McVey, E. S., Leedham, C. D., Rekasius, Z. U., Schultz, D. G. and Sridhar, R.; Specifications and Data Presentation in Linear Control Systems, Technical Report AFMDV-TR-61-5, May 1961.

18. Aizerman, M. A.; Lectures on the Theory of Automatic Control, Gostekizdat, Moscow, USSR, 2nd ed., pp. 302-320, 1958.

19. Rekasius, Z. U.; "A General Performance Index for Analytical Design of Control Systems", IRE Transactions on Automatic Control, Vol. AC-6, No. 2, pp. 217-222, May 1961.

20. Spang, H. A.; "A Review of Minimization Techniques for Nonlinear Functions", SIAM Review, Vol. 4, No. 4, p. 343, October 1962.

21. Paiewonsky, B.; "Optimal Control: A Review of Theory and Practice", AIAA Journal, Vol. 3, No. 11, pp. 1985-2006, November 1965.

22. Roberts, J. D.; "A Method of Optimizing Adjustable Parameters in a Control System", The Institution of Electrical Engineers, Paper No. 4000M, November 1962.

23. Bingulac, S. and Koktovic, P.; "Automatic Optimization of Linear Feedback Control Systems on an Analog Computer", Annales de l' Association Internationale pour le Calcul Analogique, No. 1, January 1965.

24. Whitaker, H. P. and Potter, J. E.; Optimization of the Use of Automatic Flight Control Systems for Manned Aircraft, MIT Instrumentation Laboratory Report R-558, August 1966.

25 Athans, M; "The Status of Optimal Control Theory and Applications for Deterministic Systems", IEEE Transactions on Automatic Control, Vol. AC-11, No. 3, pp. 580-596, July 1966.

26. Rynaski, E. G. and Whitbeck, R. F.; The Theory and Application of Linear Optimal Control, Technical Report AFFDL-TR-65-28, U.S. Air Force Systems Command, January 1966.

27. Reynolds, P. A. and Rynaski, E. G.; "Application of Optimal Linear Control Theory to the Design of Aerospace Vehicle Control Systems" presented at the Optimum System Synthesis Conference, Wright-Patterson Air Force Base, Ohio, September 11-13, 1962.

28. Rynaski, E. G., Reynolds, P. A., and Shed, W. H.; Design of Linear Flight Control Systems Using Optimal Control Theory, Technical Documentary Report ASD-TDR-63-376, April 1964.

29. Tyler, J. S., Jr.; "The Characteristics of Model Following as Synthesized by Optimal Control", IEEE Transactions on Automatic Control, Vol. AC-9, No. 4, pp. 485-498, October 1964.

30 Gaul, J. W., Kaiser, R. P., Onega, G. T., and DeCanio, F. T.; Application of Optimal Control Theory to VTOL Flight Control System Design, Technical Report AFFDL-TR-67-102, U.S. Air Force Systems Command, September 1967.

31. Smith, R. E., Lum, E. L., and Yamamoto, T. G.; Application of Linear Optimal Theory to Control of Flexible Aircraft Ride Qualities, Technical Report AFFDL-TR-67-136, U.S. Air Force Systems Command, January 1968.

32. Yore, E. E.; Application of MarkIII-SOC (Self-Organizing Control) to Multivariable Control Problems, Part II, Optimal Decoupling Control Applied to the Lateral-Directional Axes of a Sweep Wing Aircraft, Technical Report AFFDL-TR-68-10, Part II, U.S. Air Force Systems Command, February 1968.

33. Ashkenas, I. L.; Some Open-and Closed-Loop Aspects of Airplane Lateral-Directional Handling Qualities, AGARD Report 533, May 1966.

34. Kidd, E. A. and Harper, R. P., Jr.; Fixed-Base and In-Flight Simulations of Longitudinal and Lateral-Directional Handling Qualities for Piloted Re-entry Vehicles, Technical Report ASD-TDR-61-362, U.S. Air Force Systems Command, October 1963.

35. Salmirs, S. and Tapscott, R. J.; The Effects of Various Combinations of Damping and Control Power on Helicopter Handling Qualities During Both Instrument and Visual Flight, NASA TN D-58, 1959.

36. Seckel, E.; Stability and Control of Airplanes and Helicopters, Academic Press, New York, 1964.

37. Chalk, C. R.; Additional Flight Evaluations of Various Longitudinal Handling Qualities in a Variable Stability Jet Fighter, WADC TR 57-719, Part 2, July 1958.

38. Cooper, G. B.; "Understanding and Interpreting Pilot Opinion", Aero Engineering Review, Vol. 16, No. 3, March 1957.

39. Chalk, C. R.; Fixed-Base Simulator Investigation of the Effects of L_α and True Speed on Pilot Opinion of Longitudinal Flying Qualities, ASD TR-63-399, U.S. Air Force Systems Command, November 1963.

40. Creer, B. Y., Stewart, J. D., Merrick, R. B., and Drinkwater, F. J., III; A Pilot Opinion Study of Lateral Control Requirements for Fighter-Type Aircraft, NASA Memorandum 1-29-59A, March 1959.

41. Li, Y. T. and Whitaker, H. P.; "Performance Characterization for Adaptive Control Systems", Proceedings of the First International Symposium on Optimizing and Adaptive Control, Rome, Italy, April 1962.

42. Kalman, R. E. and Bertram, J. E.; "Control System Analysis and Design Via the 'Second Method' of Lyapunov, Part 1 Continuous-Time Systems", ASME Journal of Basic Engineering, Vol. 82D, No. 2, pp. 371-393, June 1960.

43. Gilbert, E. G.; "Controllability and Observability in Multivariable Control Systems", Journal SIAM Control, Series A, Vol. 2, No. 1, 1963.

44. Johansen, D. E.; Optimal Control of Linear Stochastic Systems with Complexity Constants, Ph.D. Thesis, Harvard University, Cambridge, Massachusetts, May 1964.

45. Denham, W. F. and Speyer, J. L.; " Optimal Measurement and Velocity Correction Programs for Midcourse Guidance", AIAA Journal, Vol. 2, No. 5, pp. 896 - 907, May 1964.

46. Bellman, R.; Introduction to Matrix Analysis, McGraw-Hill Book Co., Inc., New York, 1960.

47. Anon,; System/360 Scientific Subroutine Package, (360 A-CM-03X) Version II, Programmer's Manual, IBM Application Program.

48. Athans, M. and Levine, W. S.; On the Numerical Solution of the Matrix Riccati Differential Equation Using a Runge-Kutta Scheme, MIT Electronic Systems Laboratory Report ESL-R-276, July 1966.

49. Clark, R. N.; Introduction to Automatic Control Systems, John Wiley and Sons, Inc., New York, 1962.

50. Hildebrand, F. B.; Methods of Applied Mathematics, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1952.

51. Tremant, R. A.; Operational Experiences and Characteristics of the X-15 Flight Control System, NASA TN D-1402, December 1962.

52. Todd, M. L.; Design of a System for Longitudinal Control of VTOL Aircraft, M.S. Thesis, M.I.T., June 1966. Also M.I.T. Instrumentation Laboratory Report No. T-465.

53. Athans, M. and Falb, P. L.; Optimal Control: An Introduction to the Theory and Its Applications, McGraw-Hill Book Co., New York, 1966.

54. Kalman, R. E., Englar, T. S. and Bucy, R. E.; Fundamental Study of Adaptive Control Systems, Vol. I and II, Technical Report ASD TR 61-27, U.S. Air Force Systems Command, March 1961 and March 1962.

55. Chang, S. S. L.; Synthesis of Optimum Control Systems, McGraw-Hill Book Co., New York, 1961.

56. Potter, J. E.; A Matrix Equation Arising in Statistical Filter Theory, MIT Experimental Astronomy Laboratory Report RE-9, February 1965.

BIOGRAPHICAL NOTE

Herman Arthur Rediess was born April 2, 1936, in Pinneo, Colorado; the youngest of six children, and only son, of Mr. and Mrs. Herman H. Rediess. He attended elementary school in Alameda, California, graduating from Alameda High School in June 1954.

Entering the University of California at Berkeley in the fall of that year, he graduated with honors in September 1959, receiving the degree of Bachelor of Science in Mechanical Engineering. The Institute of the Aeronautical Sciences Student Branch Scholastic Award for the year 1959 was awarded to Mr. Rediess. During this time, Mr. Rediess participated in the Cooperative Work-Study Program of the College of Engineering. Three six-month work periods were spent at the National Advisory Committee for Aeronautics, High Speed Flight Station at Edwards, California, as a research engineering aide.

In December 1959, Mr. Rediess was employed by the National Aeronautics and Space Administration, Flight Research Center, Edwards, California. He began part-time graduate study at the University of Southern California in September 1960, receiving a Master of Science degree in Aerospace Engineering in February 1964.

As a Research Engineer for NASA, Mr. Rediess worked in the area of aircraft stability and control analysis. In 1960 he became the Project Engineer on the F-100C variable stability airplane program and conducted in-flight simulation and handling qualities studies. The Airborne Simulation Section was established in 1962 with Mr. Rediess as the Section Head. Concurrently he was Program Manager of the General Purpose Airborne Simulator program which developed a Lockheed Jet Star into an advanced in-flight simulator.

Granted two years of Graduate Study Leave by the National Aeronautics and Space Administration, Mr. Rediess was admitted to the Graduate School of M. I. T. in September 1964, and became a candidate for the degree of Doctor of Philosophy in 1965. In September 1966, he was employed by the M. I. T. Instrumentation Laboratory while continuing his graduate studies. As a Research Assistant at the Instrumentation Laboratory, Mr. Rediess participated in studies for advanced flight control systems and landing guidance systems for VTOL aircraft in association with the M. I. T. Experimental Astronomy Laboratory.

Mr. Rediess was elected to membership in the Sigma Gamma Tau Honorary Society while at M. I. T. He is also a member of the American Institute of Aeronautics and Astronautics and the Institute of Electrical and Electronics Engineers.

Mr. Rediess is married to the former Sharon Purcell of Lancaster, California. They are the parents of Sharilyn, nine years old, and Nicholas Arthur, seven years old.

Papers presented:

"Theoretical Perspective of Adaptive Control Techniques and Modern Control Theory", AGARD Guidance and Control Panel and the Flight Mechanics Panel Joint Meeting on "Advanced Flight Control Concepts", Oslo, Norway, September 1968.

"An Advanced Method for Airborne Simulation", (co-author D. A. Deets), Testing of Manned Flight Systems Conference sponsored by AIAA, AFFTC, NASA FRC, Edwards Air Force Base, California, December 4 - 6, 1963.

BIOGRAPHICAL NOTE (cont)

"Lateral-Directional Control Characteristics of the X-15 Airplane", (co-authors F. S. Petersen and J. Weil), AIAA Section Meeting, Los Angeles, June 1962, (previously presented by F. S. Petersen at the X-15 Flight Test Conference sponsored by NASA, AFFTC, Edwards Air Force Base, California, December 1961).

Publications:

"An Advanced Method for Airborne Simulation", (co-author D. A. Deets), AIAA Journal of Aircraft, Vol. 1, No. 4, July-August 1964.

Aerodynamic-Derivative Characteristics of the X-15 Research Airplane as Determined from Flight Tests for Mach Numbers from 0.6 to 3.4, (co-authors R. B. Yancey and G. H. Robinson), NASA TN D-1060, January 1962.

Lateral-Directional Control Characteristics of the X-15 Airplane, (co-authors F. S. Petersen and J. Weil), NASA TMX-726, 1962.

Flight-Determined Stability and Control Derivatives of a Supersonic Airplane with a Low-Aspect-Ratio Unswept Wing and a Tee-Tail, (co-author W. H. Andrews), NASA MEMO 2-2-59H, April 1959.

Effects of Jet Exhausts on Flight-Determined Longitudinal and Lateral Dynamic Stability Characteristics of the Douglas D-558-II Research Airplane, (co-author C. H. Wolowics), NACA RM H57G09, August 23, 1957.